



TÉCNICO
LISBOA

**Coordinated Path-Following
for
Multi-Agent Fixed-Wing Aircraft**

Hugo Sequeira Costa

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisors: Prof. Afzal Suleman
Prof. Paulo Oliveira

Examination Committee

Chairperson: Prof. Fernando Lau
Supervisor: Prof. Afzal Suleman
Member of the Committee: Prof. Pedro Batista

October 2021

Acknowledgments

A special thank you to Professor Afzal Suleman for the opportunity, guidance, and support to develop this thesis at the Centre for Aerospace Research (CfAR) of the University of Victoria (UVic), British Columbia, Canada.

To the research team at CfAR and my colleagues, a word of gratitude for their collaboration, experience and friendship gained whilst working at the research centre.

To Professor Paulo Oliveira thank you for reviewing this work and for the suggestions provided.

Last but not least, thank you to all the Professors at Instituto Superior Técnico who have over the past 5 years imparted their knowledge and experience in this journey to obtain the MSc. degree in Aerospace Engineering.

Resumo

Esta tese considera o problema de seguimento de caminho coordenado para veículos aéreos não-tripulados (VANT) de asa fixa para uma altitude constante. O problema é parcialmente desacoplado ao decompô-lo primeiro num subproblema de seguimento de caminho através do controlo da atitude de cada VANT, e em segundo no subproblema da coordenação das respetivas distâncias ao longo do caminho através do ajuste das suas velocidades em relação aos seus valores de velocidades desejadas nominais. Relativamente ao subproblema de seguimento de caminho, dois algoritmos serão abordados. O primeiro usa uma decomposição do caminho numa sequência de segmentos de reta e utiliza um controlador baseado na linearização em cada segmento. O segundo utiliza um controlador não-linear para caminhos curvos baseado num campo vetorial, cuja estabilidade é explorada através de técnicas de Lyapunov e estabilidade entrada-para-estado, apresentando-se condições para as quais estabilidade global assintótica do sistema pode ser concluída. Em relação à coordenação, uma lei de consenso proporcional-integral é utilizada, válida para uma topologia de comunicação com conectividade integral. Resultados de simulações que ilustram o funcionamento de ambos os algoritmos de seguimento de caminho coordenado são apresentados e discutidos.

Palavras-chave: seguimento de caminho coordenado, controlo coordenado, seguimento de caminho, veículos aéreos não-tripulados (VANTs), asa-fixa.

Abstract

This thesis addresses the problem of coordinated path-following for fixed-wing Unmanned Aerial Vehicles (UAVs) for a fixed altitude set-point. The problem is partially decoupled by decomposing it first in a path-following sub-problem by commanding the attitude of each UAV, and second in a sub-problem of coordinating their respective along-path parameters by adjusting their speeds about their nominal desired speed profiles. In relation to the path-following problem, two different algorithms are considered. In the first algorithm, the path is decomposed into a concatenation of straight-lines; the controller being linearisation-based. In the second algorithm, a vector field curved path-following controller is considered, whose stability is explored through Lyapunov and input-to-state stability analysis, presenting conditions for which global asymptotic stability of the system may be concluded. With respect to the coordination, a proportional-integral consensus coordination control law is considered, valid for a communication topology with integral connectivity. Simulation results illustrating the functioning of both coordinated path-following algorithms are presented and discussed.

Keywords: coordinated path-following, coordination control, path-following, unmanned aerial vehicles (UAVs), fixed-wing.

Contents

Acknowledgments	iii
Resumo	v
Abstract	vii
Contents	ix
List of Tables	xi
List of Figures	xiii
Nomenclature	xv
1 Introduction	1
1.1 Motivation and Objectives	1
1.2 Literature Review	2
1.2.1 Coordinated Path-Following	3
1.2.2 Path-Following	4
1.3 Contributions	5
2 Path-Following	7
2.1 Control Model	7
2.2 Linearised Path-Following Controller	8
2.2.1 Waypoint Switching	8
2.2.2 Straight-Line Path-Following Controller	9
2.2.3 Stability analysis	10
2.3 Vector Field Curved Path-Following Controller	10
2.3.1 Stability analysis	12
2.3.2 Implementation considerations	16
3 Coordination	17
3.1 Linearised Path-Following Coordination	18
3.2 Vector Field Curved Path-Following Coordination	18
4 Simulation Results	21
4.1 Linearised Path-Following Controller Simulation Results	21
4.2 Vector Field Curved Path-Following Controller Simulation Results	24

4.2.1	Straight-line Paths Simulation Results	25
4.2.2	Detour Simulation Results	27
5	Conclusions	29
5.1	Future Work	30
	Bibliography	31

List of Tables

4.1	Model parameters.	21
4.2	Coordination control parameters (Linearised).	21
4.3	Path-following controller parameters (Linearised).	24
4.4	Initial conditions (Linearised).	24
4.5	Path-following controller parameters (Vector-Field).	24
4.6	Coordination control parameters (Vector-Field).	25
4.7	Initial conditions (Vector-Field).	25

List of Figures

1.1	Implementation architecture.	1
2.1	Diagram illustrating the variables of the control model.	7
2.2	Half-plane waypoint switching diagram.	9
2.3	Diagram of the geometric variables of the straight-line path-following controller.	9
2.4	Diagram of the variables used in the Vector Field Curved Path-Following Controller.	11
2.5	Desired course vector field illustration.	12
2.6	Illustration of the regional character of the input-to state stability of the driven path-following error subsystem.	12
4.1	Communication topology.	21
4.2	Triangle detour simulation results (Linearised).	22
4.3	Polygonal detour simulation results (Linearised).	23
4.4	Straight-line paths simulation (Vector-Field).	26
4.5	Detour path simulation (Vector-Field).	28

Nomenclature

Greek symbols

- δ Lateral distance to the straight-line path.
- η Desired time coordination map. It assigns a virtual time to every point on the path.
- $\kappa(s)$ Curvature of the path at the point parametrized by s .
- ξ Virtual time of the aircraft.
- ϕ Bank angle of the aircraft.
- ϕ_{\max} Maximum roll angle allowed to the aircraft.
- ψ Heading of the aircraft.
- $\psi_q(s)$ Course angle of the path parametrized by s .
- $\tilde{\psi}$ Aircraft's heading relative to the path.
- $\tilde{\psi}_d$ Desired course vector field.
- ω Turn-rate command in the North-East plane.
- ω_{\max} Maximum turn-rate of the aircraft. Function of the airspeed.

Roman symbols

- $\operatorname{arctanh}$ Inverse hyperbolic tangent.
- \mathbf{e}_1 $(1, 0)^T$
- e_χ Course error relative to the path and the desired course at the given cross-track error.
- e_d Cross-track error.
- e_s Along-track error.
- g Local acceleration of gravity.
- \mathbb{I}_n Identity matrix of size $n \in \mathbb{Z}^+$.
- k Gain.

- \mathcal{K} Set of continuous strictly increasing functions $\alpha: [0, a[\rightarrow [0, \infty[$ such that $\alpha(0) = 0$.
- \mathcal{K}_∞ Set of functions of class \mathcal{K} $\alpha: [0, \infty[\rightarrow [0, \infty[$ such that $\lim_{r \rightarrow \infty} \alpha(r) \rightarrow \infty$.
- \mathcal{KL} Set of continuous functions $\beta: [0, a[\times [0, \infty[\rightarrow [0, \infty[$ such that for each fixed s the mapping $r \mapsto \beta(r, s)$ is of class \mathcal{K} , and for each fixed r the mapping $s \mapsto \beta(r, s)$ is decreasing and $\lim_{s \rightarrow \infty} \beta(r, s) = 0$.
- $\mathcal{N}_i(t)$ Set of neighbours of aircraft i in the communication graph, i.e., the aircrafts with which the it is able to communicate, at time t .
- \mathbf{p} North-East position in a local inertial coordinate system.
- q Path parametrization.
- \mathbb{R} Set of real numbers.
- $\mathbf{R}(\psi)$ Rotation matrix from the direction of the speed of the aircraft to the local inertial reference plane.
- \mathbb{R}^+ Set of positive real numbers.
- s Path-length parameter of the path.
- sat_a^b Saturation function between a and b , where $a < b$.
- sech Hyperbolic secant.
- sgn Sign function. $\text{sgn}(x)$ is 1 for positive x , -1 for negative x , and 0 otherwise.
- $u_{\text{coord},i}$ Coordination command.
- V Airspeed.
- $v_{d,i}$ Desired speed profile for aircraft i .
- V_{max} Maximum airspeed.
- V_{min} Minimum airspeed.
- \mathbb{Z}^+ Set of positive integers.

Superscripts

- ' Derivative in relation to the argument.
- \cdot Time derivative.
- T Transpose.

Chapter 1

Introduction

1.1 Motivation and Objectives

Unmanned Aerial Vehicles (UAVs) have been increasingly used for academic, military, and practical applications, such as in reconnaissance, and surveillance missions, due to their decreasing production cost. Nevertheless, their restricted capabilities in terms of power, sensing, communication and computation have generated interest in the use of teams of UAVs to improve the capability of the Unmanned Aerial System (UAS), capacity and flexibility in a variety of applications. Its main advantages vis-à-vis the use of a single UAV are the ability to parallelise individual tasks, increase tolerance to sensor and hardware failures and the possibility of giving different capabilities to each UAV. A large subset of these missions require that the UAVs coordinate their motion in relative and/or absolute time [1]. Examples of these cooperative mission scenarios are sequential auto-landing and coordinated ground-target suppression for multiple UAVs [2].

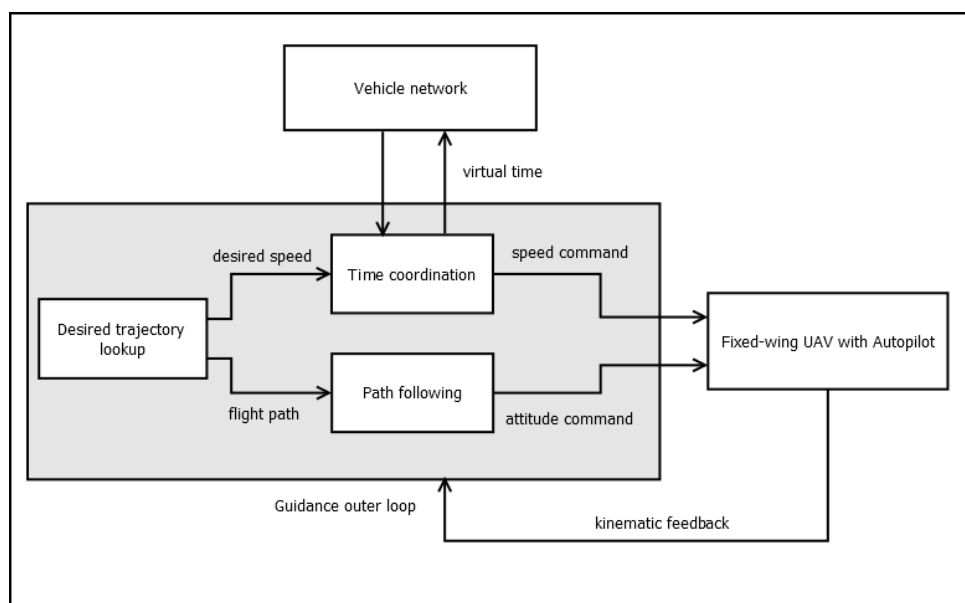


Figure 1.1: System architecture.

This work investigates the problem of coordinated flight of fixed-wing aircraft at a fixed altitude set-point, by formulating it as a coordinated path-following problem, where the aircrafts are required to follow desired spacial paths while coordinating on their positions along them to meet desired spacial and temporal constraints. It is assumed that for each aircraft a desired spacial path and a speed profile have been assigned that solves the required cooperative mission and satisfies the aircrafts' kinematic and dynamic constraints. Following the approach of Kaminer et al. [1] and Xargay et al. [2], the problem is partially decoupled by decomposing it two sub-problems. First in a path-following sub-problem by commanding the attitude of each UAV. Second coordinating their respective along-path parameters by adjustment of their speeds around their desired speed profiles to ensure the aircrafts keep their desired time separation along the path (see fig. 1.1).

The main objectives of this work are to investigate different path-following control laws capable of being integrated into this cooperative path-following formulation, and to investigate the dynamical stability of both the path-following control laws and the overall coordinated path-following.

1.2 Literature Review

Control of multi-agent systems has been receiving increasing attention from various fields in the past years [3]. The control structures used may be classified into either centralized or distributed control. In a centralized control structure, a control centre or host coordinates the information transmission, and the process of task completion. In distributed control, the agents act separately using only local information. In the centralized approach, the control centre is a single point of failure, whereas the distributed approach is more robust to single agent failures.

Information interaction is an important concern in multi-agent systems. It depends on the sensing and communication capabilities of each individual agent, which are subject to a limited range of perception, a limited bandwidth of communication, limited network resources, and other network-induced issues. To model the information exchange among the agents, graph theory is frequently employed by representing the information exchange by either a directed or undirected graph, possibly time-varying.

To coordinate behaviour and cooperate to achieve common goals, research has mainly been focused on the problems of consensus, formation control, and flocking [3].

The consensus problem refers to the problem of the agents reaching agreement on their states through distributed local interactions. Several algorithms exist such as the simple single integrator consensus algorithm [4, 5], the proportional-integral consensus algorithm used in [1, 2], Kalman-consensus [6], passivity-based designs [7–9], and other non-linear algorithms [10, 11]. When the information exchanged is described by a static communication graph, stability may be ensured if the graph contains a rooted directed spanning tree, which, in the case of an undirected graph is equivalent to it being connected (necessary and sufficient condition for the single integrator consensus algorithm). When the communication is time-varying, point-wise satisfaction of the previous condition is usually not required, only the connection of the topology in an integral sense. Detailed analysis of the stability of these algorithms makes frequent use of algebraic graph theory. The reader is referred to Godsil and Royle [12],

particularly chapters 8 and 13.

Formation control consists in driving the agents to maintain and move in a desired geometric pattern in order to perform tasks such as effective search, patrol, and exploration [3]. The main approaches for solving this problem are leader-follower control [3, 13, 14], the virtual leader approach [13], the virtual structure approach [3, 13, 14], and the graph theory method [14].

In leader-follower control there is a leader which follows the assigned trajectory or path, and followers which, based on the information received or sensed from the leader, attempt to maintain and move in their relative place from the leader. This approach may be scaled by using a hierarchy of leaders [15]. The main advantage of this approach is its simplicity, since only the reference trajectory of the leader is required to be specified [13, 16], and the stability of the formation is implied by the stability of the individual vehicle control laws [13]. The main disadvantages are its poor disturbance rejection properties [17]; the over-reliance on the leader for achieving the coordination [3, 13, 16], which may be undesirable in terms of processing and communication requirements, and fault tolerance [16]; and the inexistence of explicit feedback from the leader to the followers [16].

In the virtual leader approach [13], vehicles in the formation synthesize a single, possibly fictitious, leader vehicle whose trajectory acts as a leader for the group. This avoids the problem of disturbance rejection inherent in the leader-follower approach at the expense of high communication and computation capabilities needed to synthesize the virtual leader and communicate its position for the real-time control of the other vehicles.

In the virtual structure approach [16], the entire formation is treated as a single structure. The desired dynamics of the virtual structure is first defined, then the motion of the virtual structure is translated into the desired motion for each agent. Lastly, the tracking controls for each spacecraft are derived. The advantages of the virtual structure approach are the ease of prescription of the coordinated behaviour for the group, and the natural definition of the feedback to the virtual structure. The main disadvantage is the limitation of the class of potential applications due to the virtual structure constraint.

Flocking (also referred to as Swarming) is the problem of achieving certain desired group behaviours by means of local interaction and behavioural rules between the agents [3]. It draws inspiration from the self-organizing behaviour of groups of animals such as fish schools, and bee and bird swarms [3].

In behaviour based control, basic control behaviours of the agents are defined and weighed to obtain the final formation control inputs for the group. The advantages are its ease in deriving control strategies when the agents have multiple competing objectives, the explicit feedback to the formation (since each agent reacts according to the position of its neighbours), and the natural decentralized implementation [16]. The main disadvantages are the difficulties in defining the behaviour of the group [3, 16], and the analysis of the characteristics of the formation [16].

1.2.1 Coordinated Path-Following

Having reviewed the general topic of multi-agent control, consideration is now given to the specific topic explored in this work. As previously stated, cooperative path-following consists in coordinating each

agent the agents to meet desired temporal constraints, while each follows its assigned desired spacial path. It essentially extends a path-following controller by adding a consensus coordination mechanism to the system. There are two slightly different approaches in the literature. The first assigns both a desired spacial path and speed profile to each agent [1, 2, 18]. The second assigns only a desired spacial path to each agent, whereas the temporal information is implied by the parametrization used [15]. Both variants are equivalent, since the latter has an implied unitary desired speed profile for the evolution of its parameter. Using a change of variables in the parametrization allows both to be converted between each other. The most interesting papers identified on the subject are the following.

Ihle, Arcak, and Fossen [9] develop a passivity approach for coordinate path-following of general non-linear systems by combining a passive path-following control algorithm for each UAV with a passive synchronization algorithm.

Ghabcheloo et al. [18] studied the coordinated path-following, also for non-linear systems, under communication losses and time delays, and derived sufficient conditions for the stability of the system.

Kaminer et al. [1] and Xargay et al. [2] extend and apply this last study to address the problem of steering a fleet of fixed-wing UAVs along desired paths while meeting spatial and temporal constraints. A 3-D path-following control law is used, but the maximum angular rate constraints of each aircraft are not considered.

In Chen et al. [15], a coordinated path-following approach is proposed for group leaders. A virtual target along the desired path is assigned for each leader UAV, whose updating law ensures their coordination. The leader UAVs then track their respective virtual target. The maximum turn-rate constraint is taken into account, but only flight at a constant altitude is considered.

Chen et al. [19] presents a hybrid path-following control law for a fixed-altitude set-point. This hybrid approach defines an invariant coordination set for each aircraft in which the aircraft coordinates with the others. Outside this coordination set, the aircraft tracks its own path independently until it enters the coordination set. It considers both the velocity and turn-rate constraints.

In Wang, Wang, and Zhu [20], a control law for the more general problem of cooperative moving path-following control is provided, with consideration for speed constraints and collision avoidance. It also only considers flight at a constant altitude.

Applications of coordinated path-following control include cooperative road search and sequential auto-landing [1, 2], and formation reconfiguration [15].

1.2.2 Path-Following

The path-following problem consists in designing a control law that forces the output of the system to follow a desired path, with the secondary goal of obeying a desired speed assignment [21]. Vis-à-vis trajectory tracking, it offers the flexibility to shape the transient behaviour of the path parameter, and is less sensitive to disturbances [21, 22]. It is an integral part in solving the coordinated path-following problem and there are several solutions for fixed-wing aircraft proposed in the literature that could be used.

Hauser and Hindman [23] present a method of transforming a trajectory tracking controller to a path-following controller by using the projection of the trajectory as the reference, and using a simple control law for the path parameter to follow the projected point.

In Sujit, Saripalli, and Sousa [24], a survey of basic path-following algorithms for fixed-wing aircraft is presented. It consists entirely of path-following controllers for straight-lines and circular orbits, mostly for level-flight.

Beard and Humpherys [22] derive a path-following control law for following straight lines and orbits using a vector field and the theory of nested saturations that explicitly account for roll angle and flight path constraints. Besides constant altitude manoeuvres, climb manoeuvres are also considered.

Zhao et al. [25] extend the vector field approach to general curved paths. It uses the concepts of input-to-state stability and the result on the stability of cascade connected input-to-state stable system to try to conclude global asymptotic stability. No constraints on the turn-rate are considered.

There are some more recent results for following general 3-dimensional curved paths [26–28].

As the effect of wind is as important factor in fixed-wing flight, some of the methods above adapt their control laws by means of the wind triangle to use ground-speed and course controls in place of airspeed and heading, but do not analyse the effects of the wind estimator on the stability of the overall system. The following papers address this issue.

Brezoescu et al. [29] estimate the parameters of unknown wind disturbances with the use of an adaptive back-stepping control law for path-following of straight lines.

Liu, McAree, and Chen [30] develops a non-linear disturbance observer to estimate and compensate undesirable wind in the Serret-Frenet frame.

Yang et al. [27] develops a disturbance observer-based method by means of higher order sliding mode to provide velocity and acceleration estimates of the wind disturbances.

The path-following strategies that are able to follow only straight-line and/or circular paths, a concatenation of several segments may be used to produce more complex paths [31]. Then, a switching strategy between consecutive segments must be chosen. The strategies presented in the literature are ball waypoint switching, half-plane waypoint switching, fillet waypoint transition, and Dubins paths waypoint transition [31].

1.3 Contributions

The subsequent chapters in this work describe and analyse the proposed algorithms to solve each of the sub-problems produced by the decomposition of the coordinated path-following considered at the end of section 1.1. The algorithms are implemented in a simulation environment and their results discussed.

Chapter 2 focuses on the path-following sub-problem. The control model used for path-following control law design is first described in section 2.1, followed by the presentation of the two alternative path-following algorithms considered in the work.

In section 2.2, the first path-following algorithm is presented, a linearised path-following controller

able to follow paths composed of sequences of straight-line segments. The simple half-plane switching method is chosen to perform the switching between segments (subsection 2.2.1), and the controller for each segment is synthesized (subsection 2.2.2). The section is concluded by describing the conditions required for local asymptotic stability (subsection 2.2.3).

In section 2.3, the second algorithm is presented, a non-linear vector-field path-following controller capable of following general curved paths. The stability analysis performed in Zhao et al. [25] was found to be incomplete and has been corrected and extended to account for time-varying speeds (subsection 2.3.1), presenting conditions for which global asymptotic stability of the system may be concluded.

In chapter 3, the coordination problem is described, and the chosen coordination algorithm presented. The integration of this algorithm with the path-following algorithms presented in chapter 2 is described in detail in sections 3.1 and 3.2. The stability of the second cooperative path-following algorithm is also discussed in section 3.2.

Chapter 4 presents the obtained simulation results of the implementation of the algorithms in a Simulink simulation environment. Based on the results, the control behaviour and performance of the algorithms is analysed and compared.

The work is concluded in chapter 5 by summarizing the objectives set and the results achieved. It also includes a roadmap for the extension of the work undertaken and suggestions for further investigation.

Chapter 2

Path-Following

2.1 Control Model

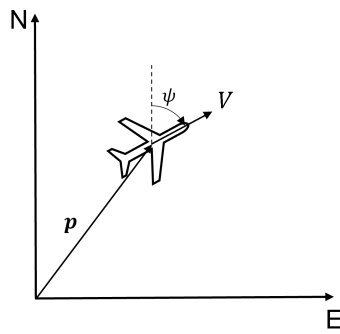


Figure 2.1: Diagram illustrating the variables of the control model.

In the following, it is assumed that each aircraft is equipped with an autopilot capable of maintaining the desired altitude set-point, and of controlling it in a coordinated turn. In order to simplify the analysis of the following path-following control algorithms, it is also assumed the aircraft flies in no wind. Under these assumptions, the kinematic model for the aircraft is

$$\dot{\mathbf{p}} = \mathbf{R}(\psi)V\mathbf{e}_1 \quad (2.1)$$

$$\dot{\psi} = \frac{g}{V} \tan \phi, \quad (2.2)$$

where $\mathbf{p} = (x, y)^T$ is the North-East position of the aircraft in a local inertial reference frame, ψ is the heading of the aircraft,

$$\mathbf{R}(\psi) = \begin{pmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{pmatrix} \quad (2.3)$$

is the rotation matrix from the direction of the speed of the aircraft to the local inertial reference frame, V is the airspeed of the aircraft, $\mathbf{e}_1 = (1, 0)^T$, g is the local acceleration of gravity, ϕ is the roll angle of the aircraft. The airspeed and the roll angle are considered to be the inputs, since it is assumed that the

autopilot controls these aircraft variables on a much faster time-scale than the path-following algorithms that will be presented. The geometric variables may be visualized in figure 2.1.

The roll angle input is transformed to a turn-rate command by

$$\omega^c = \frac{g}{V} \tan \phi. \quad (2.4)$$

Leading to the model that will be considered for the path-following algorithms to be presented

$$\dot{\mathbf{p}} = \mathbf{R}(\psi)V\mathbf{e}_1 \quad (2.5)$$

$$\dot{\psi} = \omega^c, \quad (2.6)$$

subject to the kinematic input constraints

$$0 < V_{\min} \leq V \leq V_{\max}, \quad (2.7)$$

$$|\omega^c| \leq \omega_{\max}(V). \quad (2.8)$$

These constraints are imposed by the flight characteristics of fixed-wing UAVs. In the first constraint, the lower bound is due to the existence of a minimum airspeed under which the aircraft is not capable of maintaining the level flight, the stall speed, and the upper bound is due to the aircrafts maximum thrust capability. The second constraint is imposed by the maximum roll angle ϕ_{\max} that it is to be allowed to the aircraft, where

$$\omega_{\max}(V) = \frac{g}{V} \tan \phi_{\max}. \quad (2.9)$$

2.2 Linearised Path-Following Controller

In this method, the desired path is first approximated by a concatenation of straight-line segments, whose end-points will be denoted by $\{\mathbf{w}_0, \mathbf{w}_1, \dots, \mathbf{w}_N\}$, and with constant desired speed profiles on each segment. First, the segment switching method is described in subsection 2.2.1, followed, in subsection 2.2.2, by the straight-line straight-line path-following controller used for each segment.

2.2.1 Waypoint Switching

To transition between the different straight-line path segments, the half-plane switching method [31] is chosen, illustrated in figure 2.2. In this method, the path controller switches segments when it has crossed the bisecting plane between the current segment being followed and the segment to be followed next. This switching condition is satisfied when the aircraft's position is within the half-plane

$$\mathcal{H} = \{\mathbf{p} \in \mathbb{R}^2 : (\mathbf{p} - \mathbf{w}_i)^T \mathbf{n}_i \geq 0\}, \quad (2.10)$$

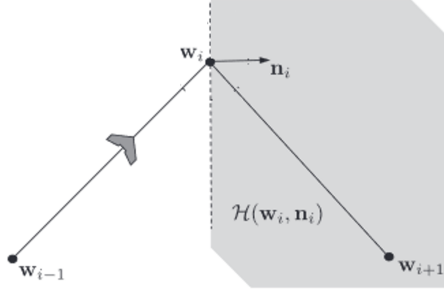


Figure 2.2: Half-plane waypoint switching diagram (src: [31]).

where $\overline{w_{i-1}w_i}$ is the current segment, $\overline{w_iw_{i+1}}$ is the next segment, the unit vectors of each segment are given by

$$\mathbf{q}_i = \frac{\mathbf{w}_{i+1} - \mathbf{w}_i}{\|\mathbf{w}_{i+1} - \mathbf{w}_i\|}, \quad (2.11)$$

and a normal to the half-plane that points to the inside is given by

$$\mathbf{n}_i = \mathbf{q}_{i-1} + \mathbf{q}_i. \quad (2.12)$$

2.2.2 Straight-Line Path-Following Controller

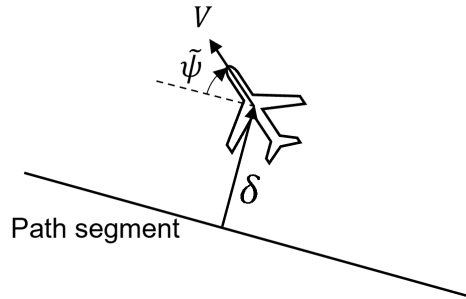


Figure 2.3: Diagram of the geometric variables of the straight-line path-following controller.

The straight-line path-following controller to be used in each segment is now designed. According to the control model considered in section 2.1, the evolution of the lateral distance δ to straight-line path, considered positive when the aircraft is to the right of the path, is described by

$$\dot{\delta} = V \sin \tilde{\psi}, \quad (2.13)$$

where $\tilde{\psi} = \psi - \Psi_0 \in] -\pi, \pi]$ is the aircraft's heading relative to the heading of the current straight-line path Ψ_0 (cf. figure 2.3).

Linearising this last equation around the path heading for a given airspeed V , combining with equation (2.6), and introducing the integral controller state $\dot{\delta}_I = \delta + \delta_{aw}$ to ensure asymptotic convergence to

the path, produces the linearised system

$$\begin{pmatrix} \dot{\delta}_I \\ \dot{\delta} \\ \dot{\tilde{\psi}} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & V \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \delta_I \\ \delta \\ \tilde{\psi} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega^c + \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \delta_{aw}. \quad (2.14)$$

This system is controllable, so the following PID control law is chosen

$$\omega_\ell^c = -k_{\delta_I} \delta_I - k_\delta \delta - k_{\tilde{\psi}} \tilde{\psi}. \quad (2.15)$$

In order to satisfy the turn rate constraint (2.8), the control is saturated to ω_{\max}

$$\omega^c = \text{sat}_{-\omega_{\max}}^{\omega_{\max}} \omega_\ell^c, \quad (2.16)$$

and the back-calculation anti-wind-up solution is implemented by discharging the integral state by the amount

$$\delta_{aw} = k_{\delta_I, aw} (\omega_\ell^c - \omega^c). \quad (2.17)$$

2.2.3 Stability analysis

Linearising the closed-loop path-following system for a given constant speed V , the characteristic polynomial is

$$\Delta(s) = s^3 + k_{\tilde{\psi}} s^2 + V k_\delta s + V k_{\delta_I}. \quad (2.18)$$

By the Ruth-Hurwitz stability criterion, the linearised system is stable for

$$k_{\tilde{\psi}} > 0, \quad k_\delta k_{\tilde{\psi}} > k_{\delta_I}, \quad k_{\delta_I} > 0. \quad (2.19)$$

Hence, by Lyapunov's indirect method [32, Theorem 4.7], the non-linear closed-loop path-following system is locally asymptotically stable for a given speed, and this choice of gains.

2.3 Vector Field Curved Path-Following Controller

In this method, a non-linear curved path-following controller [25] based on a desired vector field heading is considered. The geometric variables considered are represented in figure 2.4, where $q: I \subset \mathbb{R} \rightarrow \mathbb{R}^2$ is the parametrization of the path of a given UAV with sectionally continuous second derivative, parametrized by the path-length variable s . A virtual point is introduced on the path, where s is the path-length parameter of the virtual point on the path and $q(s)$ is its position. Define, respectively, the along-track and the cross-track errors as

$$\begin{pmatrix} e_s \\ e_d \end{pmatrix} = R^T(\psi_q)(p - q(s)), \quad (2.20)$$

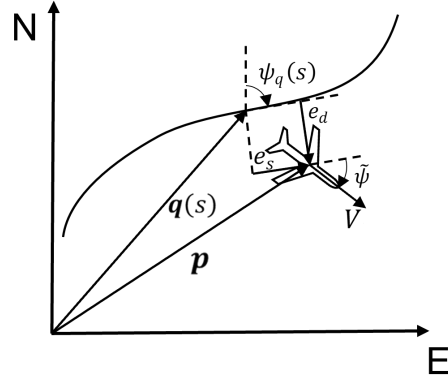


Figure 2.4: Diagram of the variables used in the Vector Field Curved Path-Following Controller.

and the course error as

$$e_\chi = \psi - \psi_q - \tilde{\psi}_d, \quad (2.21)$$

where ψ_q is the course angle of the path in the point parametrized by s , which may be calculated as

$$\psi_q(s) = \arctan(q'_2(s), q'_1(s)). \quad (2.22)$$

The error system is then described as [25]

$$\dot{e}_s = V \cos(\tilde{\psi}_d(e_d) + e_\chi) - (1 - \kappa(s)e_d)\dot{s}, \quad (2.23)$$

$$\dot{e}_d = V \sin(\tilde{\psi}_d(e_d) + e_\chi) - \kappa(s)e_s\dot{s}, \quad (2.24)$$

$$\dot{e}_\chi = \omega^c - \kappa(s)\dot{s} - \dot{\tilde{\psi}}_d. \quad (2.25)$$

where $\kappa(s)$ is the curvature of the path at the point parametrized by s , which may be calculated by¹

$$\kappa(s) = \frac{q'_1(s)q''_2(s) - q'_2(s)q''_1(s)}{\|q'(s)\|^3}, \quad (2.26)$$

and $\tilde{\psi} = \psi - \psi_q$ is the relative course of the aircraft with regard to the course angle of the path.

The system is a cascade interconnection of the driving subsystem (2.25) and the driven subsystem (2.23)–(2.24).

The following updating and control law are chosen as follows [25]

$$\dot{s} = k_s e_s + V \cos \tilde{\psi}, \quad (2.27)$$

$$\omega = -k_\omega e_\chi + \kappa(s)\dot{s} + \dot{\tilde{\psi}}_d, \quad (2.28)$$

for a positive control gains k_s , and k_ω , and a desired course differentiable vector field (see example figure 2.5)

$$\tilde{\psi}_d(e_d) = -\tilde{\psi}^\infty \tanh(k_d e_d), \quad (2.29)$$

¹Formula valid for any parametrization.

where for a positive control gain k_d , and far-away desired course $\tilde{\psi}^\infty \in]0, \frac{\pi}{2}[$.

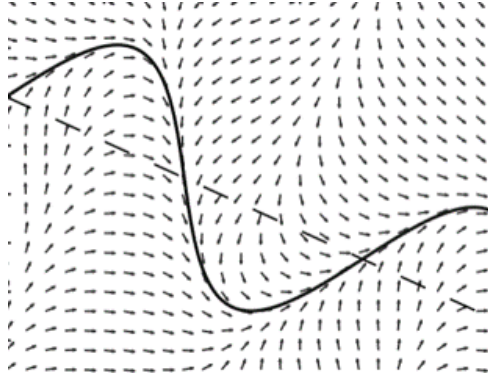


Figure 2.5: Desired course vector field illustration (src: [33]).

The use of a dynamic update law (2.27) for the virtual point on path, as opposed to choosing it to be the closest point on the path, avoids singularities when there is no single closest point [1].

2.3.1 Stability analysis

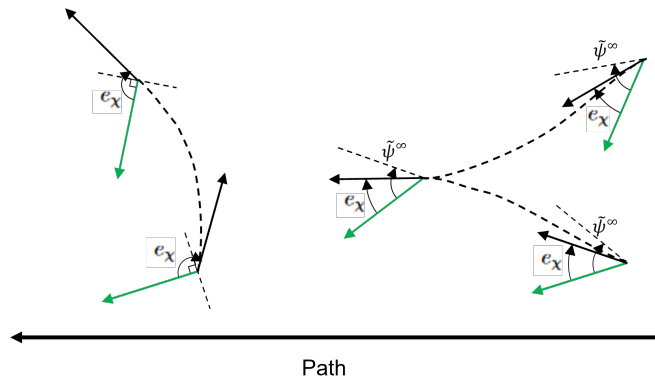


Figure 2.6: Illustration of the regional character of the input-to state stability of the driven path-following error subsystem (2.23)–(2.24). The system becomes unbounded for a bounded course error with $\frac{\pi}{2} < e_\chi < \pi$ (left), and bounded for $|e_\chi| < \tilde{\psi}^\infty$ (right). In green is the desired course at the point, and in black is the actual course.

Contrary to what is claimed in Zhao et al. [25], the driven subsystem (2.23)–(2.24) is not input-to-state stable. To see this, consider the case of following a straight-line, which is pictured in figure 2.6 on the left. Consider a bounded course-error input that satisfies the bounds

$$\frac{\pi}{2} < e_\chi(t) < \pi, \quad \text{for all } t \geq t_0, \quad (2.30)$$

and an initial state with $e_d > 0$. By equation (2.24),

$$\dot{e}_d = V \sin(\tilde{\psi}_d + e_\chi), \quad (2.31)$$

since a straight-line has no curvature. For a positive cross-error, $-\pi/2 < \tilde{\psi}_d < 0$, which implies that

$0 < \tilde{\psi}_d + e_\chi(t) < \pi$, hence the sine is always positive. Thus, $\dot{e}_d > 0$ for all $t \geq t_0$, so the cross-error grows unbounded. In conclusion, the system cannot be input-to-state stable, since there are initial conditions that produce an unbounded state for a bounded input.

However, as suggested by the right side of figure 2.6, for smaller course errors the cross-track error is seen to be bounded. Thus, if the course error is smaller than the far-away desired course, there is a cross-track error for which the course is parallel to the path. For larger cross-track errors the course is directed towards the path, whereas for smaller course errors it is directed away from the path. If the course error has the opposite direction of the one depicted, the aircraft will cross to the other side of the path, and the situation will be the same as the previous one, since it will be the mirror of the one depicted. The following regional input-to-state stability property generalizes this line of reasoning for generic curved paths.

Proposition 1. *Considering V as a time-varying signal satisfying the bounds in equation (2.7). Then, system (2.23)–(2.24) along with the updating law (2.27) with state (e_s, e_d) , and input e_χ is regionally input-to-state stable with respect to $\mathbb{R}^2 \times B_r$, with $r = \theta\tilde{\psi}^\infty$, for some $0 < \theta \leq \frac{\pi}{2\tilde{\psi}^\infty} - 1$.*

Proof. The following proof is based on [33, sec. III. D], where the author considers a similar desired course vector field². The main difference being that it uses the geometrically closest point on the path for the determination of the cross-track error. Here the argument is extended for the case being considered where a dynamic update law (2.27) is used, with the consequent addition of the along-track error and its dynamics (2.23). In addition, explicit account of the regional character of the input-to-state stability is made, which was absent in the referenced work.

Let

$$V_1 = \frac{1}{2}(e_s^2 + e_d^2). \quad (2.32)$$

Taking the time derivative,

$$\dot{V}_1 = -k_s e_s^2 + V e_d \sin(\tilde{\psi}_d(e_d) + e_\chi) \quad (2.33)$$

$$= -k_s(1 - \alpha)e_s^2 + V e_d \sin((1 - \theta)\tilde{\psi}_d(e_d)) \quad (2.34)$$

$$\begin{aligned} & - \alpha k_s e_s^2 \\ & + V e_d (\sin(\tilde{\psi}_d(e_d) + e_\chi) - \sin((1 - \theta)\tilde{\psi}_d(e_d))), \end{aligned}$$

for some $\alpha, \theta \in]0, 1[$.

It will be shown that

$$\dot{V}_1 \leq -k_s(1 - \alpha)e_s^2 + V_{\min} e_d \sin((1 - \theta)\tilde{\psi}_d(e_d)), \quad (2.35)$$

for $\|(e_s, e_d)\| \geq \rho(|e_\chi|)$, for a class \mathcal{K} function ρ .

The sine function is strictly increasing in the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$. In this interval,

$$|e_\chi| \leq \theta\tilde{\psi}^\infty \tanh(k_d |e_d|) \quad (2.36)$$

²Instead the vector field being constructed from a hyperbolic tangent, an arc tangent, multiplied by $\frac{2}{\pi}$ to preserve the interpretation of $\tilde{\psi}^\infty$, is used.

ensures that the last term in equation (2.34) is non-positive. Choosing

$$\theta \leq \frac{\pi}{2\tilde{\psi}^\infty} - 1, \quad (2.37)$$

guaranties that the arguments of the sine lie in the interval, since

$$|\tilde{\psi}_d(e_d) - e_\chi| \leq (1 + \theta)\tilde{\psi}^\infty. \quad (2.38)$$

Inverting inequality (2.36),

$$|e_d| \geq \frac{1}{k_d} \operatorname{arctanh} \left(\frac{|e_\chi|}{\theta\tilde{\psi}^\infty} \right). \quad (2.39)$$

Now assume this last inequality does not hold. Noting that

$$|\sin(\tilde{\psi}_d(e_d) + e_\chi) - \sin((1 - \theta)\tilde{\psi}_d(e_d))| \leq 2|e_\chi| + 2, \quad (2.40)$$

the last term of equation (2.34) is less than or equal to

$$\frac{2V}{k_d} \operatorname{arctanh} \left(\frac{r}{\theta\tilde{\psi}^\infty} \right) (|e_d| + 1). \quad (2.41)$$

Thus, if

$$|e_s| \geq \sqrt{\frac{2V}{\alpha k_s k_d} \operatorname{arctanh} \left(\frac{|e_\chi|}{\theta\tilde{\psi}^\infty} \right) (|e_\chi| + 1)}, \quad (2.42)$$

then inequality (2.35) is verified.

Noting the bounds (2.7), defining the class \mathcal{K} function as

$$\begin{aligned} \rho(r) = & \frac{1}{k_d} \operatorname{arctanh} \left(\frac{r}{\theta\tilde{\psi}^\infty} \right) \\ & + \sqrt{\frac{2V_{\max}}{\alpha k_s k_d} \operatorname{arctanh} \left(\frac{r}{\theta\tilde{\psi}^\infty} \right) (r + 1)}, \end{aligned} \quad (2.43)$$

achieves the desired result. \square

Proposition 2. *System (2.25) with the turn rate command control law (2.28) is globally asymptotically stable, if*

$$|\kappa(s)\dot{s} + \dot{\tilde{\psi}}_d| < \omega_{\max}(V_{\max}). \quad (2.44)$$

Proof. Let

$$V_\chi = \frac{1}{2} e_\chi^2. \quad (2.45)$$

Its time derivative subject to control law (2.28) is

$$\dot{V}_\chi = -k_\omega e_\chi^2, \quad (2.46)$$

if the turn rate is not saturated.

If it is saturated, then it is equal to

$$\begin{aligned}\dot{V}_\chi &= e_\chi(-\omega_{\max} \operatorname{sgn}(e_\chi) - \kappa(s)\dot{s} - \dot{\tilde{\psi}}_d) \\ &< -\omega_{\max}|e_\chi| + |e_\chi|\omega_{\max} = 0.\end{aligned}\quad (2.47)$$

Hence [32, Theorem 4.9], the system is globally asymptotically stable. \square

Proposition 3. *System (2.23)–(2.25) with control laws (2.27)–(2.28) is globally uniformly asymptotically stable if condition (2.44) is always satisfied.*

Proof. System (2.23)–(2.25) is a cascade connection between an input-to-state stable for all states and a subset of the input and a globally asymptotically stable system. Thus, an argument similar to Khalil [32, Lemma 4.7] shows uniform asymptotic stability for any state with $|e_\chi| < \theta\tilde{\psi}^\infty$.

For $|e_\chi| \geq \theta\tilde{\psi}^\infty$, since system (2.25) with control law (2.28) is uniformly asymptotically stable, there is a T such that $|e_\chi(T)| < \theta\tilde{\psi}^\infty$. The solution of system (2.23)–(2.24) is defined in this interval, so the system progresses to state $(e_s(T), e_d(T), e_\chi(T))$ applying the last argument to this as the initial state yields the desired result. \square

The last remaining issue is to determine the conditions under which condition (2.44) is guaranteed to be satisfied. For $e_s = 0$,

$$|\kappa(s)\dot{s} + \dot{\tilde{\psi}}_d| = |\kappa(s)V \cos \tilde{\psi} + V \sin \tilde{\psi} \tilde{\psi}'_d(e_d)| \quad (2.48)$$

$$\leq V(|\kappa(s)| + \tilde{\psi}^\infty k_d). \quad (2.49)$$

Thus, it is sufficient for

$$|\kappa(s)| < \frac{\omega_{\max}(V_{\max})}{V_{\max}} - \tilde{\psi}^\infty k_d. \quad (2.50)$$

Considering now the case of a non-zero along-track error,

$$\kappa(s)\dot{s} + \dot{\tilde{\psi}}_d = -\kappa(s)\tilde{\psi}_d k_s e_s^2 + \kappa(s)(k_s - \tilde{\psi}_d(e_d)V \cos \tilde{\psi})e_s + V(\kappa(s) \cos \tilde{\psi} + \tilde{\psi}'_d(e_d) \sin \tilde{\psi}), \quad (2.51)$$

which is quadratic on along-track error. With the constraint on the curvature (2.50) being satisfied, there is a $d_s > 0$ in which condition (2.44) is satisfied for $|e_s| < d_s$. This constant may be calculated by solving the quadratic equation where it intersects $\omega_{\max}(V_{\max})$ and $-\omega_{\max}(V_{\max})$, and taking into account the bounds (2.7),

$$|\tilde{\psi}_d(e_d)| \leq \tilde{\psi}^\infty, \quad \text{and} \quad (2.52)$$

$$|\tilde{\psi}'_d(e_d)| \leq \tilde{\psi}^\infty k_d. \quad (2.53)$$

The constant may be taken as the minimum of the absolute value of these solutions.

Although with this argument a condition on the curvature has been established that guarantees the satisfaction of bound (2.44) for a suitably small along-track error, proposition 3 leaves open the possibility

of the along-track error growing, which could result in the bound stopping being satisfied.

2.3.2 Implementation considerations

To simplify the theoretical development it was assumed that the path q is parametrized by a path-length parameter s , i.e., for all $s \in I \subset \mathbb{R}$

$$\|q'(s)\| = 1. \quad (2.54)$$

In an application, it might be inconvenient to re-parametrize the path to ensure this condition is satisfied. Thus, for a general path parametrized by s , now not necessarily a path-length parameter, the only change that needs to be made is to replace equation (2.27) by

$$\dot{s} = \frac{1}{\|q'(s)\|} (k_s e_S + V \cos \tilde{\psi}). \quad (2.55)$$

Chapter 3

Coordination

The desired path-length of the aircraft along its assigned path for time t_d is given by [1]

$$\ell_{d,i}(t_d) = \int_0^{t_d} v_{d,i}(\tau) d\tau, \quad (3.1)$$

where $v_{d,i}(\tau)$ is the desired speed of aircraft i at time τ . Since the desired speed of an aircraft must always be strictly positive, this defines a strictly increasing function of the desired time t_d , so it is invertible [1]. This inverse function η gives the desired time that the aircraft should be at the given length of the path, and shall henceforth be referred to as the aircraft's virtual time.

The dynamics of the virtual times are given by

$$\dot{\xi}_i = \frac{\dot{\ell}_i}{v_{d,i}(\xi_i)}. \quad (3.2)$$

For each of the path-following algorithms (sections 3.1 and 3.2), this equation is transformed to the single integrator dynamics

$$\dot{\xi}_i = u_{\text{coord},i}. \quad (3.3)$$

The coordination control law implemented for each the leaders i is [1]

$$u_{\text{coord},i} = -k_P \sum_{j \in \mathcal{N}_i(t)} (\xi_i - \xi_j) + 1, \quad (3.4)$$

while the one implemented for each follower i is [1]

$$\begin{aligned} u_{\text{coord},i} &= -k_P \sum_{j \in \mathcal{N}_i(t)} (\xi_i - \xi_j) + \chi_{I,i}, \\ \dot{\chi}_{I,i} &= -k_I \sum_{j \in \mathcal{N}_i(t)} (\xi_i - \xi_j) + k_{\text{coord,aw}}(V - V_{\text{unsat}}), \quad \chi_{I,i}(0) = 1, \end{aligned} \quad (3.5)$$

where the set of communication neighbours of i at time t $\mathcal{N}_i(t)$ is the set of aircraft with which i is able to communicate (where the communication is assumed to be bidirectional), and k_P and k_I are positive coordination gains. The introduction of the integral term provides disturbance rejection capabilities [34].

The second term is a back-calculation anti-wind-up term, with $k_{\text{coord,aw}}$ being a positive gain. This is done, since the speed commands are saturated between the bounds in equation (2.7) [1].

It is assumed that the communication between each UAV is bidirectional and that the information is transmitted continuously with no delays, each vehicle only exchanges its coordination state $\xi_i(t)$ with its neighbours, and the communications graph is connected in an integral sense, i.e., it satisfies the condition

$$\frac{1}{N} \frac{1}{T} \int_t^{t+T} QL(\tau)Q^T d\tau \geq \mu \mathbb{1}_{N-1}, \quad \text{for all } t \geq 0, \quad (3.6)$$

where $L(t) \in \mathbb{R}^{N \times N}$ is the Laplacian of the communications graph, Q an $(N-1) \times N$ matrix such that $Q\mathbf{1}_N = 0$ and $QQ^T = \mathbb{1}_{N-1}$, $\mathbf{1}_N$ is the vector in \mathbb{R}^n whose components are all 1, and the parameters $T, \mu > 0$ represent the level of connectivity of the communications graph [2].

3.1 Linearised Path-Following Coordination

For this simple algorithm, the speed profile is assumed to be constant along each path segment. As such, equation 3.1 may be expressed as

$$\eta(s) = \frac{\Delta \ell_{i,j}}{v_{d,i,j}} + \sum_{j \in \text{previous segments}} \frac{L_{i,j}}{v_{d,i,j}}, \quad (3.7)$$

where $L_{i,j}$ is the total length of path segment j of aircraft i , $v_{d,i,j}$ is the desired speed at path segment j of aircraft i , and $\Delta \ell_{i,j}$ is the length the aircraft has covered in the current path segment, given by

$$\Delta \ell_{i,j} = (\cos \Psi_0, \sin \Psi_0)(\mathbf{p} - \mathbf{w}_{i-1}). \quad (3.8)$$

For the considered case of straight-line path-following, the dynamics of the path length are

$$\dot{\ell}_i = V_i \cos \tilde{\psi}_i. \quad (3.9)$$

The commanded velocity is transformed by

$$V_{\ell,i} = \frac{v_{d,i}}{\cos \tilde{\psi}_i} u_{\text{coord},i}, \quad (3.10)$$

valid for $\tilde{\psi}_i \in]-\frac{\pi}{2}, \frac{\pi}{2}[$, which under linearisation achieves the single integrator dynamics (3.3).

3.2 Vector Field Curved Path-Following Coordination

Consider now the vector field curved path-following algorithm. Contrary to the previous algorithm, for this one additional restrictions are placed on the coordination map. Substituting equation (2.27) into equation (3.2), yields

$$\dot{\xi}_i = \frac{k_s e_{s,i} + V_i \cos \tilde{\psi}_i}{v_{d,i}(\xi_i)}. \quad (3.11)$$

Performing feedback linearisation by setting the commanded airspeed to

$$V_i = \frac{-k_s e_{s,i} + v_{d,i} u_{\text{coord},i}}{\cos \tilde{\psi}_i}, \quad (3.12)$$

valid for $\tilde{\psi}_i$ in the coordination region $\tilde{\psi}_i \in]-\frac{\pi}{2}, \frac{\pi}{2}[$, transforms it to the form of equation (3.3). The speed is also saturated between the bounds in constraint (2.7) to ensure its satisfaction.

In the case where $\tilde{\psi}_i$ is not in the coordination region, the aircraft is not able to effectively coordinate its position along the path with the others by changing its speed. The aircraft should maintain its speed while it turns until the relative course to the path enters the coordination region. From the result in proposition 2, this will happen in finite time.

Once all the aircraft have entered the coordination region, the path-following errors will converge to zero, since the results established in the previous chapter are valid for any speeds satisfying constraint (2.7). In addition, the coordination states will converge since system (3.3) with coordination control laws (3.4) and (3.5), transformed by (3.12), with the communication topology satisfying condition (3.6) is input-to-state stable with the velocity tracking error considered as the input [2, Lemma 3][1, Lemma 4.2].

Chapter 4

Simulation Results

Table 4.1: Model parameters.

g	V_{\min}	V_{\max}	ϕ_{\max}
9.81 m s^{-2}	15 m s^{-1}	44 m s^{-1}	40°

In this chapter, simulations of the working of the algorithms explained in chapters 2 and 3 are presented. The simulations were performed in Simulink using the model described in section 2.1, with the parameters in table 4.1.

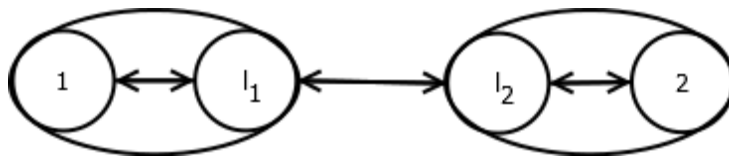


Figure 4.1: Communication topology. 1 and 2 represent the physical vehicles, while l_1 and l_2 are the virtual leaders.

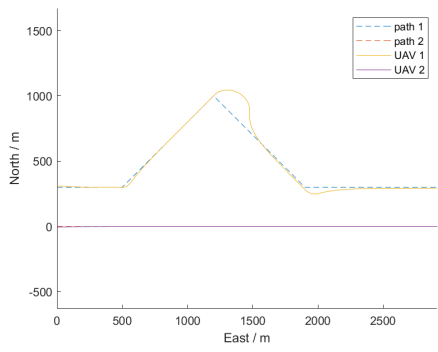
For the simulations, the chosen communication topology is presented in figure 4.1. In this topology, each aircraft implements its own virtual leader, with each aircraft communicating its actual virtual time only with its own virtual leader, and the virtual leaders communicating their virtual times between themselves. This addition of virtual leaders with “uncertainty-free dynamics” improves coordination robustness in the presence of disturbances like winds and gusts [1, 35].

4.1 Linearised Path-Following Controller Simulation Results

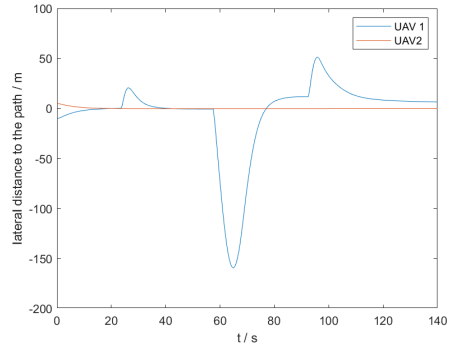
Table 4.2: Coordination control parameters (Linearised).

k_P	k_I	$k_{\text{coord,aw}}$
0.9 s^{-1}	0.03 s^{-2}	$0.02 \text{ m}^{-1} \text{ s}^2$

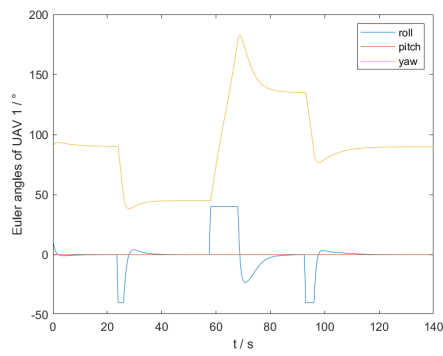
In this section, the results of two simulation scenarios using the linearised path-following controller are shown in figures 4.2 and 4.3, while tables 4.2, 4.3, and 4.4. In either of the scenarios, the mission of



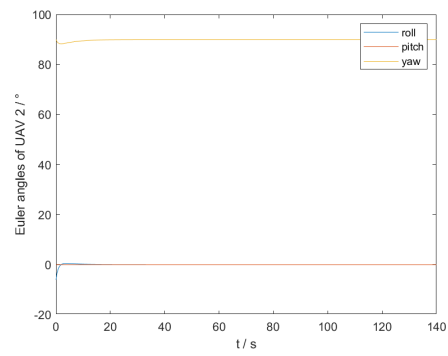
(a) UAVs trajectories.



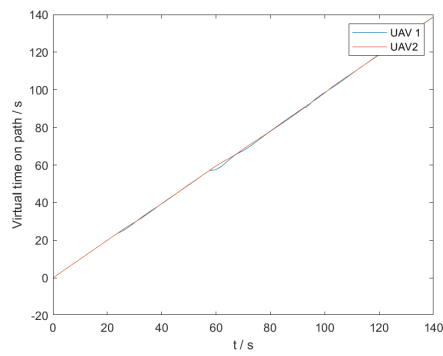
(b) Lateral distance to current path-segment.



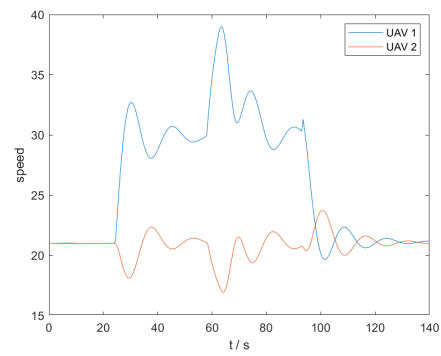
(c) UAV 1 attitude (Euler angles).



(d) UAV 2 attitude (Euler angles).

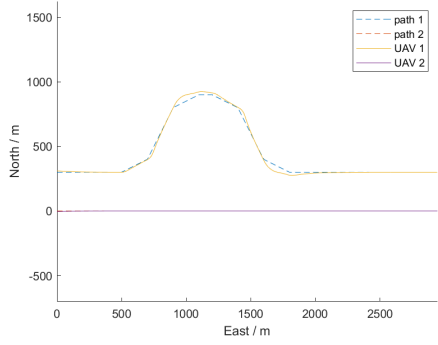


(e) Virtual-time.

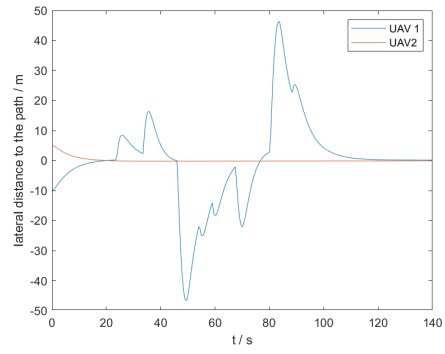


(f) Airspeed.

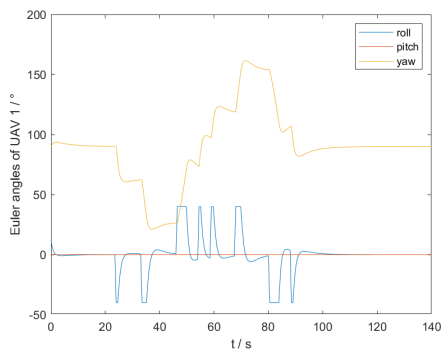
Figure 4.2: Triangle detour simulation results (Linearised).



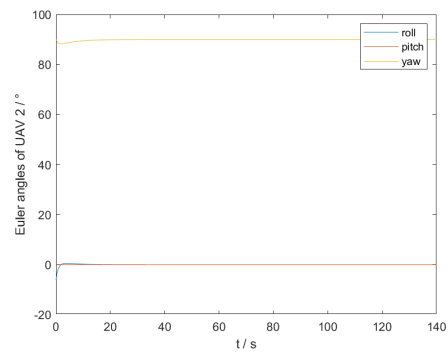
(a) UAVs trajectories.



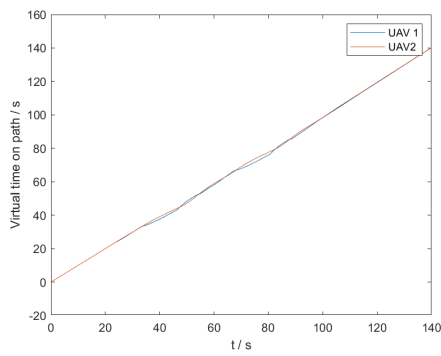
(b) Lateral distance to current path-segment.



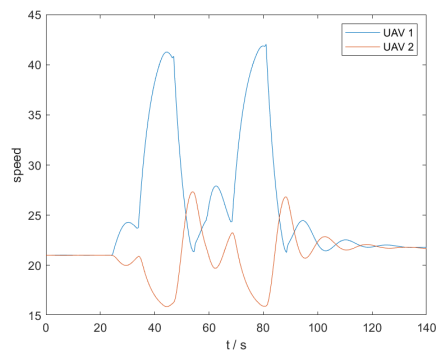
(c) UAV 1 attitude (Euler angles).



(d) UAV 2 attitude (Euler angles).



(e) Virtual-time.



(f) Airspeed.

Figure 4.3: Polygonal detour simulation results (Linearised).

Table 4.3: Path-following controller parameters (Linearised).

k_δ	$k_{\tilde{\psi}}$	k_{δ_I}	$k_{\delta_I,aw}$
$0.01 \text{ m}^{-1}\text{s}^{-1}$	1.4 s^{-1}	$7\text{e-}5 \text{ m}^{-1}$	20 m

Table 4.4: Initial conditions (Linearised).

\mathbf{P}_1	$\psi_{0,1}$	\mathbf{P}_2	$\psi_{0,2}$
$(310, 0, 0)^T \text{ m}$	90°	$(-5, 0, 0)^T \text{ m}$	90°

the UAVs is to travel side by side, while UAV 1 makes a detour, for example to avoid an obstacle. In the first scenario (figure 4.2), the detour is composed by only two segments, making the transitions more abrupt, while in the second (figure 4.3), the detour is composed of six segments.

The aircraft are able to track their respective paths, even though the roll set-point is saturated upon each switch. It may be seen that the path segment switching introduces discontinuities in the lateral distance to the path, having a larger amplitude in the first, where it reaches 160 m at 65 s on the greatest change in course, while in the second it is never greater than 50 m. These discontinuities in turn result in discontinuities in the roll angle set-point. The discontinuities also produce oscillations in the speed commands of both UAVs.

In both simulations, 2 min 10 s of flight are simulated. In the first simulation, the difference in virtual times at the end is of 63 ms which corresponds to a difference in distance of 1.3 m, with aircraft 2 in the lead. In the second, the differences are 98 ms and 2 m, with aircraft 2 in the lead. It may be observed that the waypoint switching implemented causes significant path-following errors after each switch. This in turn propagates to the time-coordination, which leads to significant oscillations in the commanded airspeeds of both UAVs. This is also the case even in the second simulation, where the angles between consecutive segments are more reduced.

To mitigate the path-following errors produced by each transition and its consequences, alternative switching algorithms that provide for smoother transitions, such as a fillet transition [31] or others mentioned in the literature review, could be investigated. For the time-coordination, the consideration of the heading of the aircraft in the feedback linearisation of the speed command in section 3.1, and the introduction of variable speed profiles in each segment could prove beneficial.

4.2 Vector Field Curved Path-Following Controller Simulation Results

Table 4.5: Path-following controller parameters (Vector-Field).

k_s	k_{ω}	χ^∞	k_d
1 s^{-1}	10 s^{-1}	70°	0.01 m^{-1}

In this section, two simulations using the vector-field curved path following controller are presented. In the first, the UAVs are required to follow two straight-line paths side-by-side, while in the second, UAV is required to circumvent an obstacle, similarly to the simulations already presented previously. Both

Table 4.6: Coordination control parameters (Vector-Field).

k_P	k_I	$k_{\text{coord,aw}}$
2 s^{-1}	0.1 s^{-2}	$2 \text{ m}^{-1} \text{ s}^2$

Table 4.7: Initial conditions (Vector-Field).

\mathbf{P}_1	$\psi_{0,1}$	\mathbf{P}_2	$\psi_{0,2}$
$(310, 0, 0)^T m$	90°	$(-5, 0, 0)^T m$	90°

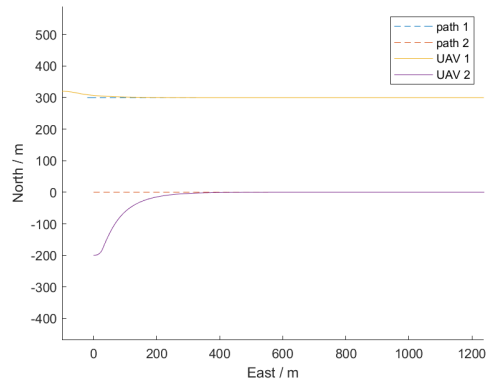
simulations use the same path-following, and coordination control parameters, presented in tables 4.5, and 4.6, respectively. In addition, the aircrafts in both simulations share the same initial conditions, presented in table 4.7.

4.2.1 Straight-line Paths Simulation Results

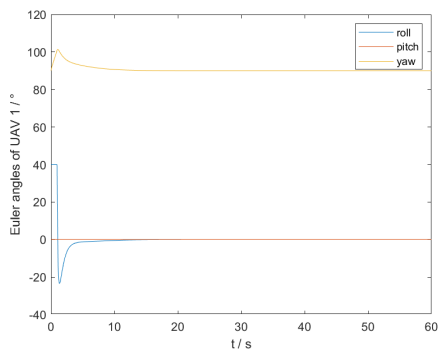
The results of a simulation scenario where the two UAVs are required to fly side-by-side are presented in figure 4.4. Figure 4.4a shows the trajectories of the UAVs, as well as the trajectories of their respective virtual points; figures 4.4b and 4.4c show the attitudes of both UAVs during the simulation; figures 4.4d, and 4.4e show the along and cross-tracking errors, respectively; figure 4.4f shows the commanded airspeed; and finally figure 4.4g shows the virtual time difference between UAV 1 and 2.

The starting position of UAV 1 relative to its assigned path is comparable to the ones for the UAVs in the linearised path following controller simulation. The controller achieves a faster convergence to the path (8 s settling-time at 99%, figure 4.4e) in relation to the initial convergence (20 s) to the straight-line segments. UAV 2 starts with a bigger offset from its path (200 m) and manages to converge to the path with a similar speed (19 s, figure 4.4e) as in the linearised simulation, for which the UAVs start much closer.

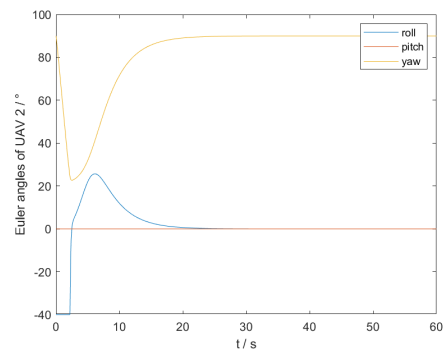
In terms of the coordination, the produced commanded airspeeds (figure 4.4f) are smooth. First, UAV 1 speeds-up, and UAV 2 reduces its speed, since 1 starts behind 2. This reverses at 4 s, since by then 1 has converged to the path, and has its heading aligned with it, while 2 needs to speed-up due to difference in heading from its path due to the convergence to it. At the end the commanded speeds have converged to the desired speed profile assigned on the path (21 m/s). Turning now to the coordination error (figure 4.4g), there is an initial spike in the error (1.75 s) due to the convergence of the virtual point of 1. Then, it decreases (in absolute value) until at the end of the simulation it is only 100 ms.



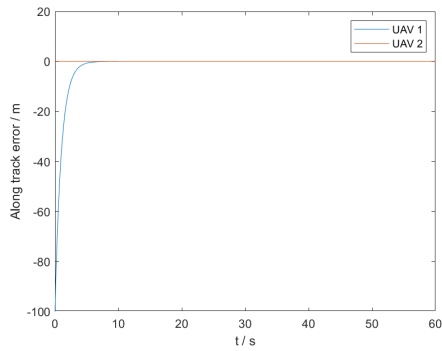
(a) Path



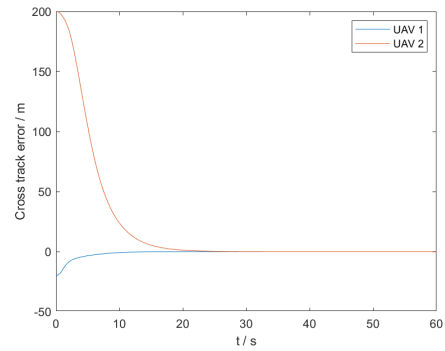
(b) UAV 1 attitude (Euler angles).



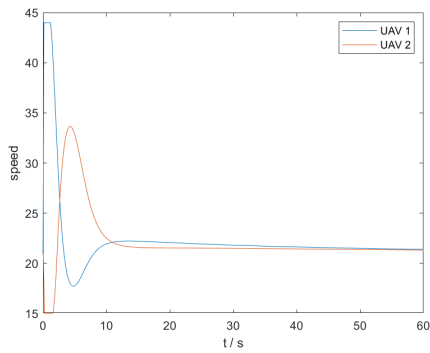
(c) UAV 2 attitude (Euler angles).



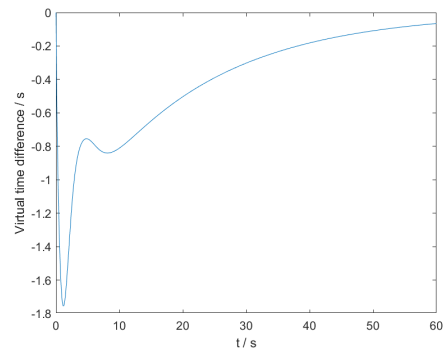
(d) Along-track errors.



(e) Cross-track errors.



(f) Commanded airspeeds.



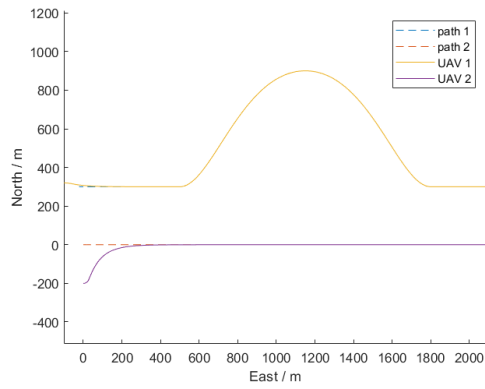
(g) Virtual-time coordination error.

Figure 4.4: Straight-line paths simulation (Vector-Field).

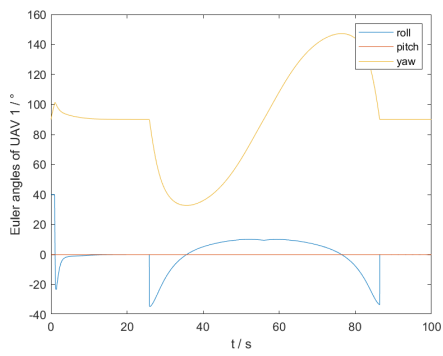
4.2.2 Detour Simulation Results

The former scenario is changed to make UAV 1 make a detour, similarly to the simulations presented for the linearised path-following controller in section 4.1. The detour is described by two cubic spline segments, chosen to be similar to the polygonal detour. The results are presented in figure 4.5 with the same simulation results format as in the previous figure.

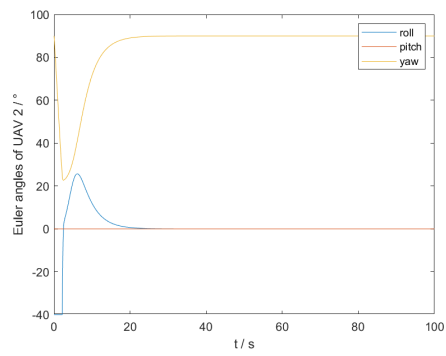
The initial behaviour of the system follows the one presented in the previous simulation. When UAV 1 arrives at 24 s the aircraft are close to being coordinated, and the generated speed commands are able to maintain the convergence of the coordination errors without any noticeable change. These speed commands are much smoother compared with the ones for the linearised controller. Also, the paths are followed more precisely, with a corresponding smoother roll angle set-point signal.



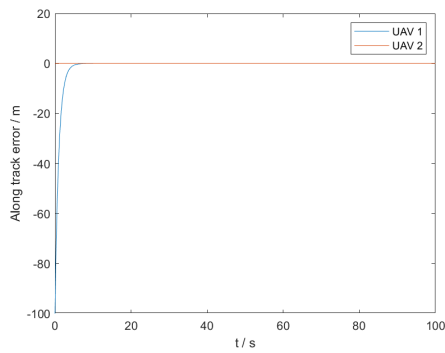
(a) UAV trajectories.



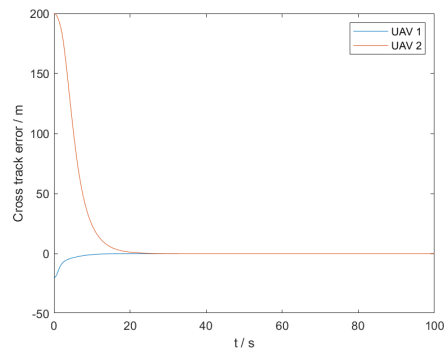
(b) UAV 1 attitude (Euler angles).



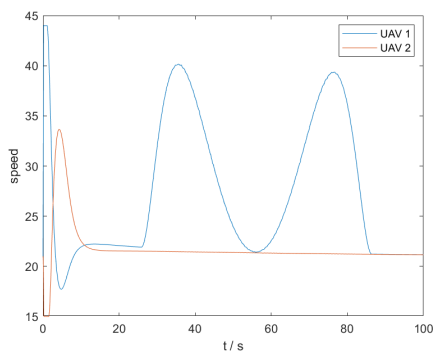
(c) UAV 2 attitude (Euler angles).



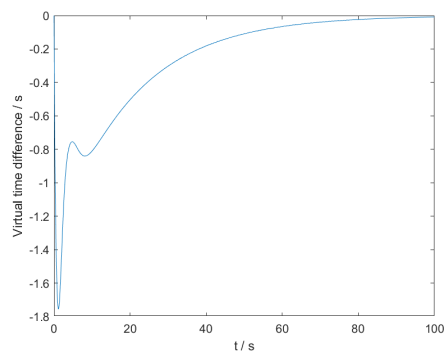
(d) Along-track errors.



(e) Cross-track errors.



(f) Commanded airspeed.



(g) Virtual time coordination error.

Figure 4.5: Detour path simulation (Vector-Field).

Chapter 5

Conclusions

This work explored the problem of coordinated flight of fixed-wing aircraft at a fixed altitude set-point by utilizing the framework of coordinated path-following. More specifically, the problem was partially decoupled by decomposing it first in an individual path-following problem by means of the attitude control of each aircraft (chapter 2), and then in a coordination problem of their along-path parameters by adjustment of their speeds around their nominal desired speed profiles (chapter 3).

Two path-following approaches capable of being integrated into the considered framework were considered in chapter 2. The first, a path-following controller for a straight-line path was synthesized (section 2.2). Its application was extended to paths consisting of sequences of straight-line segments by the use of the half-plane switching method. Its stability was studied, being concluded local asymptotic stability by the application of Lyapunov's indirect method (section 2.2.3).

The second path-following controller, capable of following general curved paths by means of a desired course vector-field, was described (section 2.3). The shortcomings of the stability analysis performed in [25] were corrected and extended to account for time-varying speeds, presenting conditions for which global asymptotic stability of the path-following system may be concluded (subsection 2.3.1).

In chapter 3, each of the path-following control laws were integrated with the proportional-integral consensus coordination law from Kaminer et al. [1] and Xargay et al. [2], used to coordinate the aircraft relatively on their desired time along their respective paths (virtual-time). For each of the previous path-following approaches a coordination map was defined from the along-path-length to the virtual time, and the dynamics of the virtual-time transformed into the canonical single integrator dynamics to which the coordination law may be applied (sections 3.1 and 3.2). In addition, the overall stability of the cooperative vector-field path-following control law was discussed (section 3.2).

The two methods developed in this work were implemented in a Simulink simulation environment, and the results of their simulations were then analysed and compared (chapter 4). The results illustrate that the non-linear algorithm performs better than the linearisation-based, since it tracks the path more precisely, and is able to achieve the coordination of the aircraft using smoother velocity commands and roll set-points for each of the vehicles.

5.1 Future Work

The analysis of the non-linear curved path-following algorithm in section 2.3.1 should be further analysed in an attempt to prove that for a sufficiently small initial along-track error, it remains small in order to be able to guarantee asymptotical stability for any path that satisfies the curvature constraint (2.50). Although it seems intuitive that, given the path parameter updating law (2.27), the along-track error is at least bounded for small initial values, a detailed analytical proof of this fact was not able to be confirmed.

Flight testing these cooperative path-following solutions, the next step would be to test these algorithms in simulation with the complete model of an aircraft and an inner-loop autopilot capable of tracking the commands of these cooperative path-following solutions.

In order to keep the complexity of the path-following algorithms within the scope of this work, the influence of the wind on the analysis was not considered. This, however, is an important direction of research for future work. In this respect, the use of the integral state in the linearised path-following controller allows for some degree of rejection of wind disturbances, but in the case of the vector-field curved path-following controller, the presence of wind leads to a steady-state path-following error. According to Zhao et al. [25], to compensate the effect of the wind, the airspeed and heading used in the control law could be substituted by the estimated ground-speed, and estimated course, respectively, by means of the wind triangle. To achieve this, a wind estimator would have to be integrated, whose combined stability together with the path-following controller would have to be studied. This analysis would be challenged further by the need to satisfy the kinematic input constraints (2.7) and (2.8).

Alternatively, 3-D path-following control laws could be explored, such as the ones developed in Yang et al. [27], Brezoescu et al. [29], and Liu, McAree, and Chen [30].

In regard to the time-coordination, there are other algorithms that could be explored such as the Kalman-consensus [6], or other non-linear algorithms [10, 11]. Additionally, a closer consideration to the questions of communication delays and losses could be undertaken.

Bibliography

- [1] Isaac Kaminer et al. *Time-Critical Cooperative Control of Autonomous Air Vehicles*. Cambridge, MA: Elsevier, 2017. ISBN: 978-0-12-809946-9. DOI: 10.1016/C2015-0-06603-5.
- [2] E. Xargay et al. “Time-Critical Cooperative Path Following of Multiple Unmanned Aerial Vehicles over Time-Varying Networks”. In: *Journal of Guidance, Control, and Dynamics* 36.2 (Mar.–Apr. 2013), pp. 499–516. DOI: 10.2514/1.56538.
- [3] Peng Shi and Bing Yan. “A Survey on Intelligent Control for Multiagent Systems”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 51.1 (Jan. 2021), pp. 161–175. DOI: 10.1109/TSMC.2020.3042823.
- [4] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. “Consensus and Cooperation in Networked Multi-Agent Systems”. In: *Proceedings of the IEEE* 95.1 (Jan. 2007), pp. 215–233. DOI: 10.1109/JPROC.2006.887293.
- [5] Wei Ren, Randal W. Beard, and Ella M. Atkins. “Information Consensus in Multivehicle Cooperative Control”. In: *IEEE Control Systems Magazine* 27.2 (Apr. 2007), pp. 71–82. ISSN: 1941-000X. DOI: 10.1109/MCS.2007.338264.
- [6] Derek B. Kingston, Wei Ren, and Randal W. Beard. “Consensus Algorithms Are Input-to-State Stable”. In: *Proceedings of the 2005, American Control Conference, 2005*. Vol. 3. June 8–10, 2005, pp. 1686–1690. DOI: 10.1109/ACC.2005.1470210.
- [7] E. Bıyık and M. Arcak. “Passivity-Based Agreement Protocols: Continuous-Time and Sampled-Data Designs”. In: *Group Coordination and Cooperative Control*. Ed. by K. Y. Pettersen, J. T. Gravdahl, and H. Nijmeijer. Vol. 336. Lecture Notes in Control and Information Science. Berlin, Heidelberg: Springer, 2006. ISBN: 978-3-540-33469-9. DOI: 10.1007/11505532_2.
- [8] Murat Arcak. “Passivity as a Design Tool for Group Coordination”. In: *IEEE Transactions on Automatic Control* 52.8 (Aug. 2007), pp. 1380–1390. DOI: 10.1109/TAC.2007.902733.
- [9] Ivar-André F. Ihle, Murat Arcak, and Thor I. Fossen. “Passivity-based designs for synchronized path-following”. In: *Automatica* 43.9 (Sept. 2007), pp. 1508–1518. DOI: 10.1016/j.automatica.2007.02.018.
- [10] Hao Chen et al. “Consensus in Networks of Nonlinear Integrators with Applications to Coordinated Path Following Control of Fixed-Wing UAVs”. In: *2020 59th IEEE Conference on Decision and Control (CDC)*. Dec. 14–18, 2020, pp. 5348–5353. DOI: 10.1109/CDC42340.2020.9304199.

- [11] Hao Chen et al. “Convergence Analysis of Signed Nonlinear Networks”. In: *IEEE Transactions on Control of Network Systems* 7.1 (Mar. 2020), pp. 189–200. DOI: 10.1109/TCNS.2019.2913550.
- [12] Chris Godsil and Gordon Royle. *Algebraic Graph Theory*. Graduate Texts in Mathematics 207. Harrisonburg, VA: Springer, 2001. ISBN: 0-387-95241-1.
- [13] J. Alexander Fax and Richard M. Murray. “Information Flow and Cooperative Control of Vehicle Formations”. In: *IEEE Transactions on Automatic Control* 49.9 (Sept. 2004), pp. 1465–1476. DOI: 10.1109/TAC.2004.834433.
- [14] Jialong Zhang, Jianguo Yan, and Pu Zhang. “Multi-UAV Formation Control Based on a Novel Back-Stepping Approach”. In: *IEEE Transactions on Vehicular Technology* 69.3 (Mar. 2020), pp. 2437–2448. DOI: 10.1109/TVT.2020.2964847.
- [15] Hao Chen et al. “Formation Flight of Fixed-Wing UAV Swarms: A Group-Based Hierarchical Approach”. In: *Chinese Journal of Aeronautics* 34.2 (Feb. 2020), pp. 504–515. DOI: 10.1016/j.cja.2020.03.006.
- [16] Randal W. Beard, Jonathan Lawton, and Fred Y. Hadaegh. “A Coordination Architecture for Spacecraft Formation Control”. In: *IEEE Transactions on Control Systems Technology* 9.6 (Nov. 2001), pp. 777–790. DOI: 10.1109/87.960341.
- [17] Diana Yanakiev and Ioannis Kanellakopoulos. “A Simplified Framework for String Stability Analysis in AHS”. In: *IFAC Proceedings Volumes* 29.1 (June–July 1996), pp. 7873–7878. DOI: 10.1016/S1474-6670(17)58959-4.
- [18] R. Ghabcheloo et al. “Coordinated Path-Following in the Presence of Communication Losses and Time Delays”. In: *SIAM Journal on Control and Optimization* 48.1 (Feb. 11, 2009), pp. 234–265. DOI: 10.1137/060678993.
- [19] Hao Chen et al. “Coordinated Path-Following Control of Fixed-Wing Unmanned Aerial Vehicles”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2021). DOI: 10.1109/TSMC.2021.3049681.
- [20] Yuanzhe Wang, Danwei Wang, and Senqiang Zhu. “Cooperative Moving Path Following for Multiple Fixed-Wing Unmanned Aerial Vehicles with Speed Constraints”. In: *Automatica* 100 (Feb. 2019), pp. 82–89. DOI: 10.1016/j.automatica.2018.11.004.
- [21] He Bai, Murat Arcak, and John Wen. *Cooperative Control Design. A Systematic, Passivity-Based Approach*. Vol. 89. Communications and Control Engineering. New York, NY: Springer, 2011. ISBN: 978-1-4614-0013-4. DOI: 10.1007/978-1-4614-0014-1.
- [22] Randal W. Beard and Jeffrey Humpherys. “Following Straight Line and Orbital Paths with Input Constraints”. In: *Proceedings of the 2011 American Control Conference*. June 29–July 1, 2011, pp. 1587–1592. DOI: 10.1109/ACC.2011.5990840.
- [23] John Hauser and Rick Hindman. “Maneuver Regulation from Trajectory Tracking: Feedback Linearizable Systems”. In: *IFAC Proceedings Volumes* 28.14 (June 1995), pp. 595–600. DOI: 10.1016/S1474-6670(17)46893-5.

- [24] P.B. Sujit, Srikanth Saripalli, and João Borges Sousa. “Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicles”. In: *IEEE Control Systems Magazine* 34.1 (Feb. 2014), pp. 42–59. DOI: 10.1109/MCS.2013.2287568.
- [25] Shulong Zhao et al. “Integrating Vector Field Approach and Input-to-State Stability Curved Path Following for Unmanned Aerial Vehicles”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50.8 (Aug. 2020), pp. 2897–2904. DOI: 10.1109/TSMC.2018.2839840.
- [26] Kazuo Tanaka et al. “3-D Flight Path Tracking Control for Unmanned Aerial Vehicles Under Wind Environments”. In: *IEEE Transactions on Vehicular Technology* 68.12 (Dec. 2019), pp. 11621–11634. DOI: 10.1109/TVT.2019.2944879.
- [27] Jun Yang et al. “Optimal Path Following for Small Fixed-Wing UAVs Under Wind Disturbances”. In: *IEEE Transactions on Control Systems Technology* 29.3 (May 2021), pp. 996–1008. DOI: 10.1109/TCST.2020.2980727.
- [28] Yueqian Liang and Yingmin Jia. “Combined Vector Field Approach for 2D and 3D Arbitrary Twice Differentiable Curved Path Following with Constrained UAVs”. In: *Journal of Intelligent & Robotic Systems* 83 (2016), pp. 133–160. DOI: 10.1007/s10846-015-0308-x.
- [29] A. Brezoescu et al. “Adaptive Trajectory Following for a Fixed-Wing UAV in Presence of Crosswind”. In: *Journal of Intelligent & Robotic Systems* 69 (2013), pp. 257–271. DOI: 10.1007/s10846-012-9756-8.
- [30] Cunjia Liu, Owen McAree, and Wen-Hua Chen. “Path-Following Control for Small Fixed-Wing Unmanned Aerial Vehicles under Wind Disturbances”. In: *International Journal of Robust and Nonlinear Control* 23.15 (July 7, 2012), pp. 1682–1698. DOI: 10.1002/rnc.2938.
- [31] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft. Theory and Practice*. Princeton, N.J: Princeton University Press, 2012. ISBN: 978-0-691-14921-9.
- [32] Hassan K. Khalil. *Nonlinear Systems*. 3rd ed. Always Learning. Upper Saddle River, N.J.: Prentice Hall, 2002. ISBN: 0130673897.
- [33] Stephen R. Griffiths. “Vector Field Approach for Curved Path Following for Miniature Aerial Vehicles”. In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Keystone, Colorado, Aug. 21–24, 2006. DOI: 10.2514/6.2006-6467.
- [34] Enric Xargay et al. “Multi-Leader Coordination Algorithm for Networks with Switching Topology and Quantized Information”. In: *Automatica* 50.3 (Mar. 2014), pp. 841–851. DOI: 10.1016/j.automatica.2014.02.004.
- [35] Enric Xargay et al. “Convergence of a PI Coordination Protocol in Networks with Switching Topology and Quantized Measurements”. In: *51st IEEE Conference on Decision and Control*. Maui, HI, Dec. 10–13, 2012, pp. 6107–6112. DOI: 10.1109/CDC.2012.6426392.