Master of Science Thesis in Electrical Engineering
Department of Electrical Engineering, Linköping University, 2020

Efficient Route Optimization for UAVs in Maritime Search and Rescue

Tobias Fridén

LINKÖPING UNIVERSITY

Master of Science Thesis in Electrical Engineering

Efficient Route Optimization for UAVs in Maritime Search and Rescue:

Tobias Fridén

LiTH-ISY-EX--YY/NNNN--SE

Supervisor: **Jonatan Olofsson**
ISY, Linköpings universitet
Fredrik Falkman
Airpelago

Examiner: **Daniel Axehill**
ISY, Linköpings universitet

Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Sammanfattning

Det här som vi har hållit på med är jätteviktigt faktiskt och det vi gjort blev bara sååå bra. Kanske inte helt otippat, men det glass är sååå gott!

Förresten har vi blivit bäst på att skriva rapporter, så nu ska ska vi inte gå in närmare på några detaljer såhär i sammanfattningen.

Abstract

If your thesis is written in English, the primary abstract would go here while the Swedish abstract would be optional.

Contents

Notation **ix**

1 Fixed-wing Unmanned Aerial Vehicles **1**
1.1 Fixed-wing dynamics 1
1.1.1 General definitions and terminology 1
1.1.2 Dynamic equations 3
1.2 ArduPlane autopilot 6
1.2.1 Navigation control loop 6
1.2.2 Altitude and velocity control loop 8
1.2.3 Mission representation and flight modes 8

2 Motion planning **11**
2.1 Motion planning with differential constraints 11
2.1.1 General definitions and terminology 11
2.1.2 Sampling based motion planning 12

Bibliography **15**

Index **16**

Notation

Några mängder

Notation	Betydelse
$\mathbb{N}$	Mängden av naturliga tal
$\mathbb{R}$	Mängden av reella tal
$\mathbb{C}$	Mängden av komplexa tal

Förkortningar

Förkortning	Betydelse
ARMA	Auto-regressive moving average
PID	Proportional, integral, differential (regulator)

1

Fixed-wing Unmanned Aerial Vehicles

Though not recieving the same amount of research and commercial interest as their multirotor counterparts, fixed-wing UAVs offer a number of advantages in many usecases. In the following sections a thorough description of general fixed-wing dynamics as well as a description of the specific platform used in this work will be presented.

1.1 Fixed-wing dynamics

This first section will give the reader a quick overview of some common definitions and terminology. A general purpose 6-DOF model of a fixed-wing UAV will also be derived.

1.1.1 General definitions and terminology

We begin with establishing some common definitions and terminology which will be used throughout this thesis. These definitions are used in many other works related to fixed-wing aircraft, such as [1], [2], [10].

Coordinate reference frames

Three different coordinate reference frames are used, an inertial frame, a body frame and a wind reference frame. This is convenient since some sensors, like GPS report their values in the inertial frame while others, like IMU sensors report values in the body frame. Other advantages of the different references will be made apparent when formulating the dynamic equations.

Definition 1.1 (Inertial frame). The inertial frame, denoted with subscript I is fixed relative to the earth. A position vector in the intertial frame is defined in the NED order as

$$\boldsymbol{p}_I = (p_N, p_E, -p_H) \tag{1.1}$$

where p_N points in the north direction, p_E points east and p_H points down towards the earth, in order to form a right hand positive coordinate system.

Definition 1.2 (Body frame). The body frame, denoted with subscript B is fixed in the UAV center of gravity. A position vector in the body frame is defined as

$$\boldsymbol{p}_B = (x, y, z) \tag{1.2}$$

where x points forward through the UAV, y points to the right and z points down as shown in Figure TODO

Definition 1.3 (Wind reference frame). The wind reference frame, denoted with subscript W is related to the current direction of motion through the air. A position vector in the wind reference frame is defined as

$$\boldsymbol{p}_W = (x_w, y_w, z_w) \tag{1.3}$$

where x_w points in the same direction as the current velocity vector $\boldsymbol{v}_I$, y_w points to the right of x_w and z points down relative x_w and y_w.

Attitude representation

The attitude of the UAV is represented by the *Euler angles*.

Definition 1.4 (Euler angles). The Euler angle vector is defined as

$$\boldsymbol{\Phi} = (\phi, \theta, \psi) \tag{1.4}$$

where the *roll angle* ϕ is rotation around the north inertial axis, the *pitch angle* θ is rotation around the east inertial axis and the *yaw angle* ψ is rotation around the downwards inertial axis.

The relationship between coordinates in the body frame and inertial frame is given by the rotation matrix

$$\mathcal{R}_B^I = \mathcal{R}_\phi^x \mathcal{R}_\theta^y \mathcal{R}_\psi^z = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.5}$$

This attitude representation is not defined for $\theta = \pm\pi/2$. However, such attitudes were deemed very unlikely in this work as the main focus is on level flight scenarios.

Aerodynamic angles

The *aerodynamic angles* relate the wind reference frame to the other coordinate frames.

Definition 1.5 (Aerodynamic angles). We first define the body frame velocity vector as

$$\boldsymbol{v}_B = (u, v, w) \tag{1.6}$$

The *angle of attack* α and *side slip* β are then defined as

$$\alpha = \arctan \frac{w}{u} \tag{1.7}$$

$$\beta = \arcsin \frac{v}{V_a} \tag{1.8}$$

where $V_a = \sqrt{u^2 + v^2 + w^2}$

Fixed-wing UAV

A fixed-wing UAV is equipped with two horizontal wings that are fixed in the body frame. In order to stay in the air, it needs to keep a minimum forward velocity

$$V > V_s \tag{1.9}$$

where V_s is the airframe-dependent *stall speed*. In order to navigate through the air, it is equipped with some or all of the following control surfaces:

- *Ailerons* to control ϕ
- *Elevators* to control θ
- *Rudders* to control ψ

The UAV is also equipped with one or several propellers that are used to create the thrust which increases the total energy of the system. These might be facing towards or against the direction of motion.

1.1.2 Dynamic equations

In the following section the dynamic equations of a general fixed-wing aircraft will be derived. The equations are based on the following common state representation:

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{p}_I & \boldsymbol{v}_B & \boldsymbol{\Phi} & \boldsymbol{\omega} \end{pmatrix} \tag{1.10}$$

where $\boldsymbol{\omega} = (p, q, r)$ are the angular rates in the body reference frame. These can be divided into translational and rotational dynamics which are presented separately.

Translational dynamics

The dynamic equation for the position $\boldsymbol{p}_I$ is directly given as

$$\dot{\boldsymbol{p}}_I = \mathcal{R}_B^I \boldsymbol{v}_B \tag{1.11}$$

Furthermore, Newtons second law of motion gives

$$\boldsymbol{F}_{tot} = m\dot{\boldsymbol{v}}_B + \boldsymbol{\omega} \times m\boldsymbol{v}_B \tag{1.12}$$

where m is the UAV mass and $\boldsymbol{F}_{tot}$ is the sum of all forces working on the UAV. These can be divided into

$$\boldsymbol{F}_{tot} = \boldsymbol{F}_g + \boldsymbol{F}_{aero} + \boldsymbol{F}_{thr} \tag{1.13}$$

where $\boldsymbol{F}_g$ is the gravitational force, $\boldsymbol{F}_{aero}$ is the aerodynamic force and $\boldsymbol{F}_{thr}$ is the thrust force. The thrust force is assumed to only work in the same direction as the x-axis in the body frame, so that

$$\boldsymbol{F}_{thr,B} = (T, 0, 0) \tag{1.14}$$

Rotational dynamics

The dynamic equations for the Euler vector $\boldsymbol{\Phi}$ are given by

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & -\sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi\sec\theta & \cos\phi\sec\theta \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \tag{1.15}$$

By again using Newtons second law for the moment we derive

$$\boldsymbol{M} = \boldsymbol{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \boldsymbol{I}\boldsymbol{\omega} \tag{1.16}$$

The moment $\boldsymbol{M}$ is assumed to mainly come from aerodynamics.

Aerodynamics

The aerodynamic forces and moments generated require some further analysis. The aerodynamic force $\boldsymbol{F}_{aero}$ can be written in the wind reference frame as

$$\boldsymbol{F}_{aero} = \begin{pmatrix} -D \\ Y \\ -L \end{pmatrix} = \begin{pmatrix} \bar{q}SC_D \\ \bar{q}SC_Y \\ \bar{q}SC_L \end{pmatrix} \tag{1.17}$$

where $\bar{q} = \frac{1}{2}\rho(h)V^2$ is the free-stream dynamic pressure, S is the wing surface area and C_D, C_Y and C_L are dimensionless constants. D, Y and L are called the *drag*, *sideforce* and *lift* respectively. The corresponding coefficients are dependent on a number of variables, such as the aerodynamic angles and airframe dependent parameters [10].

The aerodynamic moments are usually defined in the body reference frame as

$$\boldsymbol{M}_{aero} = \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} = \begin{pmatrix} \bar{q}SbC_l \\ \bar{q}SbC_m \\ \bar{q}SbC_n \end{pmatrix} \tag{1.18}$$

where b is the wing-span of the aircraft and C_l, C_m and C_n are dimensionless constants mainly dependent on the aerodynamic angles [1].

Wind effects

The dynamic equations derived in (1.11)-(1.18) are under the assumption that the air through which the UAV is travelling is at rest relative to the earth. During real flights, wind will cause the air to move which has non-negligible effects on the UAV. The wind vector is often defined in the inertial frame as

$$\boldsymbol{w} = (w_N, w_E, w_H) \tag{1.19}$$

$\boldsymbol{w}$ can be decomposed to

$$\boldsymbol{w} = \boldsymbol{w}_m + \boldsymbol{w}_s \tag{1.20}$$

where $\boldsymbol{w}_m$ is the mean wind and $\boldsymbol{w}_s$ is some stochastic process [8]. In this work, we only consider the mean wind. The mean wind magnitude $W = |\boldsymbol{w}|$ varies with altitude as

$$W(h) = W_{10}(\frac{h}{h_{10}})^a \tag{1.21}$$

where W_{10} is the wind measured at $h = 10$ meters and a is the Hellman exponent, which depends on the shape and coastal location of the underlying terrain [1]. The wind can be incorporated in the dynamic equations of motion by extending (1.11) to

$$\dot{\boldsymbol{p}} = \mathcal{R}_B^I \boldsymbol{v}_B + \boldsymbol{w} \tag{1.22}$$

Also, we need to extend (1.12) by replacing $\boldsymbol{v}_B$ with

$$\boldsymbol{v}_g = \boldsymbol{v}_B + \boldsymbol{w}_B \tag{1.23}$$

where $\boldsymbol{v}_g$ is the body-fixed velocity relative the ground and $\boldsymbol{w}_B$ is the wind vector in the body frame.

Complete dynamics model

The complete dynamic model is given by combining the above equations. It will not be written out here, but is given in e. g. [1]. The dynamics are clearly non-linear and quite complex. Extensive modeling and experimentation, such as described in [9] is needed to identify the different parameters which describe the behaviour of a specific airframe.

It is often interesting to study the *longitudal* and *lateral* dynamics separately, where *longitudal* denotes movement in the vertical plane and *lateral* denotes movement out of the vertical plane. The decoupled state vectors are given by

$$\boldsymbol{x}_{lng} = (p_N, p_H, u, w, \theta, q) \tag{1.24}$$

and

$$\boldsymbol{x}_{lat} = (p_E, v, \phi, \psi, p, r) \tag{1.25}$$

By introducing the polar inertial components (V, γ) and writing the dynamic equations in the wind reference frame the longitudal dynamics become

$$\dot{V} = \frac{1}{m}[T\cos\alpha - D - mg\sin\gamma] \tag{1.26}$$

$$\dot{\gamma} = \frac{1}{mV}[T\sin\alpha + L - mg\cos\gamma] \tag{1.27}$$

$$\dot{q} = \frac{M_y}{I_{yy}} \tag{1.28}$$

$$\dot{\alpha} = \dot{\theta} - \dot{\gamma} = q - \frac{1}{mV}[T\sin\alpha + L - mg\cos\gamma] \tag{1.29}$$

Assuming level flight in the longitudal direction, i. e. u, V and p_H are constant, the lateral dynamics can be formulated as

$$\dot{p}_E = u\sin\psi + v\cos\phi\cos\psi \tag{1.30}$$

$$\dot{\psi} = r\cos\phi \tag{1.31}$$

$$\dot{\phi} = p \tag{1.32}$$

$$\dot{v} = \frac{Y_B}{m} + g\sin\phi - ru \tag{1.33}$$

$$\dot{p} = \frac{I_{zz}M_x + I_{xz}M_z}{I_{xx}I_{zz} - I_{xz}^2} \tag{1.34}$$

$$\dot{r} = \frac{I_{xz}M_x + I_{xz}M_z}{I_{xx}I_{zz} - I_{xz}^2} \tag{1.35}$$

$$\tag{1.36}$$

1.2 ArduPlane autopilot

The ArduPlane autopilot is an open source autopilot for fixed-wing UAVs [6]. It contains high-level controllers for navigation, velocity and altitude control as well as low level logic to command the attitude and throttle of the vehicle. In the following section the underlying theory of the relevant control loops for this thesis will be presented.

1.2.1 Navigation control loop

The ArduPlane autopilot the L_1 control law described in [7] for navigation. The goal of the control loop is to follow a straight line from a start coordinate (x_s, y_s)

to a goal coordinate (x_g, y_g). This is obtained by aiming towards a point P which is located at a fixed distance L_1 from the UAV. The logic behind the controller is illustrated in Figure 1.1, where (x, y) is the UAV position and ψ is the UAV yaw angle.

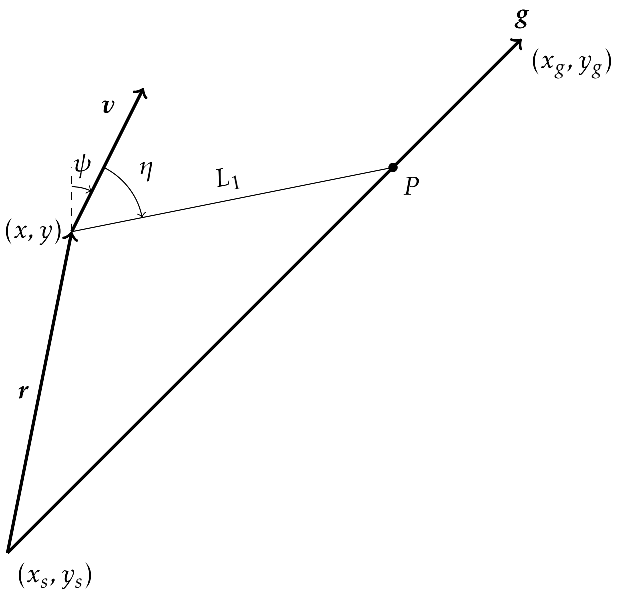

Figure 1.1: *L_1 controller logic*

In the ArduPilot implementation, the distance L_1 is calculated as

$$L_1 = \begin{cases} \frac{1}{\pi}\zeta \Delta T V & \text{if} \quad |\frac{1}{\pi}\zeta \Delta T V| > |\boldsymbol{r} - \boldsymbol{g}| \\ |\boldsymbol{r} - \boldsymbol{g}| & \text{otherwise} \end{cases} \tag{1.37}$$

where $V = |\boldsymbol{v}|$, ζ is the damping factor and ΔT is the update period of the controller [7].

In each time step, the control law corresponds to following a circular segment with radius

$$R = \frac{L_1}{2 \sin \eta} \tag{1.38}$$

which is tangent to $\boldsymbol{v}$ in (x, y). η is defined as the angle between the UAV velocity vector $\boldsymbol{v}$ and the line from the UAV to P. This circular segment is followed by issuing a lateral acceleration command

$$a_{cmd} = 2\frac{V^2}{L_1} \sin \eta \tag{1.39}$$

The lateral acceleration command is translated to a desired roll angle

$$\phi_{cmd} = \cos\theta \tan^{-1}(a_{cmd}/g) \tag{1.40}$$

where g is the gravitational constant. The low-level attitude controller is then used to track the desired roll.

1.2.2 Altitude and velocity control loop

ArduPlane uses a combined control loop to handle both desired altitude and velocity, called TECS (Total Energy Control System). This controller is based on the total energy of the UAV, which is defined as

$$E_T = \frac{1}{2} m V^2 + mgh \tag{1.41}$$

where h is the altitude relative to the takeoff point. The total energy rate is derived by taking the derivative with respect to time as

$$\dot{E}_T = mV\dot{V} + mg\dot{h} \tag{1.42}$$

The specific energy rate is then

$$\dot{E}_S = \frac{\dot{E}_T}{mgV} = \frac{\dot{V}}{g} + \frac{\dot{h}}{V} = \frac{\dot{V}}{g} + \sin\gamma \tag{1.43}$$

If γ is small, we get

$$\dot{E}_S \approx \frac{\dot{V}}{g} + \gamma \tag{1.44}$$

The longitudinal aircraft dynamics give

$$T - D = \frac{\dot{V}}{g} + \gamma \tag{1.45}$$

Thus, by increasing the thrust energy is added to the system. By changing the pitch angle using the elevators, the balance between kinetic and potential energy can be modified.

1.2.3 Mission representation and flight modes

A *mission* $\mathcal{M}$ is defined as

$$\mathcal{M} = \{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\} \tag{1.46}$$

i. e. a sequence of n *waypoints* represented in the inertial frame as

$$\boldsymbol{p} = (p_N, p_E, -p_H, c_{wp}) \tag{1.47}$$

where c_{wp} represents the waypoint command. There are many different waypoint commands available in ArduPlane, but this work will be focused on

$$c_{wp} \in \{Waypoint, Takeoff, Land\} \tag{1.48}$$

Waypoint mode

In the *waypoint* mode uses the navigation control loop to navigate along the line from $\boldsymbol{p}_i$ to $\boldsymbol{p}_{i+1}$. When $\boldsymbol{p}_{i+1}$ is reached, the flight mode is updated depending on the next c_{wp}. The next waypoint is assumed to be reached when

$$|\boldsymbol{p}_{UAV} - \boldsymbol{p}_{wp}| < R_{wp} \tag{1.49}$$

where R_{wp} is defined by the user.

Takeoff mode

In *takeoff* mode, the plane will attempt to reach the given altitude $h_{takeoff}$, while keeping a constant heading given by the launch direction. The takeoff can be triggered either by throwing the UAV, launching it from a catapult or on a runway [5].

Land mode

In *Land* mode, the plane will attempt to land at a given coordinate. The landing approach is divided into two different stages, the *approach* stage and *flare* stage.

During the approach stage, the UAV tries to accomplish the commanded *glide slope*, which is dependent on the previous waypoint position relative to the landing point. When the altitude decreases below h_{flare}, it enters the flare stage which means the throttle is completely turned off. During this stage the UAV will simply try to hold a target descent rate $\dot{h}_{flare}$ which is defined by the user [4].

2

Motion planning

In this chapter the necessary background regarding motion planning is introduced. In this thesis, motion planning is defined as the task of finding a path from a starting state to a goal state which fulfills a given set of constraints. These constraints might include differential constraints of the system and obstacle avoidance among others.

2.1 Motion planning with differential constraints

2.1.1 General definitions and terminology

We begin with introducing general definitions and terminology that are used to describe motion planning in this thesis.

Motion planning terminology

First we shall define some common terms used in motion planning.

Definition 2.1 (State and action spaces). We define the *state space* $\mathcal{X}$ and *action space* $\mathcal{U}$ as the set of obtainable states x and available actions u for the studied system. $\mathcal{X}$ can be further divided into

$$\mathcal{X} = \mathcal{X}_{free} + \mathcal{X}_{obs} \tag{2.1}$$

where $\mathcal{X}_{obs}$ are states which contain some kind of obstacle.

Definition 2.2 (Motion plan). A motion plan is defined as a sequence of states

$$\{x(t_1), \dots, x(t_n)\} \in \mathcal{X}_{free} \tag{2.2}$$

and actions

$$\{u(t_1), \dots, u(t_n)\} \in \mathcal{U} \tag{2.3}$$

which takes the system from a specified initial state $x(t_1) = x_S$ to a goal state $x(t_n) = x_G$ while fulfilling

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} f(x, u)dt \tag{2.4}$$

where $f(x, u)$ is the *transition function*.

Differential constraints

Differential constraints restrict the set of possible actions and states that the system can obtain. An important class of systems under differential constraints are *non-holonomic* systems.

Definition 2.3 (Non-holonomic system). In a *non-holonomic* system, the current state $x(t)$ is dependent on in which order the actions $u(t_i), \quad t_i < t$ where performed.

A formal definition and extensive discussion of non-holonomic systems is given in [3, Chapter 15]. Systems only capable of motion in a direction dependent on the current state, such as cars and fixed-wing UAVs belong to this class of systems.

2.1.2 Sampling based motion planning

Both $\mathcal{X}$ and $\mathcal{U}$ are generally continous, and need to be sampled in some way. This means that the resulting path will only be *resolution complete*, i. e. the optimality of the plan will depend on the sampling resolution d. In sampling based motion planning a *reachability graph* is commonly used.

Definition 2.4 (Reachability graph). Given a starting state $x_0(t_0) \in \mathcal{X}_d$, we define the *reachable set* $R(x_0, \mathcal{U}_d)$ as the set of states which are reached by applying any action $u \in \mathcal{U}_d$. By incrementally calculating the reachable set for each $x \in R(x_0, \mathcal{U}_d)$ we create the *reachability tree* $\mathcal{T}_r(x_0, \mathcal{U}_d)$. The reachability tree is a directed graph where each vertex consists of a state x which is reachable from x_0 by applying some action sequence $\{u_1, \dots, u_n\} \in \mathcal{U}_d$. By pruning any duplicate states from $\mathcal{T}_r$ we finally reach the *reachability graph* $\mathcal{G}_r(x_0, \mathcal{U}_d)$

Forward simulation

The next state in G_r given a specified input action u is obtained by integrating the transition function $f(x, u)$ on $[0, \Delta t]$. In practice this integral is calculated using some numerical approximation method. A common choice is the fourth-order *Runge-Kutta integration method*

$$x(\Delta t) \approx x(0) + \frac{\Delta t}{6}(w_1 + 2w_2 + 2w_3 + w_4) \tag{2.5}$$

where

$$\begin{aligned} w_1 &= f(x(0), u) \\ w_2 &= f(x(0) + \frac{1}{2}\Delta t w_1, u) \\ w_3 &= f(x(0) + \frac{1}{2}\Delta t w_2, u) \\ w_3 &= f(x(0) + \Delta t w_3, u) \end{aligned} \tag{2.6}$$

Bibliography

[1] Cornel-Alexandru Brezoescu. *Small lightweight aircraft navigation in the presence of wind*. PhD thesis, 2013.

[2] Waqas Khan and Meyer Nahon. Dynamics modeling of a highly-maneuverable fixed-wing uav. 04 2013.

[3] Steven M. La Valle. *Planning Algorithms*. 2006.

[4] Ardupilot: The open source autopilot. Arduplane automatic landing, 2019. URL `http://ardupilot.org/plane/docs/automatic-landing.html`.

[5] Ardupilot: The open source autopilot. Arduplane automatic takeoff, 2019. URL `http://ardupilot.org/plane/docs/automatic-takeoff.html`.

[6] Ardupilot: The open source autopilot. Arduplane, 2019. URL `http://ardupilot.org/plane/index.html`.

[7] Ardupilot: The open source autopilot. Arduplane: Navigation tuning, 2019. URL `http://ardupilot.org/plane/docs/navigation-tuning.html`.

[8] L. Rodríguez, F. Balampanis, J. A. Cobano, I. Maza, and A. Ollero. Energy-efficient trajectory generation with spline curves considering environmental and dynamic constraints for small uas. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1739–1745, Sep. 2017. doi: 10.1109/IROS.2017.8205987.

[9] Jieliang Shen, Yan Su, Qing Liang, and Xinhua Zhu. Calculation and identification of the aerodynamic parameters for small-scaled fixed-wing uavs. *Sensors*, 18:206, 01 2018. doi: 10.3390/s18010206.

[10] Daniel Simon. *Fighter Aircraft Maneuver Limiting Using MPC - Theory and Application*. PhD thesis, 09 2017.

Index

ARMA
abbreviation, ix

PID
abbreviation, ix