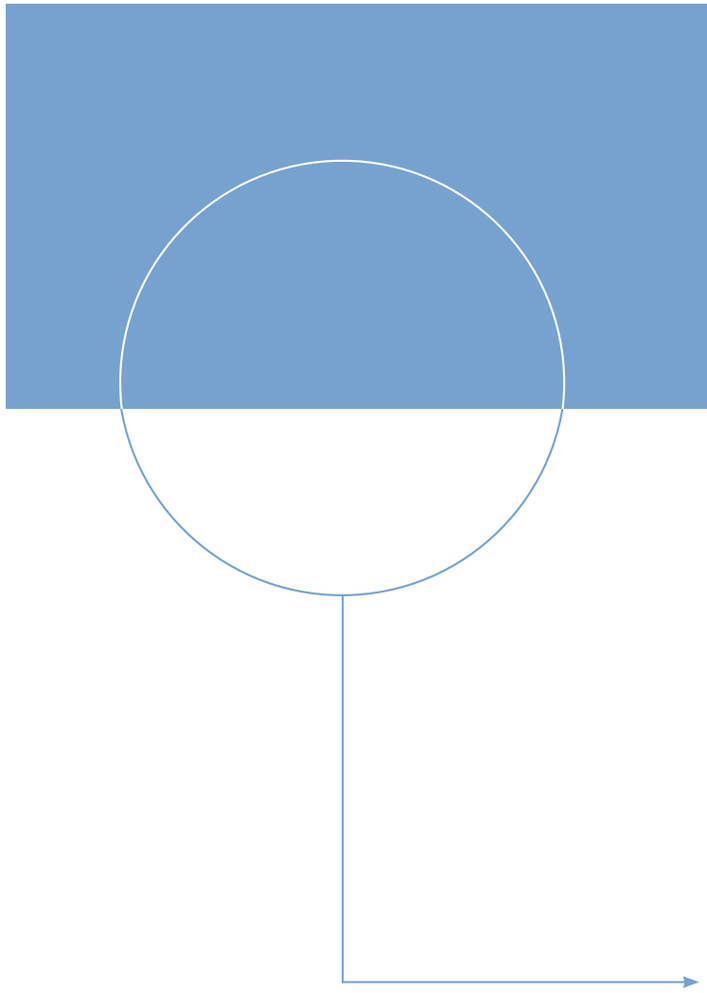

Master's thesis

2020

Martin Madsen

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology and Electrical
Engineering
Department of Engineering Cybernetics

Martin Madsen

Wing-Augmented Articulated Intervention Autonomous Underwater Vehicle

June 2020

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Wing-Augmented Articulated Intervention Autonomous Underwater Vehicle

Martin Madsen

MTTK
Submission date: June 2020
Supervisor: Kristin Y. Pettersen

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Problem description

Picture courtesy of Eelume AS

Articulated intervention-AUVs (AIAUVs) have emerged from swimming snake robots and combine the slender, articulated body of snakes with the propulsion provided by thrusters. The AIAUVs were recently proposed by the NTNU snake robotics research group and are now industrialized by the spin-off company Eelume AS.

The AIAUVs combine several benefits and capabilities of ROVs and survey AUVs into one robot: They have advantageous hydrodynamic properties and can travel long

distances, like survey AUVs; they have hovering capabilities and they can perform light intervention tasks, like ROVs. Moreover, their slender and flexible body provides access that supersedes existing marine robots. AIAUVs therefore mitigate the shortcomings of conventional marine robots, and thus enable autonomous operations for ocean sustainment and exploration, including both observation and intervention operations in the same mission, e.g., mapping the seabed and collecting sediments, inspecting and repairing the net of an aquaculture fish cage, and detecting and gathering plastic and other debris polluting the oceans. This new marine robot is already well on its way towards disrupting subsea operations in the oil and gas industry.

In this project we will continue reaping the benefits of combining the best from biology with the best from technology, and will here consider articulated gliders. Specifically, we will keep the articulated body of the AIAUVs and equip this with wings such as those of glider AUVs.

1. Continue the work from the preproject to extend the mathematical model given in Schmidt-Didlaukies et al. (2018)[1] to include the hydrodynamic effects of wings (lift and drag), and with the new addition of arbitrarily specifying placement and orientation of the wings on the AIAUV body.

2. Implement a framework in MATLAB that extends the existing AIAUV simulation-script to support easily configurable and multiple wings.

3. Develop a control system specifically tailored for the articulated glider, to fully utilize the lift to obtain energy efficiency.

4. Compare the power consumption of the AIAUV and the articulated glider with the same control method and discuss the potential benefits and drawbacks of wings.

[1] Schmidt-Didlaukies, H. M., Sørensen, A. J. and Pettersen, K. Y. (2018). Modeling of articulated underwater robots for simulation and control. DOI: `https://doi.org/10.1109/AUV.2018.8729806`

Abstract

With the effect rising global temperatures and pollution have on the worlds oceans monitoring the ocean's health and sustainability is a pressing issue. Two classes of underwater robots that have shown great promise for ocean exploration and surveying in recent years are the Articulated intervention autonomous underwater vehicle and the underwater glider. The Articulated intervention autonomous underwater vehicle (AIAUV) consists of a slender articulated body equipped with thrusters, and can perform light intervention tasks. While the underwater glider is able to travel great distances by gliding through the ocean with wings and an efficient buoyancy-engine. In this thesis we will merge these two robots by equipping the AIAUV with the wings from the glider.

The aim of this thesis is to investigate if this new wing-augmented AIAUV can utilise its wings to reduce its power consumption. To this end we extend the original dynamic model of an AIAUV to include the hydrodynamic effects of wings. We also create a tailored control-architecture for this new class of underwater robot enabling it to track waypoints by curving its body to induce a pitching motion.

With this new control-architecture we perform a simulation study that compares the power consumption for a wing-augmented AIAUV and one without wings. The results from these simulations indicate that the winged AIAUV requires significantly less energy than its unwinged counterpart as the wings enable the AIAUV to better regulate its pitch.

Sammendrag

Økende temperaturer og forurensing av havet har ført til et økt ønske om å kartlegge havets klima og bærekraftighet. To klasser av undervannsroboter som særlig har utmerket seg for dette formålet i nyere tid er den artikulerte intervensjons-AUV'en og undervannsglidere. Den artikulerte intervensjons-AUV'en (AIAUV) består av en artikulert slank kropp utstyrt med thrustere og kan utføre lette intervensjons-oppdrag. På den annen side har undervannsglideren muligheten til å dekke store distanser takket være sine vinger og effektive ballast-tank. I denne oppgaven vil vi slå sammen disse to robotene ved å utstyre AIAUV'en med vingene fra undervannsglideren.

Formålet med oppgaven er å undersøke om denne nye vinge-utrustede AIAUV'en kan benytte vingene sine for å redusere energiforbruket sitt. Mot dette formålet utvider vi den orginale dynamiske modellen for en AIAUV til å inkludere den hydrodynamiske effekten fra vinger. Vi designer også en skreddersydd kontrollarkitektur for denne nye klassen av undervannsroboter som muligjør følging av veipunkt ved å krumme kroppen sin for å indusere en stampende rotasjon (Eng. pitch).

Med denne nye kontrollarkitekturen gjennomfører vi et simuleringstudie hvor vi sammenligner effektforbruket for en AIAUV med og uten vinge. Resultatene fra disse simuleringene viser at AIAUV'en med vinger trenger markant mindre energi enn den uten vinger siden den med vinger er bedre til å regulere stamp.

Contents

List of Tables

List of Figures

Preface

This master's thesis is submitted as a part of the requirements for the master degree at the Department of Engineering Cybernetics at the Norwegian University of Science and Technology. The work presented in this thesis has been carried out under the supervision of Prof. Kristin Y. Pettersen at the Department of Engineering Cybernetics, NTNU.

This master's thesis is a continuation of a specialization project I conducted during the autumn of 2019. As is customary, the specialization project is not published. This means that important background theory and methods from the project report will be restated in full throughout this report to provide the best reading experience. Below, a complete list of the material included from the specialization project is listed.

- Chapter 3 (Covers material from the preproject but has been rewritten and extended)
- Chapter 4 (Covers material from the preproject but has been rewritten and extended)
- Chapter 5 (excluding section 5.6)
- Chapter 6
- Chapter 7 (section 7.1, the rest of the chapter has been rewritten and extended)

During the project, I have been provided with a `MATLAB`-script that implements the AIAUV-dynamics presented in Schmidt-Didlaukies et al. (2018). The script was

created by Henrik Schmidt-Didlaukies, Department of Marine Technology, NTNU. During the master's project I have created a more robust framework for performing simulations of these dynamics, extended the implementation of dynamics to support wings, implemented controller architecture that enables waypoint tracking for the AIAUV, and created plotting and system analysis tools. Professor Marilena Greco, Department of Marine Technology, NTNU, supported my work by answering questions related to hydrodynamic modeling of wings. The Ph.D. candidates Marianna Wrzos-Kaminska, and Erlend Andreas Basso, Department of Engineering Cybernics, NTNU, and Ph.D. candidate Henrik Schmidt-Didlaukies, Department of Marine Technology, NTNU, have also supported my work by answering questions related to the AIAUV model.

Unless otherwise stated, all figures and illustrations have been created by the author.

I would like to dedicate this thesis to my late father, Hans Christian Kristensen. He always believed in me and was always supportive of me pursuing this degree. I would also like to thank my mother Evy-Ann Madsen and my two brothers, Jo-Christian Kristensen and Karl Sander Kristensen for always being there for me.

Next I would like to thank Kaja Eriksen, Haakon Hammarbeck, Simen Heimly, and Aurora Haraldsen for all the wonderful times we had during our studies here at NTNU. And thanks to all my childhood friend for all the support they have given me over the years.

Last, but certainly not least, thank you to my loving girlfriend, Siri Eggen. Your never ending support brightens even the darkest days.

Martin Madsen
Trondheim, June 2020

Chapter 1

Introduction

1.1 Motivation

It has long been mankind's dream to explore the depths of the ocean, and with the effect rising global temperatures and pollution have on the worlds oceans, monitoring the ocean's health and sustainability is a pressing issue. Two classes of underwater robots have shown great promise for this purpose in recent years, the Articulated intervention autonomous underwater vehicle and the underwater glider. The Articulated intervention autonomous underwater vehicle (AIAUV) consists of a slender articulated body equipped with thrusters and can perform light intervention tasks. While the underwater glider is able to travel great distances by gliding through the ocean with wings and an efficient buoyancy-engine. In this thesis we will merge these two robots by equipping the AIAUV with the wings from the glider. Through this augmentation we hope that the AIAUV can utilise its wings to reduce its power consumption.

In an effort to investigate the energy savings of this new platform we propose to create a simulation model. The AIAUV dynamics can be modeled according to Schmidt-Didlaukies et al. (2018), while one example of modeling glider dynamics is Graver and Leonard (2001). We will attempt to merge the wing-dynamics of the glider into the AIAUV model, enabling us to simulate the behaviour of this new winged

AIAUV. The wing model builds on established theory from the marine community, and methods for modeling wings can be found in for example Faltinsen (2006).

We also develop a control system to properly utilise the wings. This is a new class of robot and as such not much research exist on the topic. As a first step in creating a control architecture tailored for this class of robots we propose to merge existing control strategies for marine vehicles with a novel new pitch controller. The control strategies for marine vehicles are summarised in Fossen (2011), while the pitch controller is a repurposed heading controller from Sans-Muntadas et al. (2017).

Finally we implement the extended model and control architecture in MATLAB in order to perform a simulation study comparing an AIAUV with and without wings. Our MATLAB-implementation is an extension of an existing script for a conventional AIAUV that was created by Henrik Schmidt-Didlaukies.

1.2 Assumptions

- Throughout this thesis we restrict our study to motion lying solely in the vertical $X - Z$-plane.

- All quantification of power and energy is idealised in the sense that we consider only the kinetic energy that actuators provide to the system. There are no actuator models or dynamics, i.e. a thruster is considered as a source that can instantly produce the commanded force.

- The base model of the AIAUV is one suited for system design and analysis and we design our extended dynamics with that in mind.

- The model for hydrodynamic forces from the wings are only valid for small angles of attack and in ideal quasi-steady flow.

- We do not explicitly model the ocean flow. The hydrodynamic effects of the ocean are incorporated into the rigid-body dynamics of the AIAUV.

- We assume no ocean currents, uniform density of water, $\rho_w = 1000\text{kgm}^{-3}$ and that we are far enough beneath the free surface so that no cavitation or broaching of the wings occur.
- All controllers operate with perfect measurements and at the same frequency as the simulation is performed.
- The direction of gravity in the inertial frame is along the negative z-axis. $g = [0, 0, -9.81]^{\text{T}}$

1.3 Contributions

The main contributions of the work presented in this thesis are as follows:

- The extension of AIAUV-dynamics to incorporate the hydrodynamic effects of wings at a resolution and fidelity suitable for system design, analysis and control purposes.
- A framework supporting the addition of multiple, arbitrarily placed wings with different geometries and hydrodynamic properties into the AIAUV simulation model.
- System evaluation tools for estimating power and energy use of the AIAUV.
- Custom control architecture encompassing guidance, pitch adjustment, speed control, and low level joint control. Together this enables waypoint tracking in the $X - Z$-plane for the AIAUV with and without wings.
- An extensive simulation comparing system performance for the AIAUV with and without wings when tracking waypoints.

1.4 Outline

The report is organized as follows. Chapter 2 covers relevant necessary mathematical background material. Chapter 3 and Chapter 4 together documents the base model

for the AIAUV based on Schmidt-Didlaukies et al. (2018). Chapter 5 is a literature review on underwater gliders. Chapter 6 presents relevant hydrodynamic theory for modeling wings. Chapter 7 details our extension of AIAUV dynamics to include wings. Chapter 8 describes the method used for quantifying the power use of the AIAUVs. Chapter 9 details the structure of our control hierarchy. Chapter 10 contains a search for optimal wing configuration for the AIAUV. Chapter 11 documents the necessary parameters for our simulations comparing AIAUV performance with and without wings, while Chapter 12 contains the necessary parameters to construct the AIAUVs used for simulation. The results from the simulation with and without wings are found in Chapter 13. Finally we conclude the thesis with Chapter 14 that summarise our major findings and give our suggestion for related future work on this topic.

Chapter 2

Mathematical Background

The purpose of this chapter is to present necessary mathematical concepts in order to develop the model for the AIAUV. The chapter will review both topics that were previously known to the author as well as new concepts that were previously unknown. Most of the material in this chapter is adapted from From et al. (2016).

Remark. *The notation in this chapter might deviate from the rest of the thesis, we apologize for the inconvenience.*

2.1 Reference frames and coordinate systems

A reference frame is a collection of points where the distance between any two points remains constant From et al. (2016). We can identify a reference frame with a rigid body. As every point on a rigid body will remain fixed with regards to other points on the rigid body.

An inertial reference frame is a reference frame where its points do not accelerate. The points can either be fixed in space or moving with constant velocity. The inertial reference frame that will be used in this thesis is the world-fixed inertial reference frame.

With an inertial reference frame and a reference frame associated with each rigid body in a system we can describe the motions of the rigid body system relative the inertial reference frame.

We can also attach a coordinate system to our reference frames to aid in the mathematical description of motion. A coordinate system consists of two parts: an origin, and a basis. The origin of the coordinate system, O, is a point that is fixed in the reference frame. A basis, $\{e_1, e_2, e_3\}$, for a coordinate system is a set of three linearly independent vectors that let us uniquely describe a points position with regards to the origin of the coordinate system.

Many possible choices of basis exist, but the most common one, and the one used in this thesis, is the orthonormal right-handed basis. For two vectors, x and y, $x \cdot y$ denotes the scalar product, and $x \times y$ denotes the vector product. The following properties must be satisfied for a orthonormal right-handed basis:

The basis vectors e_1, e_2 and e_3 are unit vectors if:

$$\begin{aligned} e_1 \cdot e_1 &= 1 \\ e_2 \cdot e_2 &= 1 \\ e_3 \cdot e_3 &= 1 \end{aligned} \tag{2.1}$$

The basis vectors e_1, e_2 and e_3 are mutually orthogonal if:

$$\begin{aligned} e_1 \cdot e_2 &= 0 \\ e_1 \cdot e_3 &= 0 \\ e_2 \cdot e_3 &= 0 \end{aligned} \tag{2.2}$$

The basis vectors e_1, e_2 and e_3 form a right-handed coordinate system if:

$$\begin{aligned} e_1 \times e_2 &= e_3 \\ e_2 \times e_3 &= e_1 \\ e_3 \times e_1 &= e_2 \end{aligned} \tag{2.3}$$

2.1.1 Representation of positions, velocities and accelerations

Given a reference frame, $\mathcal{F}_i$. with a corresponding basis, $\{e_x, e_y, e_z\}$ we can uniquely express the position of a point, b, relative the origin of the coordinate frame, O_i, as seen in Equation 2.4. Here x_{ib}, y_{ib} and z_{ib} are the components of the position along the basis vectors e_x, e_y, and e_z respectively.

$$p_{ib} = x_{ib}e_x + y_{ib}e_y + z_{ib}e_z \tag{2.4}$$

The velocity of a point can be found by time-differentiating the position of the point, as seen in Equation 2.5. Note that the expression for position and velocity depends on a fixed Cartesian basis. In a Cartesian basis each component of the distance that describes the position of a point is the distance between the point along a basis vector and the plane formed by the other basis vectors. A Cartesian basis is what is most widely used in mechanical modeling and the only one that will be considered in this thesis.

$$v_{ib}^{i} = \frac{\mathrm{d}p_{ib}}{\mathrm{d}t} = \dot{x}_{ib}e_x + \dot{y}_{ib}e_y + \dot{z}_{ib}e_z \tag{2.5}$$

The acceleration of a point viewed in a reference frame can be found by further time-differentiating the velocity of the point, as seen in Equation 2.6.

$$a_{ib}^{i} = \frac{\mathrm{d}v_{ib}^{i}}{\mathrm{d}t} = \ddot{x}_{ib}e_x + \ddot{y}_{ib}e_y + \ddot{z}_{ib}e_z \tag{2.6}$$

2.2 Vectorial representation of kinematics

This section acts as a summary of useful vectorial notation for representing kinematics of a mechanical system. Kinematics is the description of admissable configurations and velocities of a mechanical system and how they are related. Some prior knowledge in this material is assumed, for a more detail development of this theory please consult literature such as Fossen (2011) or Egeland and Gravdahl (2002).

2.2.1 Translational motion in $\mathbb{R}^3$

Translational motion is the description of position, linear velocity, and linear acceleration of the origin of a coordinate frame with respect to another. Given an inertial coordinate frame, $\mathcal{F}_0$, with a Cartesian basis $\{e_x, e_y, e_z\}$ and a coordinate frame, $\mathcal{F}_i$ rigidly attached to a rigid body the position of the body can be completely described by:

$$p_{0i} = \begin{bmatrix} x_{0i} \\ y_{i0} \\ z_{i0} \end{bmatrix} \in \mathbb{R}^3 \tag{2.7}$$

The linear velocity of the coordinate frame, $\mathcal{F}_i$ relative the inertial frame $\mathcal{F}_0$, as observed from $\mathcal{F}_0$ is given by:

$$v_{0i}^0 = \dot{p}_{0i} = \begin{bmatrix} \dot{x}_{0i} \\ \dot{y}_{i0} \\ \dot{z}_{i0} \end{bmatrix} \in \mathbb{R}^3 \tag{2.8}$$

Finally, the acceleration of coordinate frame $\mathcal{F}_i$ relative $\mathcal{F}_0$ as observed from $\mathcal{F}_0$ is:

$$a_{0i}^0 = \dot{v}_{0i}^0 = \ddot{p}_{0i} = \begin{bmatrix} \ddot{x}_{0i} \\ \ddot{y}_{i0} \\ \ddot{z}_{i0} \end{bmatrix} \in \mathbb{R}^3 \tag{2.9}$$

2.2.2 Rotational motion in $\mathbb{R}^3$

The orientation of a coordinate frame relative another coordinate frame can be described by Euler angles. Several choices of Euler angle representations exist but the one we will use is the ZYX Euler angles, or roll, pitch, and yaw angles as they are commonly called. Given two coordinate frames $\mathcal{F}_0$ and $\mathcal{F}_i$ with coinciding origin we have three forms of simple rotations; rotations around the x-, y-, and z-axis.

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi) & cos(\phi) \end{bmatrix}, \qquad R_y(\theta) = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix}, \tag{2.10}$$

$$R_z(\psi) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Chapter 3

Base model for the AIAUV

The aim of this chapter is to review the modeling procedure from Schmidt-Didlaukies et al. (2018). We then show how this method can be utilised to form a dynamic model for an AIAUV. This dynamic model was the starting point for this thesis and developing the articulated glider. A thorough introduction to this dynamic model and its construction is warranted as a good understanding of it is helpful in augmenting the AIAUV with wings. The chapter will not be a complete review of the source material but instead focus on only the parts relevant to the AIAUV model.

The work on understanding the dynamic model started during the preproject. But most parts have been rewritten to aid in readability and to introduce the notation that will be used throughout this thesis.

3.1 General definitions and notation

AIAUV: AIAUV stands for Articulated Intervention Autonomous Underwater Vehicle. It is a class of underwater robots with an eel-like body, articulated joints and thrusters. It has a slender body with advantageous hydrodynamic properties, good manoeuvrability and hovering capabilities thanks to thrusters, and good manipulability thanks to using its whole body as a moving-base robot manipulator.

Link: The body of the AIAUV consists of several connected links. A link has a shape, volume and mass. A link can also have thrusters, end-effectors, sensors or joints attached to them. By modelling the kinematics and dynamics of all links of the AIAUV one can simulate how it would behave in the real world. The following holds for the model developed in this chapter and in Schmidt-Didlaukies et al. (2018):

- Each link is a non-deformable rigid-body with fixed mass and fixed volume.
- The thrusters attached to a link are ideal, with trivial dynamics, have no geometric shape or mass, and are modelled as forces with constant direction and varying magnitude when viewed with respect to the link its attached to.
- There are no sensors or end-effectors on any links in this model. All relevant sizes such as velocities and position are assumed to be measured perfectly.
- The joints in this model are all revolute, no prismatic joints are modelled. The joints are also modelled as ideal with trivial dynamics and as such instantly produce the commanded torque.

Inertia-frame: The inertia frame is denoted $\mathcal{F}_0$ and is an inertial right-handed Cartesian coordinate frame.

Link-frame: Each link has a associated right-handed Cartesian coordinate frame rigidly attached to its body. This coordinate frame is called the link-frame of link i, and is denoted $\mathcal{F}_i$.

Link pose: As a link is modelled as a rigid body its position can be completely described by the position and orientation of a reference frame rigidly attached to the body Egeland and Gravdahl (2002). Let $p_j^i \in \mathbb{R}^3$ be the position of the origin of frame $\mathcal{F}_j$ projected onto the basis of frame $\mathcal{F}_j$, and $\mathbf{R}_j^i \in SO(3)$ be the rotation matrix formed with columns being the projection of the basis of $\mathcal{F}_j$ unto the basis of $\mathcal{F}_i$. These two quantities will represent the position and orientation, respectively, of a frame relative another. Together the position and orientation can be used to specify the pose of a link either relative another link or the inertia frame depending on the choice of frame of reference.

Link configuration: The pose between of one reference frame relative another, p_j^i and $\mathbf{R}_j^i$, can be used to construct the configuration of frame $\mathcal{F}_j$ expressed in $\mathcal{F}_i$ which we denote $\mathbf{A}_j^i$. The configuration is an element of $SE(3)$ and is a matrix with dimensions $\mathbb{R}^{6\times6}$ with the following structure:

$$\mathbf{A}_j^i = \begin{bmatrix} \mathbf{R}_j^i & [p_j^i]_\times \mathbf{R}_j^i \\ \mathbf{0}_3 & \mathbf{R}_j^i \end{bmatrix} \tag{3.1}$$

Link velocity: A links velocity can be described in 6 degrees of freedom, (DOF), by $\nu_{i,j}^k \in \mathbb{R}^6$. Where $\nu_{i,j}^k$ is a concatenation of the linear and angular velocities of frame $\mathcal{F}_i$ relative $\mathcal{F}_j$ expressed in the basis of $\mathcal{F}_k$. Let the linear velocity be denoted $v_{i,j}^k \in \mathbb{R}^3$ and the angular velocity be denoted by $\omega_{i,j}^k \in \mathbb{R}^3$ and defined as the projection of the respective velocity of $\mathcal{F}_i$ relative $\mathcal{F}_j$ expressed in the basis of $\mathcal{F}_k$. Then $\nu_{i,j}^k$ is defined as the quantity given in Equation 3.2. Two special cases of link velocities are worth mentioning as they are used in the construction of the AIAUV dynamics. These are the body-fixed link velocity, $\nu_{0,i}^i$, and the spatial link velocity, $\nu_{0,i}^0$, for a given link i. Lastly, we remark that we changed the notation from Schmidt-Didlaukies et al. (2018) where they use $\xi_{i,0}^i$ to denote link velocities.

$$\nu_{i,j}^k = \begin{bmatrix} v_{i,j}^k \\ \omega_{i,j}^k \end{bmatrix} \tag{3.2}$$

Wrench: We use the term wrench to describe a collection of linear forces and moments acting in frame $\mathcal{F}_i$ and denote it as τ^i. Let a linear force and angular momentum expressed in frame $\mathcal{F}_i$ be denoted $\mathbf{f}^i \in \mathbb{R}^3$ and $\mathbf{m}^i$ respectively. The construction of a wrench is given in Equation 3.3a. A useful property of wrenches is that they can be expressed in another frame by utilising the configuration of the two frames in question as seen in Equation 3.3b.

$$\tau^i = \begin{bmatrix} \mathbf{f}^i \\ \mathbf{m}^i \end{bmatrix} \tag{3.3a}$$

$$\tau^i = \mathbf{A}^i_j \tau^j \tag{3.3b}$$

Joint: The definition of a joint in Schmidt-Didlaukies et al. (2018) is more general than what is typical in robotics literature. In particular the term joint can refer both to a rigid interconnection between bodies and unconstrained relative motion. We denote a joint i as ϑ_i, and we will try to make it clear what constitutes the specific joint in question. Each link i is also associated with a joint coordinate, ϑ_i, and joint velocity amplitude vector, ζ_i.

3.2 System kinematics

Kinematics is the description of motion, and in this case the description of motion of the AIAUV model in particular. Let the AIAUV consist of n rigid-body links, each having its own associated link frame. For each link, i, we have a link configuration relative the inertia-frame, $\mathbf{A}^0_i$, and a body-fixed link velocity, $\nu^i_{i,0}$. Now, one could collect all link configurations into one structure, $\mathbf{A}$, and all link-velocities into one vector ν, and use that as the kinematic representation of the AIAUV system as it includes the pose and motion in 6 DOF of all links in the system. We will refer to this as the excessive coordinate representation of the AIAUV. We call $\mathbf{A}$ the robot configuration and ν the excessive velocities, their construction can be found in Equations 3.4a and 3.4b respectively.

$$\mathbf{A} = \operatorname{diag}(\mathbf{A}^0_1, \ldots, \mathbf{A}^0_n) \in \mathrm{SE}(3)^n, \quad \dim(\mathbf{A}) = \mathbb{R}^{6n \times 6n} \tag{3.4a}$$

$$\nu = \begin{bmatrix} \nu^1_{1,0} \\ \vdots \\ \nu^n_{n,0} \end{bmatrix}, \quad \dim(\nu) = \mathbb{R}^{6n} \tag{3.4b}$$

As alluded to with the choice of wording with "excessive" above in the excessive coordinate representation it is exactly that. Through some clever transformations from Schmidt-Didlaukies et al. (2018) we can transform the system into a more minimal, less computationally heavy, representation utilising the joint coordinate, ϑ_i, and the

joint velocity amplitude vector ζ_i associated with each link.

Let ϑ and ζ be the concatenation of joint coordinates and joint velocity amplitudes respectively as defined in Equation 3.5. Note especially that it is not required that the dimensionality of ϑ and ζ are equal as $a = c$ is not a requirement.

$$\vartheta = \begin{bmatrix} \vartheta_1 \\ \vdots \\ \vartheta_n \end{bmatrix} \in \mathbb{R}^a \tag{3.5a}$$

$$\zeta = \begin{bmatrix} \zeta_1 \\ \vdots \\ \zeta_n \end{bmatrix} \in \mathbb{R}^c \tag{3.5b}$$

The body-fixed link velocities can be related to the joint coordinates and velocity amplitudes through a Jacobian matrix, $J_i(\vartheta) \in \mathbb{R}^{6\times a}$ as shown in Equation 3.6a. The Jacobian matrix, $J_i(\vartheta)$ can be computed iteratively by the algorithm shown in Equation 3.6b, which is adapted from Schmidt-Didlaukies et al. (2018). In Equation 3.6b the factor $\Phi_j(\vartheta_j) \in \mathbb{R}^{6\times a_i}$ is of full rank and its structure depends on the choice of joint coordinate, ϑ_j, notably it can be constant for common choices of joint variables.

$$\nu^i_{i,0} = J_i(\vartheta)\zeta \tag{3.6a}$$

$$\nu^i_{i,0} = \sum_{j=1}^{i-1} \mathbf{A}^i_j \Phi_j(\vartheta_j)\zeta_j \tag{3.6b}$$

Remark. *We simplified the algorithm in Equation 3.6b by not including the case for more general kinematic trees with branches that are developed in Schmidt-Didlaukies et al. (2018). The implementation of the AIAUV model we used for this experiment has no branches and as such a more simple method for computing $J_i(\vartheta)$ can be used.*

We also omit the development of the factor $\Phi_j(\vartheta_j)$ and the time derivative of the Jacobian matrices as they require an appreciation of Lie Algebra on manifolds of $SE(3)^n$ that is beyond the scope of this thesis.

The Jacobians in Equation 3.6a relating individual link velocities to the joint velocity can be collected to form the system Jacobian, $J(\vartheta)$ as shown in Equation 3.7a. Utilising the system Jacobian one can transform the system velocities between its excessive representation, ν, and their minimal representation, ζ by the transformation in Equation 3.7b. This allows us to specify the dynamics with the excessive representation where literature is readily available, while using a sparser representation for simulation and control purposes.

$$J(\vartheta) = \begin{bmatrix} J_1^{\mathrm{T}}(\vartheta) & \dots & J_n^{\mathrm{T}}(\vartheta) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{6n \times a} \tag{3.7a}$$

$$\nu = J(\vartheta)\zeta \tag{3.7b}$$

3.3 System dynamics

The starting point for forming a representation for the total AIAUV dynamics is the existence of a Lagrangian function of the form shown in equation 3.8. Where the Lagrangian consists of two terms. The first term represents the kinetic energy of the system and $M(\mathbf{A})$ is the system inertia matrix containing both rigid-body and hydrodynamic inertia. The second term is the potential energy of the system, $U(\mathbf{A})$, that contains both the gravity and buoyancy potential.

$$L(\mathbf{A}, \nu) = \frac{1}{2}\nu^{\mathrm{T}} M(\mathbf{A})\nu - U(\mathbf{A}) \tag{3.8}$$

Following the approach as in Schmidt-Didlaukies et al. (2018), that is based on Lee et al. (2017), and applying the Lagrange-d'Alembert principle one obtain the dynamics for the total system as shown in equation 3.9. Where $\Gamma(\mathbf{A}, \nu)\nu$ is the configuration-dependent hydrodynamical inertia, $\Omega(\mathbf{A}, \nu)\nu$ are due to the link velocities not being generalized velocities. The drag acting on the system is represented with $D(\mathbf{A}, \nu)\nu$, while $b(\mathbf{A})$ represents the buoyancy and drag wrenches acting on the system. Finally, τ is a collection of thruster forces, joint torques, and mechanical constraints affecting the system.

$$M(\mathbf{A})\dot{\nu} + \Gamma(\mathbf{A}, \nu)\nu + \Omega(\mathbf{A}, \nu)\nu + D(\mathbf{A}, \nu)\nu + b(\mathbf{A}) = \tau \tag{3.9}$$

Utilizing the system Jacobian transformation detailed in Equation 3.7b one can transform the dynamics from its excessive representation in ν to the generalized dynamics represented in ζ. The full transformation for all system matrices is further detailed in Schmidt-Didlaukies et al. (2018). The results are the dynamics of the form shown in equation 3.10. These dynamics are the one that are implemented for the simulation of the AIAUV.

$$H(\vartheta)\dot{\zeta} + C(\vartheta, \zeta)\zeta + F(\vartheta, \zeta)\zeta + g(\vartheta) = \eta \tag{3.10}$$

In equation 3.10 $H(\vartheta)$ is the system inertia matrix, $C(\vartheta, \zeta)$ the system Coriolis and centrifugal matrix, $F(\vartheta, \zeta)$ the system drag matrix and $g(\vartheta)$ contains forces due to gravity and buoyancy. The external forces acting on the system is collected in η. For further details on the coordinate transformation of the system we refer to Schmidt-Didlaukies et al. (2018) and From et al. (2016).

3.3.1 Hydrodynamic decoupling and simplification of system dynamics

The system dynamics represented in excessive coordinates from Equation 3.9 are in general highly coupled with non-sparse matrices. In order to simplify the modeling of individual links they introduce the term hydrodynamical decoupling in Schmidt-Didlaukies et al. (2018). Hydrodynamic decoupling assumes that the fluid interaction between the links are negligible. This enables the links to be modeled independently of each other and existing modeling procedures such as from Fossen (2011) can be used.

The result of this decoupling is that the term $\Gamma(\mathbf{A}, \nu)\nu$ vanishes, and all matrices become block-diagonal in Equation 3.9, enabling a decoupling on a per-link basis. The dynamics can then be individually modeled for each link with the following procedure shown in Equation 3.11. We denote this the link-dynamics and further specify their

construction in Chapter 4.

$$M_i \dot{\nu}^i_{i,0} + \Omega_i(\nu^i_{i,0})\nu^i_{i,0} + D_i(\nu^i_{i,0})\nu^i_{i,0} + b_i(A^0_i) = \tau_i \tag{3.11}$$

Remark. *The assumption that one can hydrodynamically decouple links might not be valid in reality as thrusters mounted on the link will produce movement in the fluid surrounding the AIAUV. However this decoupling greatly simplifies the effort in specifying the link dynamics, enabling a model suitable for system design and analysis.*

In particular the addition of wings to the AIAUV that follows later in this thesis is likely to increase the coupling between links as they produce lifting forces through hydrodynamic interaction. We chose to continue expanding the model under the assumption of full hydrodynamic decoupling with the same rationale that this model is for system design and analysis.

3.4 Choice of joint coordinates for the AIAUV model

As mentioned earlier there is some freedom in the choice of joint representation for a link. We need to specify a joint coordinate, ϑ_i, and a joint velocity amplitude, ζ_i, for each link i of the AIAUV. Properties that are common for all AIAUV's in this thesis is that they consist only of rigid links with thrusters or wings, or Cardan-joint links that permit rotations around two intersecting axes. All AIAUV's also start and end with a rigid link, and all rigid links are interconnected with exactly one Cardan-joint.

We denote one end of the AIAUV the tail link, and the opposing end the head link. The kinematic tree for the AIAUV is ordered such that the tail link is the first link and the head the n-th link. The first link has the most complex joint representation and is always the link that specify the pose and velocity relative the inertia frame. A summary of all joint coordinates and velocity amplitudes for an AIAUV is shown in Equation 3.12.

$$\begin{aligned}
\vartheta_1 &= \begin{bmatrix} p_1^{0\,\mathrm{T}} & \eta_1^0 & \epsilon_1^{0\,\mathrm{T}} \end{bmatrix}^{\mathrm{T}} & \zeta_1 &= \nu_{1,0}^1 & i &= 1 \\
\vartheta_i &= q_y & \zeta_1 &= \dot{\vartheta}_i & i &\text{ is even} \\
\vartheta_i &= q_z & \zeta_1 &= \dot{\vartheta}_i & i &\text{ is odd}
\end{aligned} \tag{3.12}$$

For the first link the joint coordinate is given by the position of the link frame, $\mathcal{F}_1$, expressed in the inertia frame, $\mathcal{F}_0$, and the orientation of the link frame relative the inertia frame. The orientation is represented with a unit quaternion, $[\eta_1^0, \epsilon_1^{0\,\mathrm{T}}]^{\mathrm{T}}$, where η_1^0 and $\epsilon_1^{0\,\mathrm{T}}$ denote the real and imaginary part respectively. The rationale for a quaternion representation for orientation is to avoid singularites associated with Euler angles, a more detailed discussion on this topic is found in Egeland and Gravdahl (2002). The joint velocity amplitude for the first link, ζ_1, is chosen simply as the link body velocites, $\nu_{0,1}^1$.

For the $n-1$ links that are not the tail, exactly half of those will be Cardan-joints, where each Cardan joint permit revolutions around two orthogonal axes. Let q_y and q_z be the permitted joint angle revolution around the y- and z-axis respectively of a Cardan-joint. Then by utilising all Cardan-joint angles we have enough coordinates to represent all remaining ϑ_i's and can use their angular velocity as ζ_i.

$$\begin{aligned}
\vartheta &= \begin{bmatrix} p_1^{0\,\mathrm{T}} & \eta_1^0 & \epsilon_1^{0\,\mathrm{T}} & q_1 & \dots & q_{n-1} \end{bmatrix}^{\mathrm{T}} \\
\zeta &= \begin{bmatrix} \nu_{0,1}^0 & \dot{q}_1 & \dots & \dot{q}_{n-1} \end{bmatrix}
\end{aligned} \tag{3.13}$$

By collecting all joint coordinates and velocity amplitudes we arrive at the expression in Equation 3.13. This formulation makes it more clear that our system can be viewed as a vehicle-manipulator system with the first link acting as the base of the vehicle and the rest of the links being the manipulator. See for example From et al. (2016) or Antonelli (2018) for more details on this topic. Note the result that odd-numbered q_i-coordinates represent y-revolute joints while even-numbered q_i's represent z-revolute joints, we also refer to them as pitching and yawing joints respectively.

3.5 Control mapping for AIAUV

The last remaining piece of the model that needs to be elaborated is the mapping of external forces η into system inputs, $\mathbf{u}$. From Equation 3.10 the term η contains all motor torques from joints and all thruster forces from thrusters. For control purposes we wish to address each individual motor and thruster by a scalar value u_i and collect them into one vector $\mathbf{u}$. Let m be the number of thrusters and $u_{t,i}$ be the control input for thruster i, and $k = n - 1$ be the number of revolute joints and $u_{q,j}$ be the torque of joint motor j. Then the control mapping can be achieved by Equation 3.14.

$$\eta = \mathbf{B}(\vartheta)\mathbf{u} \qquad \eta = \begin{bmatrix} \mathbf{0}_{m\times k} & B_\star(\vartheta) \\ \mathbf{I}_k & B_\star(\vartheta) \end{bmatrix} \begin{bmatrix} u_{q,1} \\ \vdots \\ u_{q,k} \\ u_{t,1} \\ \vdots \\ u_{t,m} \end{bmatrix} \tag{3.14}$$

In Equation 3.14 $\mathbf{I}_k$ is used to denote the identity matrix of dimension $k \times k$. As the motor joint angles and velocities are directly included in the ϑ and ζ representation of dynamics their transformation into scalar control inputs are quite trivial and more importantly static. This is reflected by the left block of $\mathbf{B}(\vartheta)$ being constant. The right part of $\mathbf{B}(\vartheta)$ is responsible of mapping the thrusts into the statespace and is configuration dependent. This is achieved with the thrust configuration matrix, $B_\star(\vartheta)$, that needs to be recomputed every time-step and the algorithm to construct it is further detailed in Schmidt-Didlaukies et al. (2018).

Chapter 4

Link dynamics for AIAUV

This chapter documents the description and parameters of the different links and their dynamics used to construct the system dynamics of a complete AIAUV. Together with the material in Chapter 3 this will be used to construct the simulation model of the AIAUV. All the different links and their names are summarised in table 4.1.

Link name
Long link
Short link
Short link w/ aft thruster
Cardan joint

Table 4.1: Names of AIAUV links.

To this end we present the necessary parameters and method for constructing a link by giving a generic example. The parameters of the links used in simulations later in this thesis can be found in Appendix A.

Remark. *This chapter represents our best effort to document the implementation of link dynamics in the* MATLAB*-script we were handed as starting point for our work. It is included for completeness and to ensure that our simulations are reproducible. The credit*

for this work belongs to the research group and the author of the script, Phd Candidate Henrik Schmidt-Didlaukies.

4.1 Dynamics of a generic link

An AIAUV link is modelled as a cylindrical elongated slender body with uniform thickness and flat ends. Each link is accompanied by a Euclidean coordinate frame called the link-frame of link i, denoted $\mathcal{F}_i$. The link frame is the basis for expressing the position and direction of gravity-, buoyancy- and thrust-forces as well as the basis for developing the link dynamics. A figure showing a typical link and how the link-frame is attached is shown in Figure 4.1.

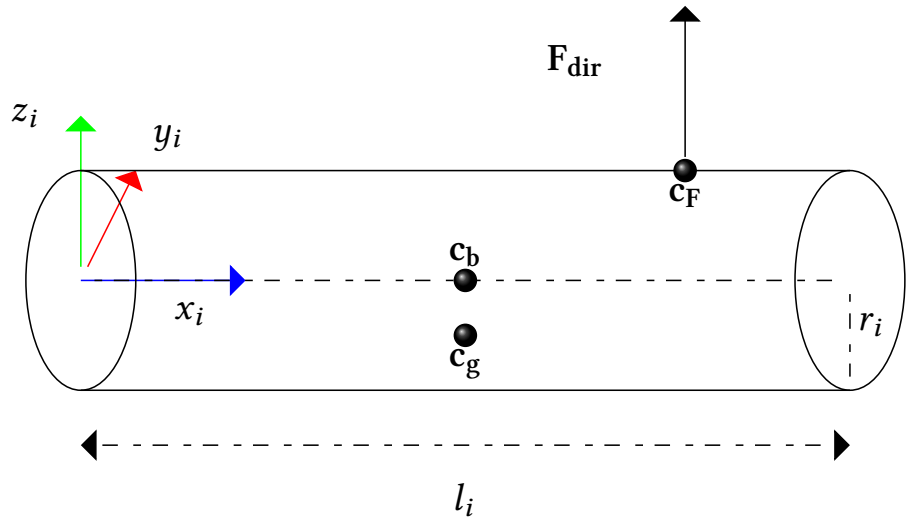

Figure 4.1: Sketch of a generic link with thruster placement, center of gravity and buoyancy, and associated link coordinate-frame.

A link's shape and size is parametrized by two quantities: the length of the link, l_i and its radius, r_i. The link frame has its origin positioned at the centre of one of the flat circular ends of the cylinder. The x-axis is directed along the centre-line of the

cylinder and the y- and z-axis form a plane coinciding with the base of the cylinder. Together, this forms an orthogonal right-handed Cartesian basis for the link.

The position of the centre of mass, c_g, centre of buoyancy, c_b, as well as the acting point of any thrusters, $c_{F,i}$ relative the link-frame must also be specified. The location of these points for a typical generic link is illustrated in Figure 4.1. In order to convert a thruster force, F_i, into a scalar control input u_i a vector describing the direction of thrust must also be specified. The thrust directions are expressed in the link-frame and assumed constant as each thruster is rigidly attached to a link. These thrust directional vectors are denoted $F_{dir,i} \in \mathbb{R}^3$.

Name	Generic link
Length $[m]$	l_i
Radius $[m]$	r_i
Mass $[kg]$	m_i
c_g $[m]$	$[x_g, y_g, z_g]^{\mathrm{T}}$
c_b $[m]$	$[x_b, y_b, z_b]^{\mathrm{T}}$
$c_{F,i}$ $[m]$	$[F_{x,i}, F_{y,i}, F_{y,i}]^{\mathrm{T}}$
$F_{dir,i}$	$[F_{dir,x,i}, F_{dir,y,i}, F_{dir,y,i}]^{\mathrm{T}}$

Table 4.2: Parameters for generic link

The necessary parameters for a given link will be summarized in tables such as this example for a generic link shown in Table 4.2.

We now want to develop the dynamics of a link in a form suitable for inclusion into the total AIAUV dynamics as presented in Chapter 3. To this end we wish to model the dynamics with the expression given in Equation 3.11. Here $\dot{\nu}^i_{i,0}$ and $\nu^i_{i,0}$ are the linear and rotational accelerations and velocities respectively of the link frame, $\mathcal{F}_i$, relative the inertia frame, $\mathcal{F}_0$.

$$M_i \dot{\nu}^i_{i,0} + \Omega_i(\nu^i_{i,0})\nu^i_{i,0} + D_i(\nu^i_{i,0})\nu^i_{i,0} + b_i(A^0_i) = \tau_i \tag{4.1}$$

Determining the parameters of Equation 4.1 from the parameters in Table 4.2 will complete the development of the dynamics. The parameters to be specified are the

mass-matrix, M_i, the Coriolis matrix, $\Omega_i(\xi^i_{i,0})$, the Drag-matrix $D_i(\xi^i_{i,0})$, the combined buoyancy- and gravity-wrench, $b_i(A^0_i)$ and finally the wrench of external forces and thrusts, τ_i.

The Mass-matrix consists of both the rigid-body mass and the added mass as shown in Equation 4.2. We also want to point out that M_i can be divided into four smaller submatrices as they are useful in the construction of the Coriolis matrix, $\Omega_i(\nu^i_{i,0})$.

$$M_i = M_{RB} + M_A = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{4.2}$$

The rigid body mass matrix, M_{RB} can be modeled using a quite standard procedure that can be found in e.g. Fossen (2011). The result is shown in Equation 4.3, where $\mathbf{I}_{3\times 3}$ is the 3×3 identity matrix, $S(\mathbf{v})$ is the skew-symmetric matrix representation of the vector $\mathbf{v}$, and I_R is the inertia-tensor for the link which is further detailed in Equation 4.4a.

$$M_{RB} = \begin{bmatrix} m_i \mathbf{I}_{3x3} & -m_i S(c_g) \\ m_i S(c_g) & I_R \end{bmatrix} \tag{4.3}$$

The expression for the inertia-tensor of a link, I_R might seem a bit unconventional at first. It is the superposition of two inertias: one for a cylinder, I_C and one for a smaller rod, I_r. This is done to reflect that the center of mass, c_g, might not coincide with the center of geometry, c_b. If we let $c_g = c_b = [\frac{l_i}{2}, 0, 0]^{\mathrm{T}}$ then $m_r = 0$ as $z_g = 0$ and it follows that $I_R = I_C$ as $I_r = \mathbf{0}$.

$$I_R = I_C + I_r = m_c \begin{bmatrix} \frac{r_i^2}{2} & 0 & 0 \\ 0 & \frac{r_i^2}{4} + \frac{l_i^2}{3} & 0 \\ 0 & 0 & \frac{r_i^2}{4} + \frac{l_i^2}{3} \end{bmatrix} + m_r \begin{bmatrix} z_g^2 & 0 & -\frac{l_i z_g}{2} \\ 0 & z_g^2 + \frac{l_i}{3} & 0 \\ -\frac{l_i z_g}{2} & 0 & \frac{l_i}{3} \end{bmatrix} \tag{4.4a}$$

$$m_r = \frac{z_g}{r_i}, \quad m_c = m_i - m_r \tag{4.4b}$$

The added mass matrix is computed with the expression in Equation 4.5, where $\rho = 1000\text{kgm}^{-3}$ is the density of water, $C_{a_C} = 1$ is the theoretical coefficient of the cross-section of the link, and here $\alpha = 0.2$ is the added mass ratio in surge and heave. For more details on how this matrix was constructed please consult literature on hydrodynamic modeling such as Faltinsen (2006).

$$M_A = \rho\pi r_i^2 C_{a_C} \begin{bmatrix} \alpha l_i & 0 & 0 & 0 & 0 & 0 \\ 0 & l_i & 0 & 0 & 0 & \frac{l_i^2}{2} \\ 0 & 0 & l_i & 0 & -\frac{l_i^2}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{l_i^2}{2} & 0 & \frac{l_i^3}{3} & 0 \\ 0 & \frac{l_i^2}{2} & 0 & 0 & 0 & \frac{l_i^3}{3} \end{bmatrix} \tag{4.5}$$

The Coriolis matrix $\omega_i(v_{i,0}^i)$ can be computed from the sub-matrices of M_i shown in Equation 4.2 and the result is shown in Equation 4.6. Note that we split the body-fixed link velocities into linear and angular parts in this expression with $v_{i,0}^i = [v_{i,0}^i, \omega_{i,0}^i]^\text{T}$.

$$\Omega_i(v_{i,0}^i) = \begin{bmatrix} \mathbf{0}_{3\times3} & -S(M_{11}v_{i,0}^i + M_{12}\omega_{i,0}^i) \\ -S(M_{11}v_{i,0}^i + M_{12}\omega_{i,0}^i) & -S(M_{21}v_{i,0}^i + M_{22}\omega_{i,0}^i) \end{bmatrix} \tag{4.6}$$

The Drag-wrench acting on a link $D_i(v_{i,0}^i)$ is modeled with a linear and quadratic term denoted $D_L(v_{i,0}^i)$ and $D_N(v_{i,0}^i)$ respectively. For further details on how they are constructed and their specific properties please consult Schmidt-Didlaukies et al. (2018), Fossen (2011), and Faltinsen (2006).

$$D_i(v) = D_N(v) + D_L(v) \tag{4.7}$$

Let $g = 9.81\text{ms}^{-2}$ be the magnitude of acceleration due to gravity, and W_i and B_i be the magnitude of the weight and buoyancy of link i respectively. Note that the direction of gravity is opposite the direction of the z-axis of the inertia-frame. Then the weight and the buoyancy force of the link can be expressed in the inertia frame and

denoted by $\mathbf{f}_g^0$ and $\mathbf{f}_b^0$ as shown in equation 4.8. Where $\rho = 1000\text{kgm}^{-3}$ is the density of water and $\nabla = \pi r_i^2 l_i$ is the volume of displaced water by the link.

$$\mathbf{f}_g^0 = \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -m_i g \end{bmatrix}, \quad \mathbf{f}_g^0 = \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \rho g \nabla \end{bmatrix} \tag{4.8}$$

However, we wish to express these forces in the link frame, as well as the moments they generate around the origin of the link frame. To this end let $\mathbf{R}_0^i(\Omega_{0,i})$ be the rotation matrix representing the rotation of the link frame relative the inertia-frame viewed from the link-frame. Expressing the gravity and buoyancy forces in the link frame can be accomplished with the transformation in Equation 4.9. The combined restoring wrench, $b_i(\mathbf{A}_i^0)$, can be computed with the expression in Equation 4.10. Note the minus as the wrench appears on the left hand side in Equation 4.1.

$$\mathbf{f}_g^i = \mathbf{R}_0^i(\Omega_{0,i})\mathbf{f}_g^0, \quad \mathbf{f}_b^i = \mathbf{R}_0^i(\Omega_{0,i})\mathbf{f}_b^0 \tag{4.9}$$

$$\mathbf{b}_i(\mathbf{A}_i^0) = -\begin{bmatrix} \mathbf{f}_g^i + \mathbf{f}_b^i \\ \mathbf{c}_g \times \mathbf{f}_g^i + \mathbf{c}_b \times \mathbf{f}_b^i \end{bmatrix} \tag{4.10}$$

The external forces and moments, τ_i, in Equation 4.1 consists of thruster forces and moments, as well as constraint forces and moments as shown in Equation 4.11. The constraint wrench, $\tau_{constraints}$, is not explicitly modelled as the kinematic constraints that they represent vanish under the transformation to joint coordinates detailed in Chapter 3. As such we assume it is zero, and trust the transformation to enforce it.

$$\tau_i = \tau_{constraints} + \sum_{j=1}^{m} \tau_{thruster,j} \tag{4.11}$$

Let's now concentrate on the thruster force and moments for a particular thruster, for brevity we denote this τ_j. τ_j will be represented as a vector $\tau_j = [F_x, F_y, F_z, M_x, M_y, M_z]^{\mathrm{T}} \in \mathbb{R}^6$ consisting of the linear forces the thruster applies to the link body and the moments

this force generates around the link-frame origin. Utilizing the parameters from Table 4.2 we can form an expression for the wrench given by Equation 4.12.

$$\tau_j = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} F_{dir,j} \\ c_{F,j} \times F_{dir,j} \end{bmatrix} u_j = B_j u_j \tag{4.12}$$

The expression above is beneficial for control-purposes as it maps the thruster force into a constant matrix B_j that represents the directionality of the thrust force on the link-dynamics and a scalar u_j representing the magnitude of the thrust in Newton. This method also scales for a link with multiple thrusters with some small modifications. Let m denote the total number of thrusters on a link i. Then let $c_{F,j}$ denote the point of action of the force from thruster j on the link, $F_{dir,j}$ is the direction this force is acting along, and this is valid for all $j \in [1, \ldots, m]$. With this we can express the combined wrench from all thrusters on the link with Equation 4.13.

$$\sum_{j=1}^{m} \tau_j = \begin{bmatrix} B_1 & \cdots & B_m \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \tag{4.13}$$

4.2 Documenting link parameters

With the method described in the Section above we show that the only necessary parameters to reproduce the link-dynamics are physical constants and the parameters found in Table 4.2. These parameters can be found in Appendix A for each link-type that are used in future simulations presented in this thesis. For the inclined reader we also include the numerical values of static matrices, $\mathbf{M}_{RB}$ and $\mathbf{B}$ for each link.

Chapter 5

Underwater Gliders

This chapter is a brief literature review on underwater gliders that was conducted during the preproject of this thesis. It explores the history of the underwater glider, mechanical components often found in their design, an example of how to model dynamics of underwater gliders, and common control strategies.

The chapter concludes with a section written as part of this thesis and not part of the preproject with Section 5.6 that pinpoints the major takeaways from this study. Also conducted during work on this thesis is a review of the work of Professor Ralf Bachmayer on augmenting a SLOCUM glider with a propeller, a brief summary of which is also included in this section.

5.1 History of Underwater gliders

The novel idea of an underwater autonomous gliding vehicle was first proposed in Henry Stommel | Woods Hole Oceanographic institution (1989). Here the author envisions a near future where the underwater gliders are an integral part of our endeavours to explore and understand the ocean. These efficient and robust vehicles can traverse great distances and due to their longevity and simple construction large scale oceanographic surveys are a possibility, similar to how we have a network of

weather-balloons monitoring the atmosphere.

Fast forward three decades and science has moved closer to Stommel's vision. Underwater gliders are now a reality and are used for oceanographic surveys. Javaid, Muhammad Yasar et al. (2014) gives a broad overview of different commercial and scientific gliders and their performance. One typical trend to observe is their long operational range which merits their study for the purpose of extending the energy-efficiency of the AIAUV.

5.2 Anatomy of an underwater Glider

To better understand how a glider achieves their energy-efficiency a study of their mechanical composition is warranted. The anatomy of a typical underwater glider makes it highly suitable for long-distance, autonomous operations as explained in Javaid, Muhammad Yasar et al. (2014). A glider will usually consist of a slender body equipped with wings, a buoyancy-regulating subsystem, an internal movable mass, power supply, and a sensor and navigation suite. Its main mode of motion is to alter its net buoyancy to either sink or float and manipulate internal mass distribution to adjust attitude. The wings will then induce forward motion from the hydrodynamic lift-forces arising and propel the glider forward. By combining sequences of controlled descent and ascend the glider can traverse the ocean in a sawtooth-like pattern. The end result is a vehicle that can generate net forward motion without the need for external thrusters.

By avoiding having thrusters exposed to the environment the gliders are able to withstand the harsh conditions of the ocean. The glider also consumes very little energy as the main work that the vehicle needs to perform in transit is to change its net buoyancy force at the bottom and top of its glide path. Maintaining this unbalanced buoyancy requires no additional work from actuators. During descent and ascend the only work that is necessary is to ensure tracking of desired attitude, a relatively cheap operation.

5.3 Glider dynamics

A brief introduction of a dynamic model of an underwater glider developed for control purposes could be useful in order to understand its working principles, as well as serving as inspiration for how the AIAUV-model can be augmented with wings. Several papers suggest ways of modeling the dynamics, one of these are Graver and Leonard (2001), where they derive the equations of motion of the ROGUE glider. The ROGUE glider is equipped with a buoyancy-drive and movable masses and its equations of motion will reflect this by having more states than e.g. a thruster-actuated AUV. A small summary of relevant equations are restated below.

We chose to only restate the reduced longitudinal dynamics, in this paper. The study of glider dynamics was conducted with the goal of understanding how they incorporate the hydrodynamic effects due to the foil and it is in the longitudinal plane it becomes the most apparent. The reduction in degrees of freedom is obtained by constraining the glider to only move in the $x-z$-plane of the inertial frame.

5.4 Longitudinal equations of motions for ROGUE glider

The equations of motion are shown in equation 5.1. The equations have been converted to follow the notation from Fossen (2011). Where $[u\ v\ w]^{\mathrm{T}}$ and $[p\ q\ r]^{\mathrm{T}}$ are body fixed linear and angular velocities, and $[x\ y\ z]^{\mathrm{T}}$ and $[\phi\ \theta\ \psi]^{\mathrm{T}}$ are linear positions expressed in the inertial frame and Euler angles respectively.

$$\dot{x} = u\cos\theta + w\sin\theta \tag{5.1a}$$

$$\dot{z} = -u\sin\theta + w\cos\theta \tag{5.1b}$$

$$\dot{\theta} = q \tag{5.1c}$$

$$\dot{\theta} = \frac{1}{I_y}\big((m_3 - m_1)uw - \bar{m}g(r_{P1}\cos\theta + r_{P3}\sin\theta) + M_{DL} - r_{P3}\mathbf{u}_1 + r_{P1}\mathbf{u}_3\big) \tag{5.1d}$$

$$\dot{u} = \frac{1}{m_1}\big(- m_3 wq - P_{P3}q - m_0 g \sin\theta + L\sin\alpha - D\cos\alpha - \mathbf{u}_1\big) \tag{5.1e}$$

$$\dot{w} = \frac{1}{m_3}\big(m_1 uq + P_{P1}q + m_0 g\cos\theta - L\cos\alpha - D\sin\alpha - \mathbf{u}_3\big) \tag{5.1f}$$

$$\dot{r}_{P1} = \frac{1}{\bar{m}}P_{P1} - u - r_{P3}q \tag{5.1g}$$

$$\dot{r}_{P3} = \frac{1}{\bar{m}}P_{P3} - w + r_{P1}q \tag{5.1h}$$

$$\dot{P}_{P1} = \mathbf{u}_1 \tag{5.1i}$$

$$\dot{P}_{P3} = \mathbf{u}_3 \tag{5.1j}$$

$$\dot{m}_b = \mathbf{u}_4 \tag{5.1k}$$

Most of the expressions in the equation of motion can be verified to conform to the representation of equations of motion for marine vehicles presented in Fossen (2011). However some additional terms and states are introduced to model how the moving masses affect the vehicle dynamics. The states and terms containing the movable mass, $\bar{m}$, its position, r_{P1} and r_{P3}, and its linear momentum, P_{P1} and P_{P3} alter the vehicle dynamics.

The last unexplained terms are L, D, M_{DL} and α, that introduce the wing forces and moments into the vehicle dynamics. The effect of lift, drag and viscous moment on the vehicle are denoted as L, D and M_{DL} respectively. While α is the angle of attack that will be explained further in chapter 6 where we expand on the hydrodynamic forces from a foil. The paper then continues to explain how they approximated these hydrodynamic forces using airfoil theory and compared them to experimental observations.

5.5 Control and motion planning strategies for Gliders

When reviewing literature on the topic of control for conventional underwater gliders most seem to follow the trend seen in Graver and Leonard (2001). There they start by

determining a desired glide path angle and a desired speed, and by studying equilibrium conditions one obtains desired values for angle of attack. The task for controller design is then to maintain these desired conditions. A review paper that documents the performance of different control strategies can be found in Ullah et al. (2015).

Finally we note that the AIAUV does not have a buoyancy engine or the ability to change its mass distribution, so the methods constructed for control of underwater gliders are not immediately available for the AIAUV. The AIAUV on the other hand is equipped with many thrusters that could be utilized for adjusting attitude.

On the topic on energy conserving motion planning for underwater gliders we have little to show. The paper Chyba et al. (2001) that considers time-optimal paths shows promise. An effort in generating optimal trajectories for underwater gliders in the presence of realistic ocean flows on a large scale is performed in Inanc et al. (2005).

5.6 Main takeaways for equipping an AIAUV with wings

It seems that conventional underwater glider gain their energy-efficiency from creating lift-forces at low speed through non-neutral buoyancy and moving masses for attitude adjustment. The lift-forces are generated by a non-zero angle of attack implying a non-zero angle between vehicle attitude and velocity that creates the characteristic sawtooth motion pattern.

We have seen examples of how to model the hydrodynamic effects of wings in a fashion that is suitable for a dynamic models used for system design and analysis as well as controller design. The description of lift and drag forces seen in this study seems like a good candidate for modeling the wings on an AIAUV.

The field of research on energy-optimal control for underwater gliders seems to be young but active. Decent real-world performance has been achieved with simple PID or LQR control, which suggests that this can be a valid starting point for the augmented AIAUV control architecture.

A question that we asked ourselves during work on this thesis was whether the

energy efficiency of gliders mostly stemmed from the proper utilisation of wings or if the main cause was their low operational speed. To this end we read much of the work of Professor Ralf Bachmayer at MARUM, and his colleagues. In particular this article on the development of an efficient propeller for the SLOCUM glider as found in Claus et al. (2010) was especially relevant. Based on his research we conclude that the low operational speed is a significant contributing factor for low energy demands of the system. The cause being that the system experinces low drag at low speeds.

Chapter 6

Hydrodynamic modeling of wings

The main purpose of this thesis is to evaluate the possibility of saving energy by augmenting the AIAUV with wings. As such there is a need to develop a model of a wing and the hydrodynamic forces that it impose on the AIAUV. The emphasis of this wing model will be on developing a model that can easily be incorporated into the existing AIAUV simulation model. Since the focus of this thesis is to reduce power consumption, its also necessary to ensure that the model is adequately realistic. This has been a challenging balancing act, and the development of the wing model and the results built upon it will be accompanied by a thorough evaluation of its limitations. The wings will hereafter be referred to as foils to conform with marine literature.

This chapter and the development of the hydrodynamic model for a wing or hydrofoil was originally written as part of the preproject and is included for completeness.

6.1 Anatomy of foils

In order to properly discuss a foil a brief introduction to air- and hydro-foil nomenclature is summarized below. Figure 6.1 shows a depiction of common geometric measures used when describing a typical wing.

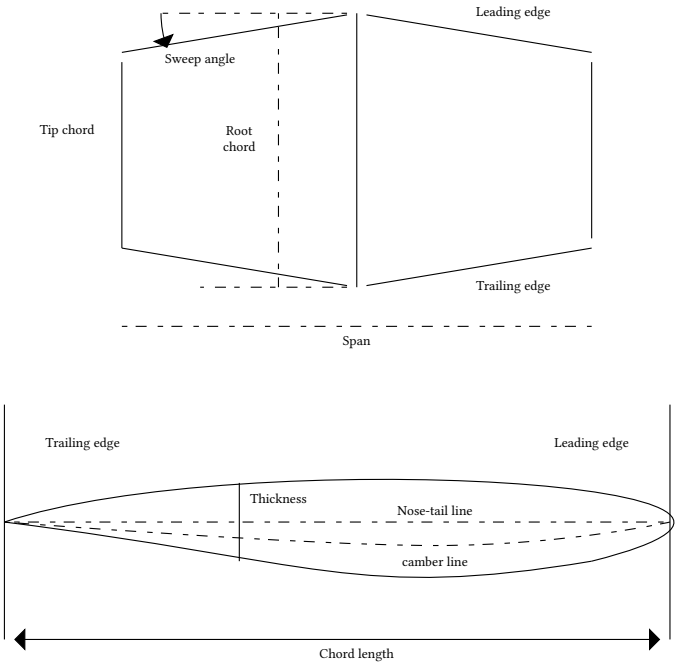

Figure 6.1: Anatomy of a typical wing or foil.

- **Hydrofoil:** A smooth, relatively thin, surface designed to create lift by generating suction on one side and pressure on the other by diverting different amount of inflowing fluid to its upper and lower side. See figure 6.1. Used to denote both two-dimensional theory of wings of infinite length and for wings of finite-length.
- **Leading edge:** The rounded front of the foil, often called the nose.

- **Trailing edge:** The sharp end at the aft of the foil, often called the tail.
- **Chord length:** The distance between the center of minimum radius of curvature of the leading edge and the sharp trailing edge. Denoted c.
- **Root Chord:** The chord length at the base of the foil where it is connected to the hull.
- **Tip Chord:** The chord length at the tip of the foil.
- **Taper ratio:** The ratio between the root chord and the tip chord.
- **Wing span:** The length from the root to the tip of the foil. Denoted s.
- **Nose-tail line:** The line between the leading and trailing edge. Sometimes also referred to as just the chord.
- **Camber:** The distance between the nose-tail line and the mean line between the upper and lower surfaces of a foil.
- **Angle of attack:** The angle between the nose-tail line and the direction of inflowing fluid, denoted α. The direction of positive rotation defined such that lift points in the direction of the suction side of the foil.
- **Lift:** A force generated by the pressure differences between the lower and upper surface. Its direction is towards the suction side of the foil and orthogonal to the inflow velocity.
- **Drag:** A drag force arising due to surface effects on the foil, acting in the same direction as the inflow velocity.
- **Planform area:** The projected area in the direction of the lift force at zero angle of attack, denoted A.
- **Aspect ratio:** A measure of a foils two-dimensionality. A high aspect ratio implies that the span of the foil is much larger relative its chord so that two-dimensional approximations are valid. Denoted $\Lambda = \frac{s^2}{A}$

- **Sweep:** The angle that determines how perpendicular the foil is relative the hull in the surge direction. Unswept foils are affixed at a right-angle to the hull.

6.2 Purpose of model and impact on design choices

As the emphasis of this paper is to develop a foil model that can easily be incorporated into the existing AIAUV model, and further used to evaluate control strategies and motion planning schemes we chose to focus on a sparse model that can represent loads using already available paramaters. The alternative is to form a more complex model either from experimental data or from explicitly simulating the fluid volume surrounding the AIAUV.

With this in mind, we develop a methodology that is able to approximate foil loads, (lift and drag), calculated only from angle of attack, inflow velocity and physical constants, all of which are readily available during simulation of the AIAUV. The model is based on 2D-theory for a foil of infinite span and then extended to the case of a foil with finite span, under some restricting assumptions. The development of this model stems from Faltinsen (2006) and relevant expressions are summarized below. We present the development for a symmetric, uncambered, unswept foil in a steady flow field valid for angle of attack $\|\alpha\| < 20$ degrees.

6.3 Loads on a foil

The loads generated by a foil submerged in a moving fluid can be sufficiently described as a lift orthogonal to the inflow direction and towards the suction side of the foil and a drag aligned with the inflow velocity. The lift, L, and drag, D depends on the density of the fluid, ρ, the planform area of the foil, A, and the magnitude of inflow velocity, V. The explicit expression for both forces are shown in equation 6.1, where $C_L(\alpha)$ and $C_D(\alpha)$ denote the coefficient of lift and drag respectively as a function of the angle of attack, α.

$$L = \frac{1}{2}\rho C_L(\alpha) A V^2 \tag{6.1a}$$

$$D = \frac{1}{2}\rho C_D(\alpha) A V^2 \tag{6.1b}$$

6.4 Foil theory

The aim of this section is to present the conceptual steps in how the marine community developed their approximations that are presented later in this paper. A more thorough presentation of this theory can be found in Faltinsen (2006). The theory is developed for a wing of infinite span surrounded by a 2D flow field of an infinite fluid, which is later adapted into the case of a wing with finite span. By starting with the Navier-Stokes equations and admitting some simplifying assumptions one can derive a description of the velocity potential for the 2D flow surrounding the foil. From the velocity potential one can describe the pressure distribution by means of Bernoulli's equation. To determine the net force on the foil one can then integrate this pressure distribution.

The end result is theory that can be used to model for example the lifting force for a flat plate of infinite span which will satisfy the equation in 6.1 with characteristic area simply as the chord length, $A = c$ and the lifting coefficient given in 6.2. Note that this theory ignores viscous effect so that the drag will be zero.

$$C_L(\alpha) = 2\pi\alpha \tag{6.2}$$

Remark. *This has been just a brief guide through the conceptual steps of the method, a more detailed derivation is beyond the scope of this work. Please consult Faltinsen (2006) for a more detailed argument on this derivation and a fluid mechanics textbook such as Cengel and Cimbala (2013) for an introduction to both the Navier-Stokes equations and Bernoulli's equation.*

6.5 Extension to wings of finite span

The processes of extending the approximation of lifting coefficient $C_L(\alpha)$ to the case of finite span involves determining the "two-dimensionality" of the foil in question. With the term two-dimensionality we mean how much of the foil can be considered to

be in conditions satisfying the ideal flow conditions for the 2D-case. Since the lifting coefficient is an approximation based on forces arising from pressure differences in a fully developed steady flow we need to investigate effects occuring at particular parts of the foil.

One of these effects is the loss of pressure difference at the wing tip. At the tip of the foil tip vortices will develop as there is nothing stopping the flow from the pressure side escaping over the foil towards the suction side. This leads to a loss of lift and flow conditions that are no longer in correspondence with ideal theory. Towards the other side of the foil where it is affixed to the vehicle hull there will also be a loss of lift due to flow differing from 2D theory.

To account for the situations where flow does not conform to the 2D theory one can utilize the aspect ratio, Λ, as a measure of two-dimensionality and augment the lifting coefficient, C_L, to take this into account. Søding's formula shown in equation 6.3 has proven to have good correspondence with experimental results for foils of finite length and low aspect ratio. Some discussion around Søding's formula can be found in Faltinsen (2006).

$$C_L(\alpha) = \frac{\Lambda(\Lambda + 1)}{(\Lambda + 2)^2} 2\pi\alpha \tag{6.3}$$

The development of drag coefficient $C_D(\alpha)$ for a foil of finite span is derived from the principle of conservation of fluid momentum and can be adapted to take the aspect ratio into account in a similar fashion to C_L. A formula for the drag coefficient for an uncambered foil of finite length and elliptical planform is shown in equation 6.4, also stemming from Faltinsen (2006).

$$C_D(\alpha) = \frac{4\pi\alpha^2\Lambda}{(\Lambda + 2)^2} \tag{6.4}$$

6.6 Limitations of model

One very important limitation of these approximations are that they are only valid for small angles of attack. For $\|\alpha\| > 20$ degrees the approximations diverge from reality

as they do not account for the stalling effects that arise at the foil and the sudden loss of lift.

The approximations above can be extended further to account for more complex fluid flows and their effects on the foil.

- The induced drag stemming from the tip-vortices can be included. The marine community have also developed approximations of the damping effect that a foil exhibit in unsteady-flows, resisting motions in heave and pitch.
- One can also adapt the model to account for non-uniform inflows as would happen if for example there was a propeller or another foil in front of the foil.
- There are better and more realistic models for the drag some of which are documented in Faltinsen (2006). Emphasis should be put on this if one is to quantify energy-savings.
- The model assumes that the ambient pressure around the foil is high enough that cavitation does not occur. This assumptions seems reasonable as long as the AIAUV operates far enough from the surface.
- The boundary effect near the hull and how it reduces the effective planform area could be estimated.

Overall we feel that this somewhat simple model for foil loads is sufficient from a control design and system analysis standpoint, as they give reasonable estimates of the loads. The more complex models built either on experimental data or through computational fluid simulation software will however be necessary if one wish to properly quantify the forces, or understand their properties in more complicated flow conditions.

Chapter 7

AIAUV Model extension

The work on extending the AIAUV dynamics to incorporate hydrodynamic forces from wings was started during the preproject. However, we have refined the method and framework to support multiple wings with individual parameters as well as introducing an explicit wing-frame allowing for individually trimmed wings. The foundation for the model extension remains the same as in the preproject and is detailed in Section 7.1. The rest of this chapter has been written as part of work on the Master's project.

7.1 Incorporating a foil into dynamics

As shown in chapter 6 the loads generated by a foil can be modeled as a lift and drag force. These forces can then be collected in a wrench vector expressed in the link-frame, $\mathcal{F}_i$, of a link in the AIAUV model. The dynamics of one link of the AIAUV can be modeled with a vectorial representaion of the Kirchhoff equations as seen in chapter 3, eq. 3.11. The addition of a foil will give an additional wrench, $\tau_{i,foil}$ in the Kirchhoff equations, as shown below in equation 7.1.

$$M_i \dot{\nu}^i_{i,0} + \Omega_i(\nu^i_{i,0})\nu^i_{i,0} + D_i(\nu^i_{i,0})\nu^i_{i,0} + b_i(A^0_i) + \tau_{i,foil} = \tau_i \tag{7.1}$$

The foil-wrench is expressed in the link-coordinates of link i and we can utilize the Jacobian transformations defined in chapter 3, equation 3.7b, to transform it into the general joint velocities, ζ. The transformation of the link-wrench, $\tau_{i,foil}$ into its representation in the joint coordinates, denoted η_{foil}, is shown in equation 7.2.

$$\eta_{foil} = J_i^{\mathrm{T}}(\vartheta)\tau_{i,foil}\,, \quad \eta_{foil} \in \mathbb{R}^c \tag{7.2}$$

The new, augmented dynamics of the AIAUV with a foil can then be represented in joint coordinates simply by the addition of this new term η_{foil} as shown in equation 7.3.

$$H(\vartheta)\dot{\zeta} + C(\vartheta,\zeta)\zeta + F(\vartheta,\zeta)\zeta + g(\vartheta) + \eta_{foil} = \mathbf{B}(\vartheta)\mathbf{u} \tag{7.3}$$

7.2 Foil frame

The first step is to introduce a reference frame rigidly fixed to the foil, hereafter denoted the foil frame, $\mathcal{F}_f$, and relate this to the reference frame of the link that the foil is attached to. The origin of the foil frame will be at the foils centre of pressure, c_p, and rotated relative the link frame through a simple rotation around the y-axis. This rotation represents the trim angle of the foil which we denote δ.

Remark. *We assume that the pressure center of the wing is at a fixed location on the wing such that the connection between the link frame and wing frame is static. This assumption does not correspond with reality as the pressure center will move with changing velocity and angle of attack. However, we deemed the assumption necessary as it allows us to have a constant homogenous transform between wing and link.*

The x-axis of the wing frame passes through the chord-line of the foil towards the leading edge. The y-axis is parallel to the y-axis of the parent link frame. Finally, the z-axis is such that it completes a right-handed Cartesian coordinate system. The foil frame and link frame is visualized in Figure 7.1. The pose of the foil frame relative its parent link frame is given by the static homogenous transform, H_f^i, given in Equation

7.4, where $\mathbf{R}_y(\delta)$ is the rotation matrix representing a simple rotation around the y-axis through the angle δ.

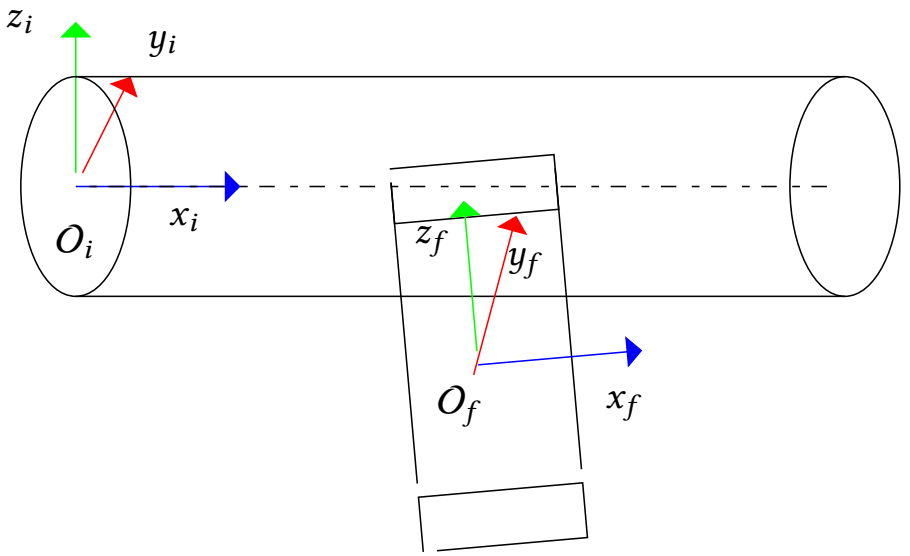

Figure 7.1: A foil mounted on a link with accompanying reference frames for the link and foil.

$$H_f^i = \begin{bmatrix} \mathbf{R}_y(\delta) & c_p^i \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{7.4}$$

Remark. *This method could be extended to actuated lifting surfaces by having a non-constant homogenous transform between link and foil frame. Adding a foil with sweep or one not mounted orthogonal to the link hull can also be accomplished by keeping the y-axis of the foil on the tip-to-root line and altering the homogenous transform, H_f^i, accordingly. Finally, if the location of the centre of pressure is known through more a sophisticated foil model this could also be incorporated as a parameter of $H_f^i(c_p)$.*

7.3 Foil forces in the foil frame

The theory in Chapter 6 expresses the loads on a foil as two forces, lift and drag, acting through the centre of pressure. The magnitude of the forces can be viewed as functions of angle of attack and inflow velocity, $L(V, \alpha)$ and $D(V, \alpha)$, while their direction is defined implicitly relative the direction of inflowing fluid, with lift always acting perpendicular to the inflowing velocity and drag always has a direction that is aligned with the inflow.

We would instead prefer to describe the forces as a function of link velocity and expressed in our wing-frame. First off, as we have neglected ocean current in the scope of this work we can state that the inflow velocity experienced by a link is the opposite of the linear speed of the link. More specifically we can state that the magnitude of the link velocity, $\left\| v^i_{i,0} \right\|$, equals the magnitude of the inflow velocity, V. Furthermore we can express the link velocity in the wing frame through a simple rotation as long as our foil frame satisfies the constraints in Section 7.2. The transformation is shown in Equation 7.5, resulting in the foil velocity $v^f_{f,0}$.

$$v^f_{f,0} = v^f_{i,0} = R_y(trim) v^i_{i,0} \tag{7.5}$$

When the foil velocity is known we can use its surge and heave components to determine the angle of attack, α. We define α according to Equation 7.6 to produce lift and drag forces with the right signage and ensure that lift points towards the suction side of the wing.

$$\alpha = -\texttt{atan2}(w^f_{f,0}, u^f_{f,0}) \tag{7.6}$$

Using the expressions developed in Chapter 6, Equation 6.1 we compute the lift, L, and drag, D, that the wings generate. However their direction is defined implicitly. In order to make the direction of the total force generated by the wing clearer we project it into the basis of the foil frame through the relation in Equation 7.7. The two new quantities N and A are the normal and axial components of the sum of lift and drag generated by a wing. The orientation of these forces are defined such that positive

N aligns with the positive z-axis of the foil frame, while positive A aligns with the positive x-axis of the foil frame.

$$N = L\cos(\alpha) + D\sin(\alpha) \tag{7.7a}$$

$$A = L\sin(\alpha) - D\cos(\alpha) \tag{7.7b}$$

7.4 Foil force in link frame

We can collect the normal and axial foil force components into the total foil force, F_f and express its direction and magnitude in both the foil frame and link frame through the expressions in Equation 7.8. Where F_f^f and F_f^i is the total foil force expressed in the basis of the foil frame and link frame respectively.

$$F_f^f = \begin{bmatrix} A \\ 0 \\ N \end{bmatrix} \in \mathbb{R}^3 \tag{7.8a}$$

$$F_f^i = R_y^{\mathrm{T}}(\delta) \tag{7.8b}$$

At last, we can construct the wrench that the foil enacts on the link. With the foil force now properly expressed in the link frame we can construct the foil wrench $\tau_{i,foil}$ according to Equation 7.9. Note the negative signs as the wrench appear on the left hand side in the link dynamics. The wrench, $\tau_{i,foil}$ encompass the linear forces and moments that the wing enact on the link frame. This wrench can now be propagated to the whole state-space of the AIAUV with the transformation in Equation 7.2.

$$\tau_{i,foil} = \begin{bmatrix} -F_f^i \\ -c_p^i \times F_f^i \end{bmatrix} \in \mathbb{R}^6 \tag{7.9}$$

7.5 Support for multiple wings

The computations for wing wrenches detailed in Sections 7.2-7.4 can be performed repeatedly for different foils. This allows us to equip the AIAUV with multiple wings either at different links, or multiple wings at the same link, or any combination of those. The framework also supports placing the links at different locations, and orientations, as long as the location of pressure center and trim angle is specified for each wing. We can also support different wing geometries or hydrodynamic coefficients by changing the expressions that compute L and D without need for any modification of the framework itself.

When the wing wrench for each individual wing j, has been transformed to its respective $\eta_{j,foil}$ by Equation 7.2 we can simply add their contributions together as shown in Equation 7.10. Where k is the total number of wings, and $\tau_{j,foil}$ is the wrench generated by foil j. The resulting $\eta_{foil,tot}$ can be added to the ζ-dynamics in Equation 7.3 where it replaces η_{foil}.

$$\eta_{foil,tot} = \sum_{j=1}^{k} \tau_{j,foil} \tag{7.10}$$

Chapter 8

Energy metric

There is a need to develop a metric of energy-use in order to discuss the effect of wings on locomotion-efficiency of the AIAUV. This chapter develops such a metric. The energy required to locomote the AIAUV is supplied by two types of actuators, proppeler driven thrusters and motors driving the rotary joints.

The thrusters are modelled as a force, $F_i, [N]$ and the joint motors as a torque $\tau_i, [Nm]$. Both type of acutators have trivial dynamics and can immediately produce the commanded output. The metric is constructed by finding the instantaneous power each actuator delivers to the system and summing the contributions together to find the total instantaneous power delivered to the AIAUV. This total instantaneous power at each time of the simulation can then be integrated over the whole simulation-time to find the total energy use for a specific AIAUV configuration performing a specific motion, enabling comparisons across different robot configurations and motions.

8.1 Instantaneous power from actuators

Lets first develop an expression for the instantaneous power delivered from the joints. The method we develop in this section is adapted from Kelasidi et al. (2018) where it was used to compute the average power consumption of a snake robot. The simulation

model presented in Chapter 3 presents us with a minimal representation of the AIAUV dynamics in the form of the ζ coordinates. ζ consists of the linear and angular velocities of one base-link, $\nu^1_{1,0}$, and the angular velocity of all joints, $\dot{q}_i$, shown in Equation 8.1.

$$\zeta = \begin{bmatrix} \nu^1_{1,0} & \dot{q}_1 & \dots & \dot{q}_i & \dots & \dot{q}_m \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{6+m} \tag{8.1}$$

The instantaneous power supplied by joint i at time t is then given by Equation 8.2 and denoted $P_{\tau,i}(t), [W]$. Note that this computation can result in negative instantaneous power when the torque and joint velocity have opposite signage. As detailed in Kelasidi et al. (2018) from simulations of underwater snake robots they experience that the system can recover some energy due to this negative work effect. Based on this we opt to include the effect in the power computations for the joints of the AIAUV as it share some characteristics with underwater snake robots.

$$P_{\tau,i}(t) = \tau_i(t)\dot{q}_i(t) \tag{8.2}$$

Now to develop an expression for the instantaneous power that the thrusters deliver to the system. First, recall that the body velocity of a link i can be recovered from ζ through its associated Jacobian J_i:

$$\nu^i_{i,0} = J_i\zeta, \quad \nu^i_{i,0} = \begin{bmatrix} v^i_{i,0} & \omega^i_{i,0} \end{bmatrix}^{\mathrm{T}} \quad v^i_{i,0}, \omega^i_{i,0} \in \mathbb{R}^3 \tag{8.3}$$

We propose that the instantaneous power delivered from a thruster j mounted on link i can be calculated by the expression in Equation 8.4. Where $\tau^i_j \in \mathbb{R}^6$ is the wrench that thruster j acts on link i with.

$$P_{T,j}(t) = |\tau^i_j(t) \cdot \nu^i_{i,0}(t)| = |\tau^i_j(t)^{\mathrm{T}}\nu^i_{i,0}(t)| \tag{8.4}$$

Note the absolute value in Equation 8.4 causing the power consumption of thrusters to always be positive or zero, never negative. Contrary to the rotary joints we do not see any evidence of a negative work effect for thrusters as there is no way for them to recover energy.

8.2 Total energy expenditure

With the power off all individual actuators accounted for one can calculate the total instantaneous power delivered to the system, $P_{tot}(t)$, by simple summation as in Equation 8.5. We can also separate the total joint power and thruster power into $P_{\tau,tot}(t)$ and $P_{T,tot}(t)$ respectively.

$$P_{tot}(t) = \sum P_{\tau,i}(t) + \sum P_{T,i}(t) = P_{\tau,tot}(t) + P_{T,tot}(t) \tag{8.5}$$

By integrating $P_{tot}(t)$ over the whole simulation time, T, one can find the total energy expenditure of the AIAUV. This integration is performed using simple Euler integration where Δt is the size of the time-step used for sampling the simulation data. The expression for total energy expenditure, $E_{tot}(t), [J]$ is found in Equation 8.6. Alternatively one can integrate the total joint and thruster power individually to separate the total energy into the total energy consumption of all joints, E_τ, and the total energy consumption of all thrusters, E_T.

$$E_{tot}(t) = \int_0^T P_{tot}(t)dt = \sum_{k=0}^{T/\Delta T} P_{tot}(k)\Delta t \tag{8.6}$$

Chapter 9

Control strategies for AIAUV's

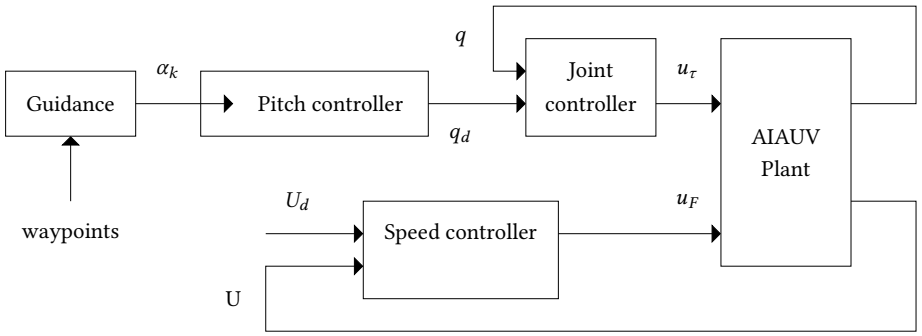

Figure 9.1: Block Diagram of the control hierarchy for the AIAUV

In order to investigate the energy-saving properties of the addition of lifting surfaces to the AIAUV we propose the following control hierarchy as seen in Figure 9.1. The control structure consists of a guidance law for waypoint tracking in the vertical plane, a pitch-controller for adjusting pitch by changing the shape of the AIAUV, a speed controller for the thruster, and finally a joint controller for tracking the joint references generated by the pitch controller.

The control hierarchy enables the AIAUV configurations detailed in Chapter 12 to track waypoints in the vertical plane, a possible real-life mission where this capability would be necessary is for example long-range oceanographic surveying. In these missions one wish to perform oceanographic measurements at different depths over a long distance.

The guidance block uses a Line of Sight based steering law for computing the desired pitch of the AIAUV. The desired pitch angle is then achieved by the pitch controller that generates joint references that curve the AIAUV with an decreasing deflection from tail to head. The tracking of joint angles to their reference value is achieved by a lower level joint controller. Finally a separate part of the control hierarchy called the speed controller controls the forward speed of the AIAUV with the thruster.

In the following sections we describe the inner workings of each of these control structures.

9.1 Guidance

The guidance law is a lookahead-based Line of Sight approach originally meant for surface-vessels adapted from Fossen (2011). The guidance law is originally meant for a surface vessel and derived in a North-East reference frame. This section will present a converted expression of the guidance law that can be used for our AIAUV and the X-Z plane of its inertia frame.

9.1.1 Waypoint selection

A waypoint $\mathbf{p}_k$ represents a fixed point in space that we wish the AIAUV will visit during transit. A waypoint is represented as the distance from the origin of the inertia frame, $\mathbf{p_k} = [x_k^0, y_k^0, z_k^0]^\mathrm{T} \in \mathbb{R}^3$. A mission for the AIAUV can be specified by a ordered list of waypoints and its the waypoint selection algorithm's job to decide what waypoint to travel towards next.

The chosen method is a circle of acceptance approach adapted from Fossen (2011).

Let each waypoint $\mathbf{p}_j$ be accompanied by a parameter R_j. This extra parameter R_j represents the radius of a circle with origin located at waypoint $\mathbf{p}_j$. When the AIAUV's position is within this circe of acceptance we deem that we are sufficiently close to the waypoint and that we can switch to transit towards the next waypoint. Let $\mathbf{p}_k$ represent the last visited waypoint, $\mathbf{p}_{k+1}$ represent the current waypoint we are traveling towards and $\mathbf{p}_{k+2}$ be the one we wish to visit after that. The waypoint selection algorithm for updating the current and next waypoint is given in Equation 9.1. When the relation in Equation 9.1 is satisfied the algorithm switches the current waypoint (the one we last visited) from $\mathbf{p}_k$ to $\mathbf{p}_{k+1}$ and the next waypoint (the one we are traveling towards) from $\mathbf{p}_{k+1}$ to $\mathbf{p}_{k+2}$.

$$[x_{k+1} - x(t)]^2 + [z_{k+1} - z(t)]^2 \leq R_{k+1}^2 \tag{9.1}$$

9.1.2 Waypoint frame

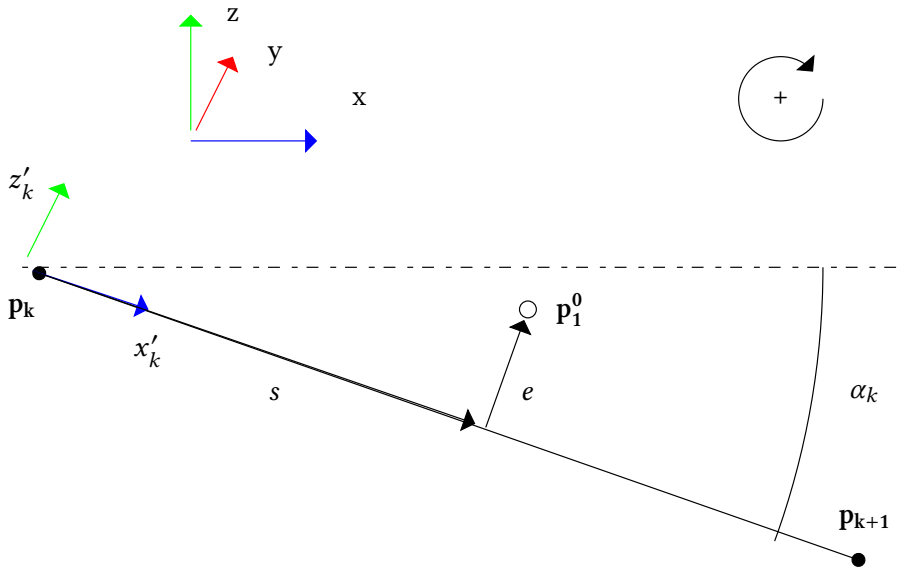

Figure 9.2: The reference frames and coordinates used in the Guidance Law

According to Figure 9.2 let, $\mathbf{p}^0$ be the position of the base link of the AIAUV, and $\mathbf{p}^0_k$ and $\mathbf{p}^0_{k+1}$ be the current and next waypoint respectively. Let α_k be the angle between the x-axis of the inertia frame and the line segment connecting the two waypoints, defined according to Equation 9.2. Here `atan2(z,x)` is the four-quadrant inverse tangent function that maps onto the closed interval $[-\pi, \pi]$. The minus in Equation 9.2 is necessary so that α_k corresponds to a rotation around the y-axis of the inertia frame, in Figure 9.2 the y-axis points into the paper to complete a right-handed reference frame.

$$\alpha_k = -\texttt{atan2}(z_{k+1} - z_k, x_{k+1} - x_k) \tag{9.2}$$

The next step is to transform the positions of the AIAUV and next waypoint into a reference frame with origin in the current waypoint, $\mathbf{p}_k$, and x-axis aligned with the line connecting $\mathbf{p}_k$ to $\mathbf{p}_{k+1}$. This frame will be called the guidance frame for waypoint $\mathbf{p}_k$ and the AIAUV position expressed in this frame will be denoted $\epsilon(t) = [s(t)e(t)]^{\mathrm{T}}$. The transformation is shown in Equation 9.3, and $\mathbf{R}_y(\alpha_k)$ is the 2×2 rotation matrix given by Equation 9.4.

$$\epsilon(\mathbf{t}) = \begin{bmatrix} s(t) \\ e(t) \end{bmatrix} = \mathbf{R}_y(\alpha_k)(\mathbf{p}^0(t) - \mathbf{p}^0_k) \tag{9.3}$$

$$\mathbf{R}_y(\alpha_k) = \begin{bmatrix} \cos\alpha_k & -\sin\alpha_k \\ \sin\alpha_k & \cos\alpha_k \end{bmatrix} \tag{9.4}$$

The coordinates $s(t)$ and $e(t)$ are denoted the along-track distance and cross-track error respectively. The desired behaviour of the AIAUV for following approaching the next waypoint can thus be described by a monotonically increasing along-track distance, $s(t)$, and a cross-track error, $e(t)$, that approaches zero.

9.1.3 Guidance Law

The objective of the guidance law is to designate a pitch-angle for the AIAUV to ensure that it reaches its destination at the next waypoint. The approach chosen here was to adapt a lookahead based guidance law for surface vessels from Fossen (2011). As the AIAUV consists of multiple links, each having their own pitch angle relative the inertia frame pitch does not have a unique meaning for this class of robot. For now, lets assume that the mean pitch of the AIAUV is the one we wish to designate in order to steer the AIAUV towards the next waypoint. We denote the mean pitch as $\bar{\theta}$ and the desired mean pitch as $\bar{\theta}_d$, how to get the vehicle to track this quantity will be detailed in the Section 9.2.

The objective of the guidance law is therefore to compute $\bar{\theta}_d$ to ensure convergence to the straight-line path connecting the current and next waypoints. To this end we utilize the Guidance frame developed in the previous section. In guidance-frame coordinates adhering to the path is achieved by increasing $s(t)$ while forcing $e(t)$ to zero before we are within the circle of acceptance for the next waypoint. To this end we propose setting a desired pitch-angle as given in Equation 9.5a.

$$\bar{\theta}_d(e) = \theta_p - \theta_r(e) \tag{9.5a}$$

$$\theta_p = \alpha_k \tag{9.5b}$$

$$\theta_r(e) = \tan^{-1}\left(\frac{-e}{\Delta}\right) \tag{9.5c}$$

As seen in Equation 9.5a-9.5c, $\bar{\theta}_d$ consists of two terms. The first term, θ_p, is simply the angle between x-axis of the inertia frame and the line segment connecting the current and next waypoints. The second term, $\theta_r(e)$, adjusts the course in pitch according to the cross-track error, e, and the lookahead distance, Δ. According to Fossen (2011) the term $\theta_r(e)$ in Equation 9.5c can be viewed as a saturating control law with gain $K_p = 1/\Delta$.

9.2 Pitch controller

The Pitch controller is responsible for generating feasible joint angles that will pitch the AIAUV to achieve the desired pitch-angle, $\bar{\theta}_d$ from the guidance law. To achieve this we propose to adapt a heading controller from Sans-Muntadas et al. (2017). This heading controller was originally used for generating a yawing motion in the $x-y$ plane and following a desired yaw-angle from a guidance law used for docking the AIAUV. We will instead be utilising it to perform a pitching motion with the AIAUV, and the desired pitch-angle will be $\bar{\theta}_d$ from the guidance law presented in the previous section.

Let $q_k^*(t)$ be the desired joint angle for joint k on the AIAUV. As we are restricting our study to motions in the vertical plane only $q_k^*(t) = 0$ for all joints corresponding to joint angles that will yaw the AIAUV. All other joints corresponds to motions in pitch and will have the reference value given by Equation 9.6.

$$q_k^*(t) = \theta_0 \left[\frac{n-k}{n+1} \right] \tag{9.6}$$

Where θ_0 is a synthetic control variable created as the output of a PI controller of the error between mean pitch and desired mean pitch. The expression for θ_0 is shown in Equation 9.7, and the mean pitch of the AIAUV is calculated according to Equation 9.8 where $\theta_{i,0}^0$ is the pitch angle of link i relative the inertia frame.

$$\theta_0 = -K_{p,\theta_0}(\bar{\theta} - \bar{\theta}_d) - K_{I,\theta_0} \int (\bar{\theta} - \bar{\theta}_d) dt \tag{9.7}$$

$$\bar{\theta} = \frac{1}{n} \sum_{i=1}^{n} \theta_{i,0}^0 \tag{9.8}$$

9.3 Low-level Joint control

Finally the commanded actuation of the joints is computed by a simple, decoupled PD controller for each joint in order to track the joint-angle references generated by the pitch controller. Let u_k denote the commanded joint torque for rotary joint k and be

given by equation 9.9.

$$u_k = -K_{p,q}(q_k - q_k^*) - K_{d,q}\dot{q} \tag{9.9}$$

9.4 Speed controller

The speed controller is decoupled from the rest of the control system and is a simple PI surge speed controller from Fossen (2011). The task of the speed controller is to provide thrust minimize the error between desired and actual speed, U_d and U respectively. The controller is shown in Equation 9.10. Where u_T denotes the commanded thrust for the AIAUV's aft thruster.

$$u_T = -K_p(U - U_d) - K_I \int (U - U_d) \tag{9.10}$$

Now we have a choice as to what we characterise as the speed of the AIAUV. We chose to use the linear speed of the base-link of the AIAUV partly because it the linear velocities of the base link is already part of the state space of the AIAUV model, and partly because the thruster is attached to the base link. We define U according to Equation 9.11 as the Euclidean norm of the linear velocity of the base link.

$$U = ||v_{1,0}^0|| \tag{9.11}$$

Chapter 10

Searching for optimal parameters for winged AIAUV

There is a lot of design freedom when implementing the extended model from Chapter 7 and the control hierarchy from Chapter 9. As this is a new class of underwater robot not many guidelines for optimal parameters exist, and due to the complex nature of the system we instead opted for a series of tests designed to identify a decent choice of parameters.

The tests are designed to study the impact of wing placement, number of wings, and diving angle have on energy demands. To this end we propose to start with the unwinged AIAUV configuration from Chapter 12 and the controller-hierarchy from Chapter 9 as a base configuration for the comparisons.

Remark. *We acknowledge that this study does not rest on a rigorous foundation of theory on choosing optimal parameters, and a deeper study of system properties could be worth exploring in future work on the topic of a wing-augmented AIAUV.*

However, we wish that this cursory exploration into optimal parameters can act as a stepping stone in this process, and showcase the potential of this new platform.

We separate the tests into three main parts. First a section devoted to the effect

of varying the position of a single wing on the AIAUV body in Section 10.1. Second, a section investigating the addition of multiple wings to the AIAUV in Section 10.2. Thirdly an exploration into the effect of varying the pitch course-angle from the guidance-block in Section 10.3. Finally we conclude the chapter with a section summarising our findings and giving an overall recommendation for parameters. Discussions comparing energy demands of rotary joints and thrusters are also saved for the last section.

There are many ways to represent and compare system performance for these studies, but we have chosen to focus on only four metrics in this chapter: the along-track distance, $s(t)$, the cross-track distance, $e(t)$, the total energy-expenditure of thrusters, E_T, and the total energy-expenditure from the joint-motors, E_τ. The definition for the guidance-metrics, $s(t)$ and $e(t)$ are found in Chapter 9, while the energy-metrics, E_T and E_τ, are defined in Chapter 8. For $e(t)$ we plot its whole time-evolution, while for $s(t)$ we only show its final value, $s(T_f = 200)$, as it is monotonically increasing after the first waypoint is reached for all tests, and only the final value is of interest.

A more thorough analysis of the complete system and its performance can be found in Chapter 13 where we compare the same AIAUV with and without wings. The main purpose of this chapter is to determine what system parameters would be used in that later study.

Common for all simulations in this study are:

- The base-dynamics of the AIAUV remained the same and its parameters are those shown in Chapter 12.
- The hydrodynamic model for the wings are the same its parameters are those found in Chapter 12. The only thing we altered was what link it was attached to.
- The choice of numerical solver with an `ode4` Runge-Kutta method, the timestep used for simulation, $\delta t = 0.02$s,
- Starting and final simulation time, $T_0 = 0$s and $T_f = 200$s respectively.
- Initial conditions with the AIAUV starting at rest and with base position at $p_1^0 = \mathbf{0}$ and all joint angles $\mathbf{q} = 0$.

- In all experiments the AIAUV were tasked with keeping the same forward speed $U_d = 0.5\text{ms}^{-1}$.

- All AIAUV's were given a running start by including a waypoint at $\mathbf{p}_1^0 = [20, 0, 0]^{\text{T}}$ with circle of acceptance $R_1 = 4\text{m}$ in order to allow them to get up to speed before changing course in pitch.

- The controller parameters and gains were also kept constant during the studies. Their values are documented in Chapter 12, with the exception of $K_{I,U} = 0$ during tests in this chapter.

- No simulations had wings experience an angle of attack larger than 20° in order to satisfy the constraints of the wing model developed in Chapter 6.

The choice to not tune the controller parameters between the different simulations will impact the system performance and as such warrants a further explanation. The system was tuned to have a acceptable performance across all tests and although the performance will vary with tuning the difference is not large enough to change the trend we see when changing wing configuration or dive angle. And by keeping the controller parameters constant we more clearly observe the changing energy demands of the system.

We also opt to not show any figures from the hydrodynamic forces from the wings in this chapter. A more detailed discussion around the lift and drag from the wings are found in Chapter 13 were a winged AIAUV is compared directly to one without.

Common for all plots of $e(t)$ later in this chapter is a sharp dip from zero to a negative value. This dip is due to the guidance algorithm switching the active waypoint and signifies when the AIAUV is within the radius of acceptance of waypoint $\mathbf{p}_1^0 = [20, 0, 0]^{\text{T}}$ and starts it dive with commanded pitch angle α_k. Positive values α_k denotes a downwards dive while negative values denote an upwards rise, for more details see Chapter 9.

10.1 Study of wing placement

For the first study we vary the placement of a single wing along the body of the AIAUV. The desired course angle in pitch was kept constant at $\alpha_k = 30^\circ$ after reaching the first waypoint, signifying a dive downwards. We number the links as the first link being the tail of the AIAUV and the last being the head, and we also present the case of when the AIAUV has no wings attached. Note that wings are only placed at odd-numbered links as the even-numbered links are Cardan-joints. The results are summarized in Table 10.1 and Figure 10.1.

wing placement	E_T	E_τ	$s(T_f = 200)$
None	1106.91J	4.88J	69.79m
Link 1	1022.66J	5.22J	64.46m
Link 3	1107.85J	5.26J	70.11m
Link 5	1107.68J	5.91J	75.61m
Link 7	936.06J	3.43J	76.72m
Link 9	818.00J	2.95J	76.03m

Table 10.1: Performance metrics for the study of wing placement on AIAUV.

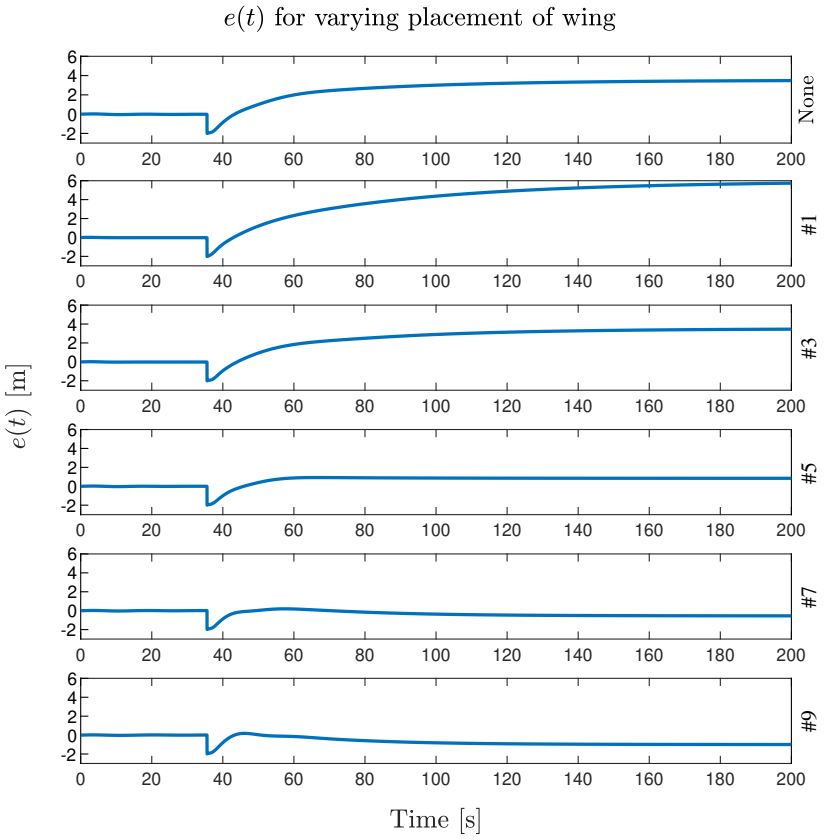

Figure 10.1: Cross-track errors for varying wing-placement on AIAUV. Label on the right of plot denotes what link the wing is connected to.

From Table 10.1 and Figure 10.1 we extract some general trends. Placing wings closer to the head is better than at the tail. In particular, placing wings at link #5, #7 and #9, greatly reduces the energy demand and increases the final along-track distance, $s(T_f)$. We also note that wings at the front greatly reduces the cross-track error

when compared to the rather poor performance in the case with no wings. The poor behaviour in the case with no wings, with a final cross-track error of $e(T_f) = 3.48m$ is further discussed in Chapter 13. Finally we note that a wing at the tail link alone, link #1, leads to a larger cross-track error and a shorter along-track distance even though it reduces the energy demand.

10.2 Study on number of wings

This section is devoted to study the effect of incorporating more than one wing to the AIAUV. Based on the data from the section above we start by introducing wings at the head of the AIAUV, link #9, before adding wings towards the tail of the AIAUV. Again we keep the dive angle, $\alpha_k = 30°$, constant between the different simulations. The result for this experiment are shown in Table 10.2, and Figure 10.2.

wing placement	E_T	E_τ	$s(T_f = 200)$
None	1106.91J	4.88J	69.79m
Link 9	818.00J	2.95J	76.03m
Link 7,9	824.26J	3.13J	76.92m
Link 5,7,9	820.45J	4.31J	77.00m
Link 3,5,7,9	735.26J	4.32J	74.77m
Link 1,3,5,7,9	648.52J	4.15J	71.93m

Table 10.2: Performance metrics for the study of number of multiple wings on AIAUV.

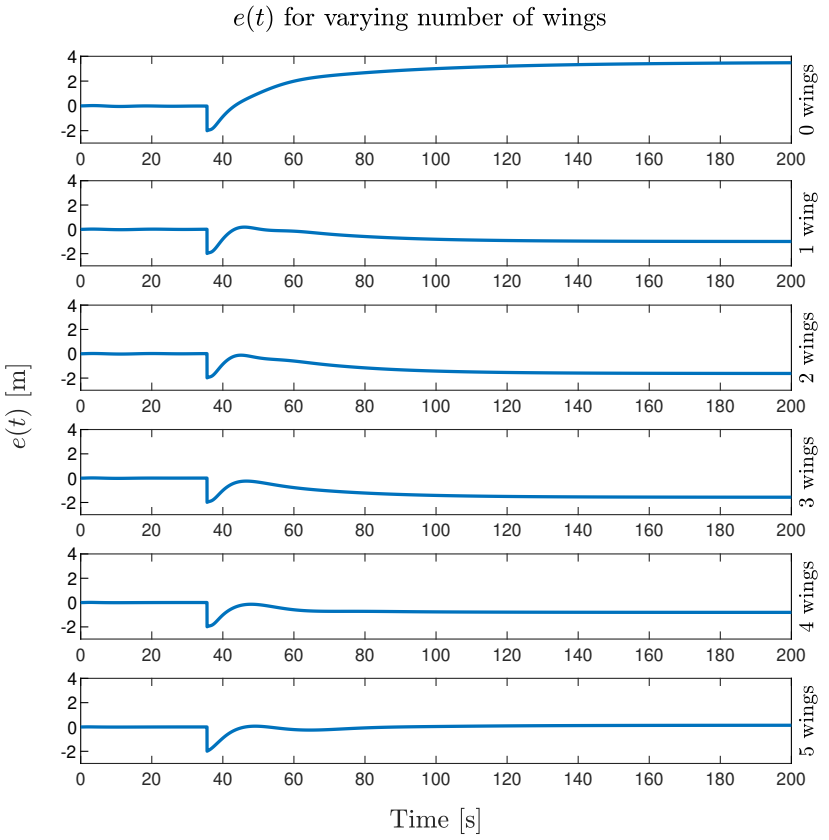

Figure 10.2: Cross-track errors for varying Number of wings AIAUV. Label on the right of plot specifies how many wings starting from head to tail.

From the data we see a clear drop in the needed energy of the aft thruster with E_T decreasing for each additional wing that is added to the AIAUV. The final along-track distance, $s(T_f)$, increases with the addition of the first wings, reaching a maximum with wings on links #5, #7 and #9 before dropping with wings on links after the

midline of the AIAUV, links #1 and #3. The addition of the first 3 wings also drive the final cross-track error to be more negative, while adding wings on link #1 and #3 restores it towards zero. With all five wings the final value is $e(T_f) = -0.14m$. Our recommendation would be the system with all five wings, one on each link, even though it has worse final along-track performance than other configurations, it has the lowest energy demands and the smallest cross-track error.

10.3 Study on diving angle

This section tests the system performance when the desired dive angle, α_k is changed. The AIAUV used is one equipped with 5 wings, one at each main link; link#1, #3, #5, #7 and #9. The purpose of this section is not to identify an optimal dive angle but instead to showcase how the system performs, as such only a small subset of dive angles are studied, starting at zero and increasing to 40° with a interval of 10°'s. The results are shown in Table 10.3 and Figure 10.3.

Remark. *Negative α_k's, signifying ascending motion is not tested in this section, for an example of negative α_k see the full simulation in Chapter 13. We do however remark that the results and overall trends are similar, as the change in system response is due to the mass-center and buoyancy center of each link not aligning, the effect of wings are symmetric in rising and diving motions.*

Dive angle	E_T	E_τ	$s(T_f = 200)$
$\alpha_k = 0°$	500.65J	0.00J	72.47m
$\alpha_k = 10°$	519.36J	0.53J	72.43m
$\alpha_k = 20°$	571.74J	2.13J	72.27m
$\alpha_k = 30°$	648.52J	4.15J	71.93m
$\alpha_k = 40°$	741.25J	6.30J	71.27m

Table 10.3: Performance metrics for varying commanded dive angles for AIAUV with wings on each link.

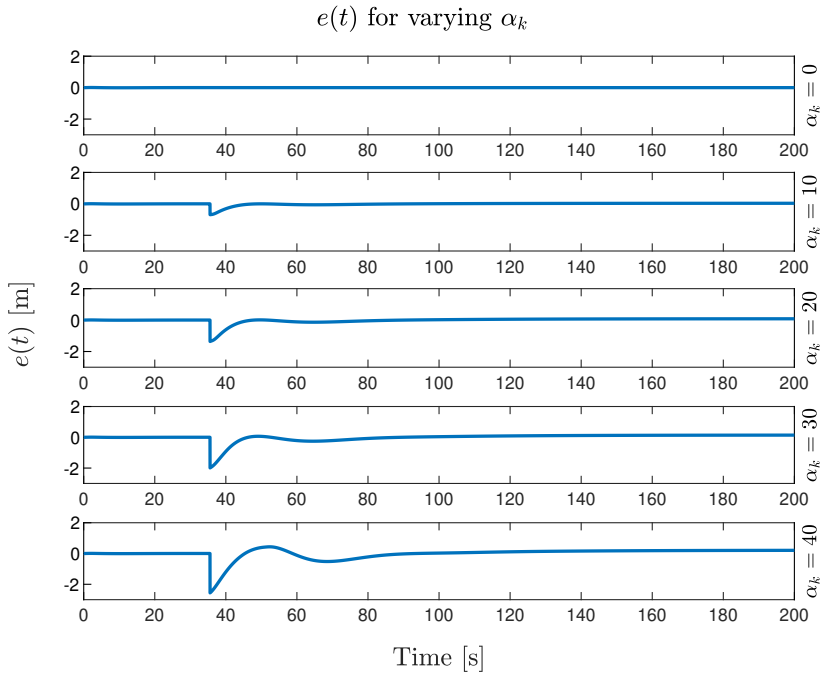

Figure 10.3: Cross-track errors for varying α_k for fully winged AIAUV. Label on the right of plot specifies what value of α_k it corresponds to.

The data from Table 10.3 Figure 10.3 showcases that the system with wings on every link is able to track the guidance objective, $e(t)$, with acceptable performance and without a too large decrease in the final along-track distance, $s(T_f)$. We also observe that a steeper dive requires more energy from the aft thruster than a shallow dive of comparable length. This is signified by E_T steadily increasing as α_k increases, and the traveled length remain relatively unchanged as $s(T_f)$ only decreases with 1.2m when performing a dive with $\alpha_k = 40°$ compared to $\alpha_k = 0°$.

The tests stopped with $\alpha_k = 40°$ as larger dive angles than that caused the wings to experience an angle of attack greater than 20°, violating their modeling-constraints.

We briefly note that the increase in energy demands and increasing angle of attack that we observed when increasing α_k is due to the way the AIAUV achieves its pitching motion. The AIAUV achieves its pitch by curving its body according to the motion pattern presented in Chapter 9. More crucially it has to maintain this curvature for the duration of the dive, and while that allows the wings to contribute more lifting forces to the system, it also increases the drag-profile of the AIAUV. This phenomenon will be further detailed in Chapter 13.

10.4 Overall discussion and recommendation

- Based on the results from the tests performed in the section above we would recommend an AIAUV with a wing at every rigid link. More wings reduce the energy demands for the thruster in all cases.
- Steeper dive- or rise-angles require more energy than ones with smaller $|\alpha_k|$.
- $|\alpha_k| > 40°$ makes the wing-model break its assumption of small angle of attacks, unmodeled stalling effects or drop in lift is likely to occur in these cases.
- In all studies the energy from thrusters dominate that of rotary joints by orders of magnitude.

Chapter 11

Description of simulation experiment

Towards the goal of determining the viability of saving energy with wings we propose the following experimental setup: Comparing the energy use between a winged AIAUV and a comparable unwinged AIAUV, when tracking waypoints in the vertical plane.

11.1 Inspiration

The inspiration for this experimental setup stems from conventional underwater gliders, in particular their mode of operation and their use-cases. As detailed in Chapter 5 conventional propeller-less underwater gliders need to travel in a sawtooth pattern in order for their wings to aid in their forward locomotion.

Another contributing factor is that one often wish to perform data-collection at different depths of the ocean when performing long-range oceanographic missions. So not only does the sawtooth pattern enable the wings to contribute to the overall forward locomotion of a winged AUV, but it reflects real-world use-cases.

11.2 Experimental setup

The main experiment consists of two individual simulations. One simulation for each of the two proposed AIAUV configurations in Chapter 12. The objective is to visit the same waypoints in each simulation. By keeping the waypoints the same one can isolate the effect the wings have on energy-demands for the AIAUV. The simulation will be performed with the same trivial initial conditions, for the same duration, and with the same time-resolution. Their values are listed in Table 11.1, where t_0 denote the initial simulation time, Δt the simulation time step, and t_f the final simulation time respectively. Note also that $[\eta_1^0(0), \epsilon_1^{0\,\mathrm{T}}(0)]^{\mathrm{T}} = [1, 0, 0, 0]$ is the zero quaternion.

Initial values
$p_1^{0\,\mathrm{T}}(0) = [0, 0, 0]$
$[\eta_1^0(0), \epsilon_1^{0\,\mathrm{T}}(0)]^{\mathrm{T}} = [1, 0, 0, 0]$
$q(0) = \mathbf{0}_{5times1}$
$\zeta(0) = \mathbf{0}_{14\times1}$
Time parameters
$t_0 = 0$ [s]
$\Delta t = 0.02$ [s]
$t_f = 900$ [s]

Table 11.1: Summary of initial values for simulation and time parameters

11.3 Waypoints and commanded speed

During our simulations we tasked both AIAUV's with maintaining the same forward speed, U_d, and navigating to the same list of waypoints. The commanded speed is shown in Equation 11.1, while Table 11.2 documents the waypoints that were used. Note that the first waypoint is located at the origin of the inertia frame and does not have a radius of acceptance associated with it. The first waypoint serves only as the anchor point for defining the desired pitch course angle, α_1 to the second waypoint, as explained in Chapter 9.

$$U_d = 0.5\text{ms}^{-1} \tag{11.1}$$

Waypoint #	**Location**, $\mathbf{p}_k^0$ [m]	**Radius** [m]
1	$[0, 0, 0]^T$	-
2	$[20, 0, 0]^T$	5
3	$[100, 0, -45]^T$	5
4	$[150, 0, -45]^T$	5
5	$[300, 0, 10]^T$	5

Table 11.2: List of waypoints used for simulations.

11.4 Implementation details

All of the various simulation have been performed in `MATLAB r2019a`. The starting point for the implementation of these simulations was a script that was capable of constructing the dynamics of an AIAUV with multiple links, and simulate them using `'MATLAB''s 'ode45'`-function as a numerical integrator to calculate the time-evolution of the system, and finally produce a video showing the AIAUV's motion.

We built upon and extended this framework in multiple ways in order to produce the results that are included in later chapters of this thesis. Some of the major contributions to the code-base is detailed below.

11.4.1 Building an experiment framework

We wrapped the existing code for simulating an AIAUV into a more fleshed out framework that can support simulation easy swapping of AIAUV link configuration, changing wing location, changing numerical solver, and swapping parts of the control architecture. This greatly reduced the iteration time, allowing us to test more solutions rapidly.

11.4.2 Controller framework

The calculations for commanded control inputs **u** documented in Chapter 9 are injected into the function describing the dynamics of the AIAUV. The control inputs are calculated with the same integration time-step as the numerical integration of the dynamics and are calculated with perfect measurements of the AIAUV state.

11.4.3 Current

The dynamics we were given had support for simulating a constant, irrotational, ocean current. The model we created for the wings are able to work under this same implementation of currents, but our controller architecture does not support it. As this is a first step in evaluating the effect of wings on an AIAUV we deemed it acceptable to include no current-effects in our simulations. As such, all simulations in this thesis are performed in perfectly still water.

11.4.4 Numerical solver

For initial development we used the built-in numerical integrator `'ode45'`, before experimenting with a built-in solver optimised for stiff mechanical systems with the `'ode23s'` solver. These numerical integrators both have self-adjusting, variable, step-size and perform function evaluation attempts forward and backwards in time which makes them easy to use and fast. However, they proved to be detrimental when comparing system energy between simulations of different AIAUV configurations and motions, as well as complicating the implementation of the integrating term in control laws.

In the end we chose to implement a fixed step Runge-Kutta numerical integrator of order 4, that we use for all simulations presented in this thesis. The method can be summarised by its Butcher tableau shown in Table 11.3. For more details on numerical solvers in general we refer to Egeland and Gravdahl (2002).

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

Table 11.3: Butcher tableau for the RK4 method used for simulating dynamics.

11.4.5 Datalogging and generating plots

The AIAUV is a fairly complicated mechanical system, consisting of a high-dimensional state-space and many signals to monitor if one wish to describe the system performance. The numerical solver will only give us the time-evolution of the state-space, so we created a framework for computing auxiliary information. The framework takes the results from the numerical integration of the dynamics and re-compute every step of the dynamics, enabling us to log quantities such as control-inputs, guidance-coordinates and wing forces. The framework is flexible and support logging of any data-type that is given an entry in the datalog structure.

11.4.6 Integrating term in controllers

As this framework is implemented solely in `MATLAB` and not in `SIMULINK` we could not use the existing tools for creating integrating terms in controllers. The solution we devised is to simply use Euler-integration on the error associated with each integrating term and store the accumulated error in its own global and persistent variable. This inevitably leads to messy code, where the logic responsible for handling the integrating terms are intertwined into large parts of the framework. The integrators also does not support any extra features such as saturation, resetting during simulation, or only accumulating small errors. Based on this we would not recommend this method if work on this topic is to continue beyond this thesis, and would instead recommend converting the framework to work in `SIMULINK` where more mature features for control exist.

Chapter 12

AIAUV configurations for experiment

This chapter documents the AIAUV configurations and model parameters that are used in the simulations in Chapter 10 and Chapter 13. The parameters listed in this chapter can be used together with the material in Chapter 3 and Chapter 4, that details the modeling of AIAUV- and link-dynamics respectively. The control parameters listed in this chapter correspond to the controller hierarchy developed in Chapter 9.

Remark. *We would like to remark that the only difference between the winged and unwinged AIAUV presented in this thesis is the addition of wings into the total AIAUV dynamics. The link configuration and their parameters as well as the controller gains remain the same. This choice was made to isolate the effect of wings on the system.*

We will also remark that not changing the controller parameters between the two configurations of the AIAUV could give an unfair advantage to one of the systems. During preliminary testing for this thesis we experienced relatively small changes in performance between the two systems from altering controller gains. We do however, see a difference in performance between a winged and unwinged system but attribute this to flaws in the controller architecture that can not be fixed by simply changing gains. This discussion is

detailed in full in Chapter 14.

12.1 Base configuration of AIAUV

Utilising the theory from Chapters 3 and 4 that details the construction of AIAUV dynamics from link parameters we summarise the link configuration with the list in Table 12.1. The parameters for the individual links in this table is further detailed in Appendix A.

Link #	**Link type**
1	Short link with aft thruster
2	Cardan joint
3	Short link
4	Cardan joint
5	Long link
6	Cardan joint
7	Short link
8	Long link
9	Short link

Table 12.1: Link configuration for base AIAUV.

Link #1 is chosen as the base-link and its pose and velocity is used as ϑ_1 and ζ_1 respectively as detailed in Chapter 3. The rest of the joints are ordered according to Equation 3.12 in Chapter 3. We commonly denote link #1 as the "tail" of the AIAUV, and link #9 as the "head".

12.2 Controller parameters

The numerical values for all controller gains that are part of the control-hierarchy in Chapter 9 is listed in Table 12.2. The process for determining these specific parameters consists of trial and error through multiple simulations with different AIAUV configu-

rations, both with and without wings. We do admit that better system performance could likely be achieved with better tuning. Note also that for the exploratory tests in Chapter 10 $K_{I,U} = 0$, due to an error on our part.

Controller gain	Numerical value
Δ	16
K_{p,θ_0}	1.1
K_{I,θ_0}	0.2
K_{p,q_i}	120
K_{d,q_i}	70
$K_{p,U}$	150
$K_{I,U}$	20

Table 12.2: Controller parameters for base AIAUV.

12.3 Winged AIAUV

Creating the wing augmented AIAUV is accomplished by injecting hydrodynamic wing forces into the system dynamics. The method developed for this is explained in Chapter 7. This section lists the necessary parameters to create the model for the hydrodynamic forces used in simulations as well as what link-frame the wing is attached to.

As mentioned earlier we reuse the link configuration of the base, unwinged, AIAUV summarised by Table 12.1. We also keep the controller architecture and parameters the same and use the parameters from Table 12.2. We create a fully winged AIAUV with a wing at every rigid link of the AIAUV, which wing is connected to what link is shown in Table 12.3.

The wing that is attached at every link is identical and a symmetric in the sense that there is a lifting surface on both the port and starboard side of the link. The wing is unswept, untrimmed and uncambered and with lift- and drag-coefficients according to those in Chapter 6.

Wing #	**Connected to link #**
Wing 1	Link 1
Wing 2	Link 3
Wing 3	Link 5
Wing 4	Link 7
Wing 5	Link 9

Table 12.3: Configuration of wings, showing which link each wing is rigidly attached to for winged AIAUV.

The wing model used for simulations also deviates somewhat from the method described in Chapter 7 as we combine the lift and drag forces from each of the two lifting surfaces into one expression. As we are restricting the AIAUV to motion only in the $X-Z$-plane, there is no reason to include the rolling or yawing moments generated by the wings. This can be accomplished by locating the pressure center of the wing, $\mathbf{c_p}$, at the central line of symmetry of the cylindracal link it is attached to. This in turn causes $\mathbf{c_p}$ to coincide with the x-axis of the link-frame, and the wrench it enacts on its parent link have only non-zero elements in the links x and z direction and only a moment around pitch. To reflect that there are two lifting surfaces, on each side of the link, we double the expression for lift and drag from Chapter 6 to reflect that each of the two wings will produce its own lift and drag, making the total lift and drag experienced by the link double.

The expression used to generate the Lift and Drag forces are shown in Equation 12.1. Table 12.4 state the parameters used for calculating the point of attack and the coefficitents for lift and drag. Again remember that the parameters in Table 12.4 only represents the parameters for either the port or starboard wing, but they are identical for both sides, and Equation 12.1 makes sure the lift and drag depicts the correct total force from combining both wings.

$$L = 2\frac{1}{2}\rho C_L(\alpha)AV^2 \tag{12.1a}$$

$$D = 2\frac{1}{2}\rho C_D(\alpha)AV^2 \tag{12.1b}$$

Parameter name	**Symbol**	Value
Wingspan	s	0.4m
Chord length	c	0.15m
Planform area	A	0.06m^2
Aspect ratio	Λ	2.67
Pressure center	$\mathbf{c}_p$	$[0.375, 0, 0]^{\text{T}}$ [m]
Trim	α_0	0°

Table 12.4: Parameters for generating lift and drag model for wing used in simulations.

Remark. *Lastly, we would like to remark that the wingspan and chord length of the wings were chosen by us due to their reasonable dimensions relative to the length of links on the AIAUV, and that they give an aspect ratio within the region of validity for our lift and drag approximations. The choice is not backed by any literature but purely by common sense and intuition. We encourage that a more thorough investigation into wing-geometry be performed if further work is to be performed on this winged AIAUV.*

We would also like to comment that our intent was to locate the pressure center of all wings at the geometric center of the link it is attached to. This is the reasoning behind the particular value of $\mathbf{c}_p = [0.375, 0, 0]^{\text{T}}$. *This gives the right location for pressure centers for the short links of the AIAUV, but misses the mark for the long link, link #5 with length,* $l_5 = 1m$. *This is an oversight on our part, but should not significantly change the simulation results as it only changes the moment-arm for the pitching moment of the wings on a single link by* $0.5 - 0.375 = 0.125m$.

Chapter 13

Simulation results

This chapter is devoted to the analysis and discussion of the results from the simulation comparison detailed in Chapter 11. A brief summary is that these results are a comparison of the same AIAUV, with the same control-hierarchy and the same guidance objectives, but with and without wings at every link.

The chapter is divided into five main sections. The first section compares the performance of the two AIAUV's with regards to their guidance objectives. The second examines the response of the pitch controller. The third section compares the time evolution of their state-space. The fourth section compares the energy demands of the two AIAUVs. The last section documents the wing forces for the wing-equipped AIAUV.

Remark. *Throughout this chapter we only present plots showing the whole simulation-time, and no figures adequately represent the transient phenomena occurring when switching active waypoints. We understand that the transient behaviour is both interesting and deserve a proper discussion. We address the transient behaviour from our simulations further in our final conclusion found in Chapter 14.*

Further more, we argue that for the purpose of comparing the energy demands with and without wings most of the savings are achieved in the time-frame when travelling between waypoint and not at the time of switching of active waypoints.

We would like to point out that every plot in this chapter includes the time-stamp when switching of waypoints occured. The switching-time of the active waypoint is marked with a black vertical dashed line, and labeled "wp(k)", where k signifies the index of the waypoint the AIAUV is traveling towards from that time onwards. Finally, after reaching waypoint "wp(5)", the AIAUV is tasked with maintaining the same speed U_d and keeping the same pitch course until the simulation ends at $t_f = 900$[s].

We would also like to emphasise that this thesis restricts the study only to motion lying in the $X-Z$ plane even though the AIAUV is modeled in full degrees of freedom. We have omitted the simulation results for coordinates not lying in this plane from this chapter but they can be found in Appendix B for completeness. As expected the results in Y, roll, and yaw, are zero for the whole simulation duration.

13.1 Guidance comparison

First lets see if our systems are able to reach the desired waypoints. In Figure 13.1 we plot the x_1^0 and y_1^0 base-position at each timestep of the simulation for both the base configuration of the AIAUV and the one equipped with wings. The red circles in Figure 13.1 denotes the circle of acceptance associated with each waypoint.

From the figure we can see that both configurations are able to visit all waypoints, however there are some discrepancies in their trajectories. The winged AIAUV has a steeper dive angle, a more horizontal trajectory in the bottom trench, and a steeper rise. From our study cross-track errors in Chapter 10 we remember the trend we observed of the winged AIAUV having a smaller cross-track error than the AIAUV without wings and conclude that winged AIAUV is closer to the reference straight-line trajectory formed by connecting the waypoints.

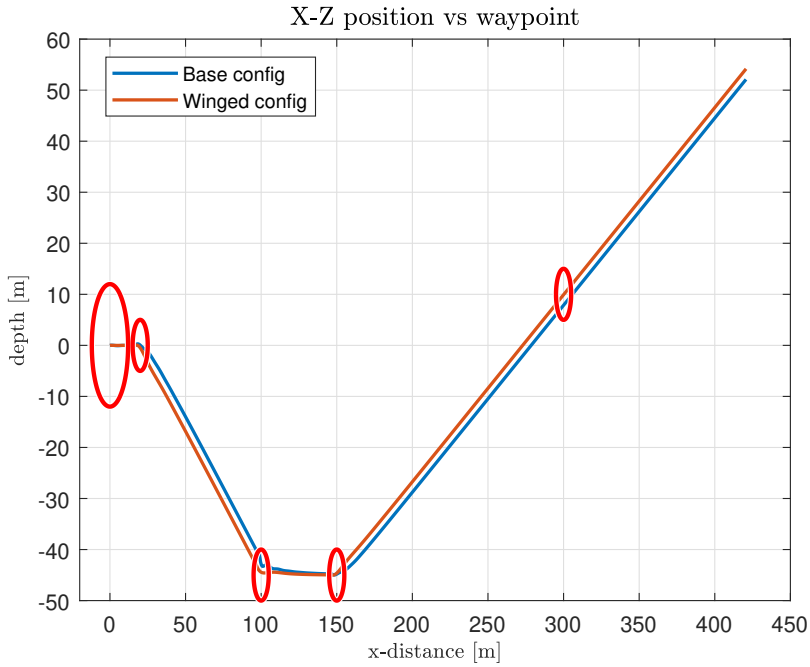

Figure 13.1: X-Z position of base link for AIAUV with and without wings.

The guidance law from Chapter 9 computes a desired mean pitch for the AIAUV, $\bar{\theta}_d$, as a mean to steer the AIAUV. Figure 13.2 compares the desired and actual mean pitch for an unwinged AIAUV in the top plot and a winged AIAUV in the bottom plot. The mean pitch of the AIAUV, $\bar{\theta}$, is computed as the average of all individual link's pitch angles.

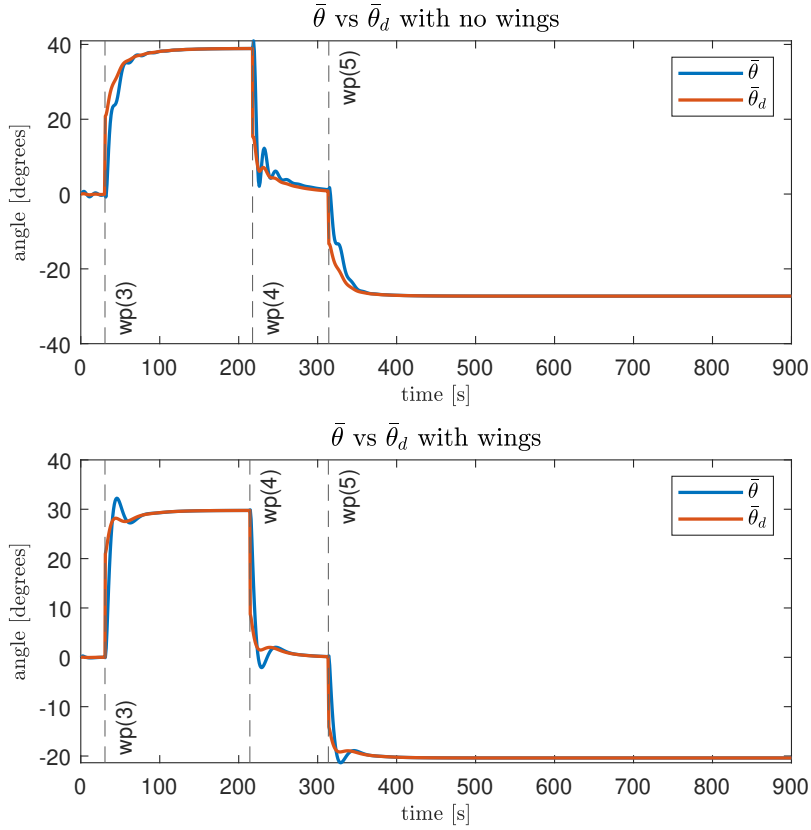

Figure 13.2: Mean AIAUV pitch vs reference for both unwinged and winged AIAUV.

Lets now divide the discussion of Figure 13.2 into three parts, one part corresponding to the tracking of each waypoint. When tracking the third waypoint we observe a larger angle for both the desired and actual mean pitch of the base AIAUV when compared to the one with wings. A similar trend can be observed for the last waypoint where we are trying to rise. Compare this with Figure 13.1 where we remarked that

the base cofiguration of the AIAUV strays further from the prescribed trajectory. We note that the base AIAUV requires a steeper mean body pitch to dive or climb than the one with wings. In other words, the winged AIAUV have better pitching capabilities under this control hierarchy.

The ideal trajectory when "wp(4)" is the active waypoint is a flat horizontal line between it and the previous waypoint. Also here we see worse performance for the AIAUV without wings, as $\bar{\theta}_d$ takes longer to settle to approach zero. The transient behaviour when switching waypoints give an overshoot for the winged AIAUV, likely caused by too high proportional gain for its pitch controller. The same high proportional gain could also explain the oscillatory behaviour of the base AIAUV when setting "wp(4)" as the active waypoint.

13.2 Pitch controller response

As seen in the previous section the winged AIAUV has a smaller $\bar{\theta}$ and $\bar{\theta}_d$ for both the dive and rise portion of the simulation when compared to the base AIAUV. The pitching controller from Chapter 9 is responsible for aligning the mean pitch of the AIAUV with its desired value by generating references for the rotary joints. The joint references are generated from the variable θ_0 as seen in Equation 9.6, where each joint reference of the AIAUV is increasing in amplitude from the head towards the tail. A positive value for θ_0 creates joint references that curve the head upwards, while negative θ_0 curve the head downwards. The variable θ_0 is a synthetic control variable generated as the output of a PI controller of the error in mean pitch, $\bar{\theta}$ as detailed in Equation 9.7.

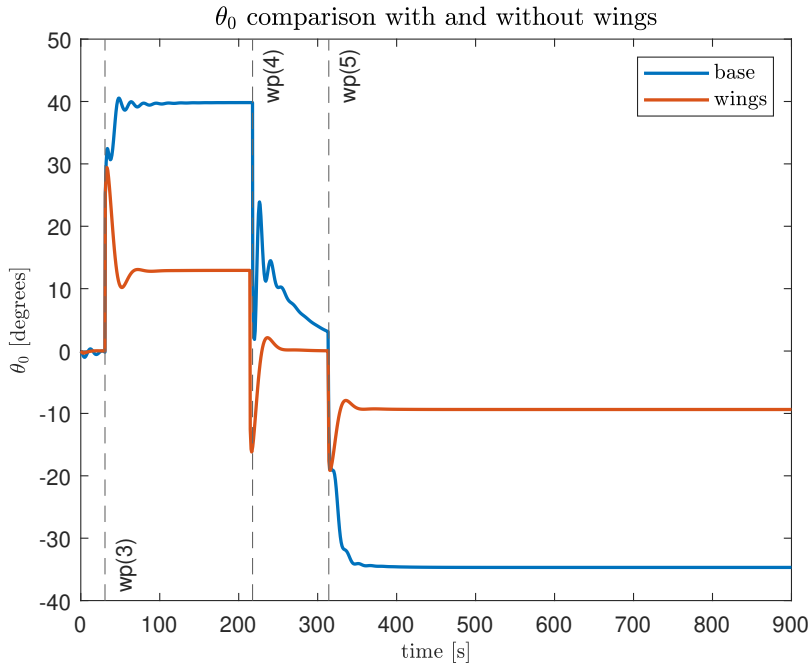

Figure 13.3: Comparing θ_0 for the winged and unwinged AIAUV

The time evolution of θ_0 for both the winged and unwinged AIAUV is shown in Figure 13.3. First note the overshoot occuring at each waypoint switch for the winged AIAUV, this is further evidence supporting the theory that the proportional gain for the pitch-controller might be to high. For both the rise and dive portion of the simulation θ_0 approaches a non-zero value for both the winged and unwinged AIAUV. This is quite worrysome and requires further explanation.

First off, the non-zero values will cause a curvature of the AIAUV body for the whole duration of the dive and rise. As shown in Figure 13.2 the error in $\bar{\theta}$ approaches zero in the later stages of both the dive and rise for both AIAUV's. This means that the non-zero θ_0 is due to the accumulated error in the integral term of the pitch controller.

The high stationary values for θ_0 might be lowered through a lower gain for the integrating term, or a more sophisticated implementation for the integration itself as aluded to in Chapter 11.

The main takeaway from this section is that the winged AIAUV can keep its course with a smaller body curvature than the AIAUV without wings. This is beneficial as we want a low body curvature and a more elongated body to reduce the drag experienced by the links. However, some body curvature has to be expected as both AIAUV's experience restoring moments from their non-coinciding mass- and buoyancy-centers. If the body curvature is zero and the AIAUV is completely elongated it will eventually spring back to a horizontal orientation with zero pitch.

13.3 Results for state-space variables

This section includes plots of all linear positions, and angles lying in the $X-Z$-plane as well as their velocities for both the winged and unwinged AIAUV. The plot of joint angles also include their respective reference value as generated by the pitch controller. Not every plot will be discussed in detail, but all of them are included for the completeness of this thesis.

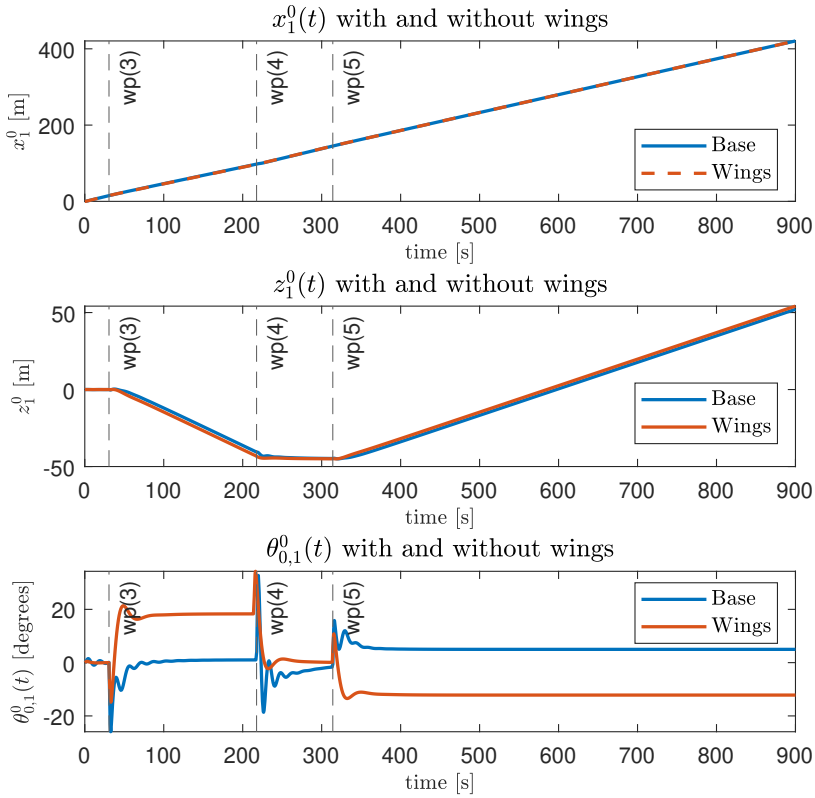

Figure 13.4: Comparing position of base link for winged and unwinged AIAUV

First we would like to draw attention to the plot of pitch angle for the base link shown as the bottom plot of Figure 13.4. The general trend is for the base-link of the winged AIAUV to have a larger degree of pitch in both the dive and rise segment than its unwinged counterpart. Together with the information from Figure 13.3 in the previous section we can establish that it has a more elongated body oriented along

the desired course. This is further evidence that it is better at course-keeping while keeping the drag experienced by the links low compared to the unwinged AIAUV. We attribute this to the addition of wings as all other parameters are identical.

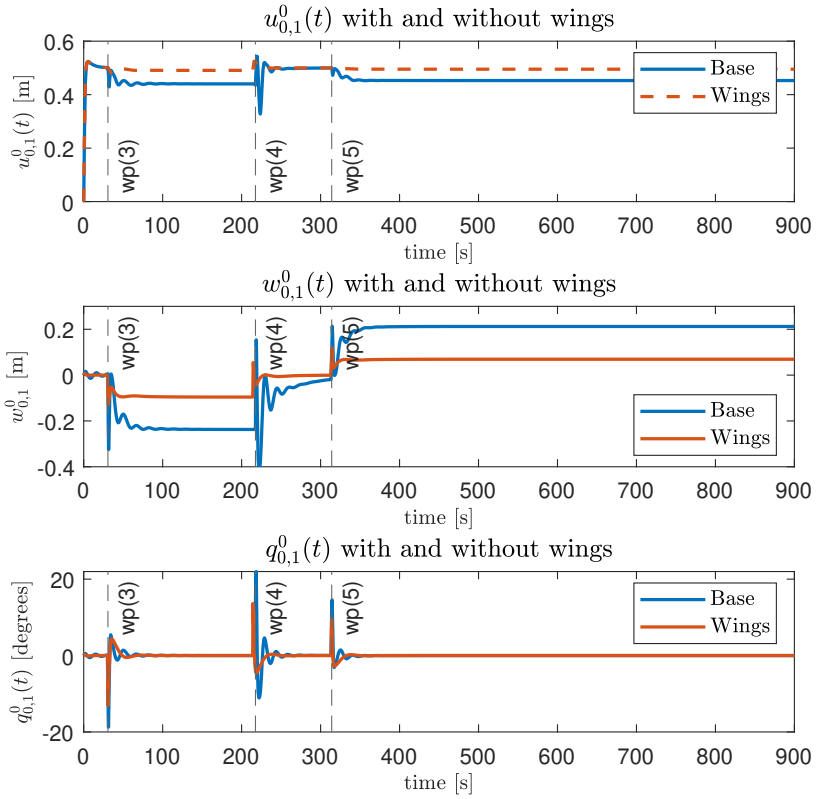

Figure 13.5: Comparing body-velocites of base link for winged and unwinged AIAUV

In Figure 13.5 we see that the winged AIAUV is able to orient its velocity direction

more along its body compared to the unwinged AIAUV. This is signified by the winged AIAUV having a base-link with surge-speed, $u^1_{1,0}$ closer to the desired forward speed, $U_d = 0.5[\text{m/s}]$, and much lower heave, $w^1_{1,0}$, than the unwinged AIAUV.

Remark. *We would like to remark that the plot labels in Figure 13.5 are incorrect and we are infact plotting body-velocities, $v^1_{1,0}$, and not spatial velocities, $v^0_{0,1}$.*

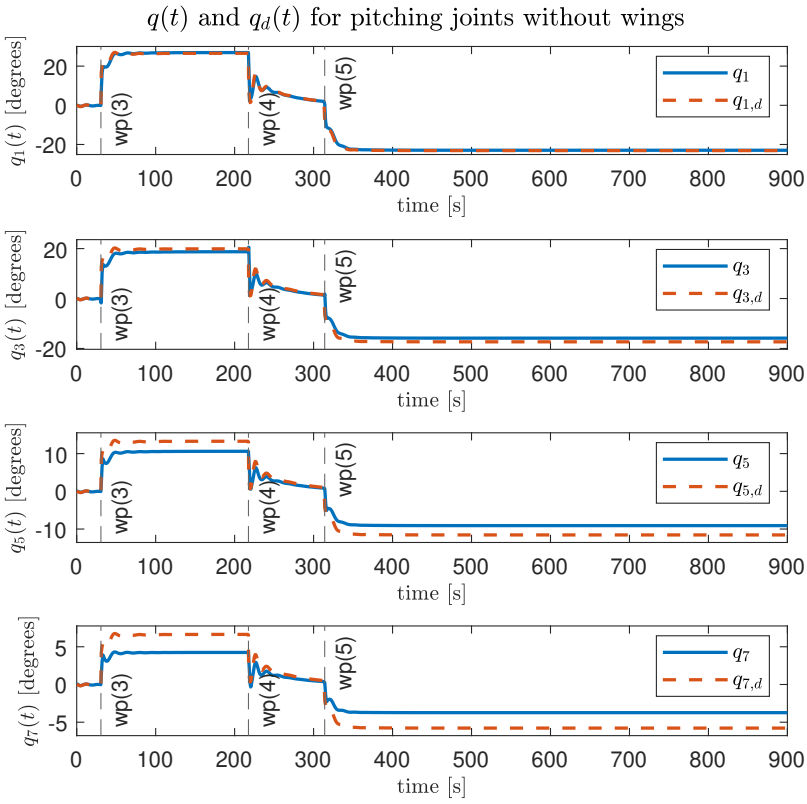

Figure 13.6: Joint angles and reference for pitching joint of unwinged AIAUV.

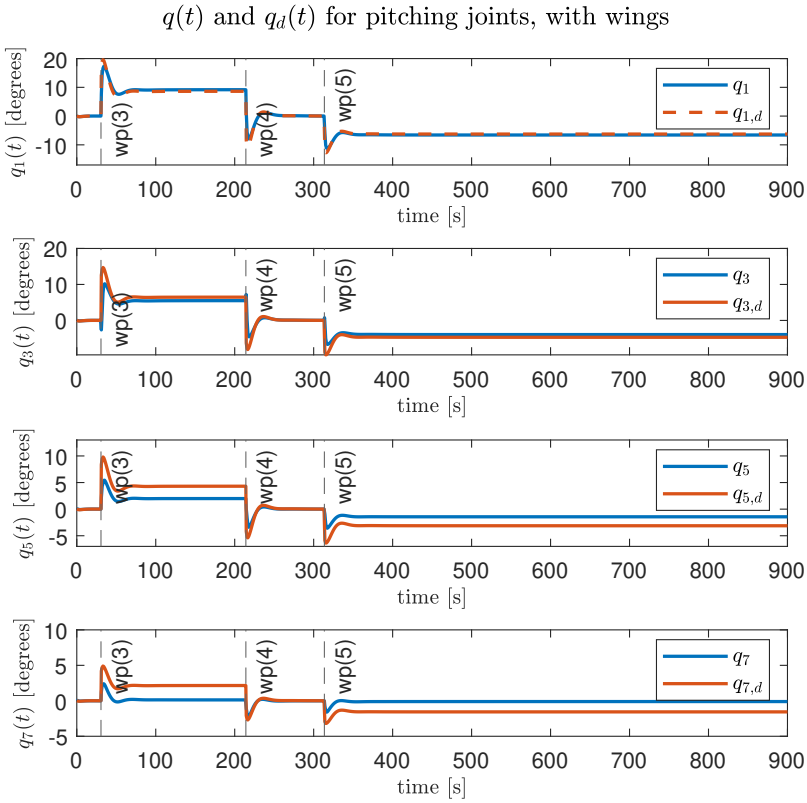

Figure 13.7: Joint angles and reference for pitching joint of winged AIAUV.

Figure 13.6 and Figure 13.7 show the joint angles q_i and their reference, $q_{i,d}$ for joints acting in pitch for the unwinged and winged AIAUV respectively. Common for both configurations are good tracking for the joints closest to the tail, q_1 and q_3, with performance deteriorating towards the head, q_5 and q_7, where the actual joint amplitude is smaller than the commanded amplitude. One likely explanation for this is

that the internal joint controller only includes a proportional term and a rate-limiting derivative term and that seems insufficient for tracking the joint references.

All joints for the winged AIAUV approaches a stationary value of smaller amplitude than the unwinged AIAUV, again underlining that it has a more elongated body. The joint rates $\dot{q}_i$ for both the unwinged and winged AIAUV are presented in Figures 13.8 and 13.9 respectively. They will not be discussed but are included for completeness. And finally Figures 13.11 and 13.10 show the control inputs for the joint of the winged and unwinged AIAUV respectively. The control inputs for joint q_i is u_i and correspond to the magnitude of the torque produced by the joint. The joint torques were saturated at $|u_| \leq 11$Nm and outside of the transient behaviour when changing waypoints we are well within the limits.

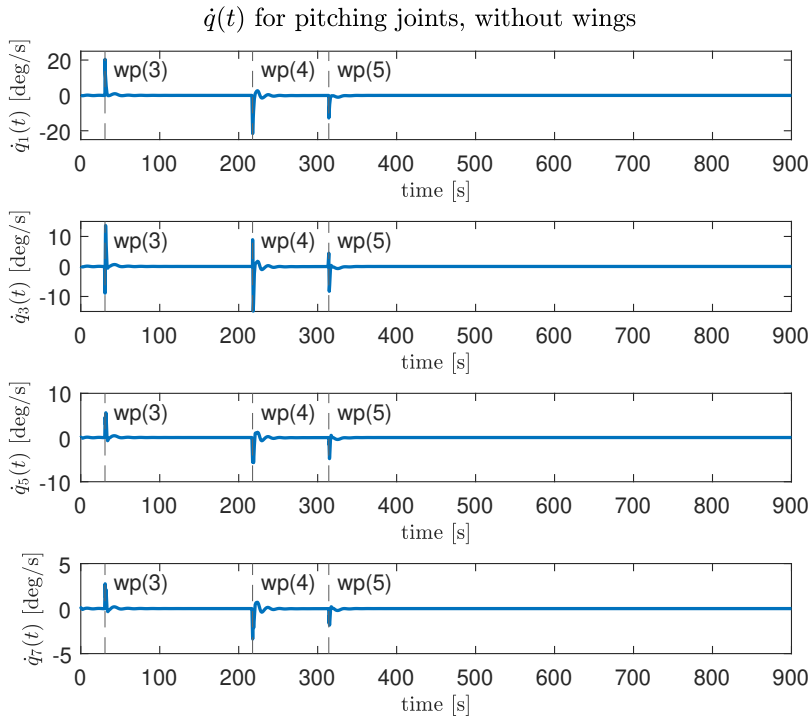

Figure 13.8: Angular velocity for pitching joint of unwinged AIAUV.

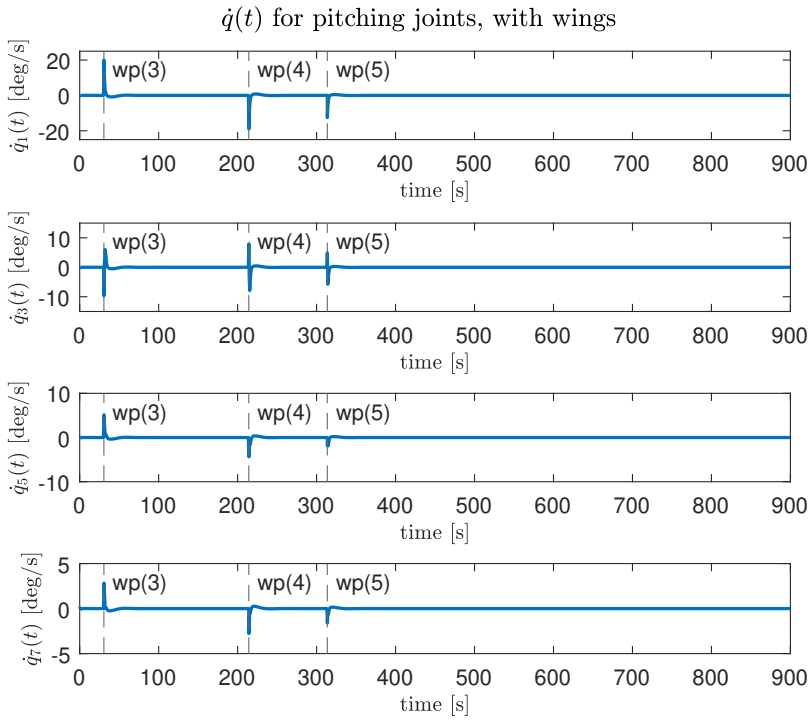

Figure 13.9: Angular velocity for pitching joint of unwinged AIAUV.

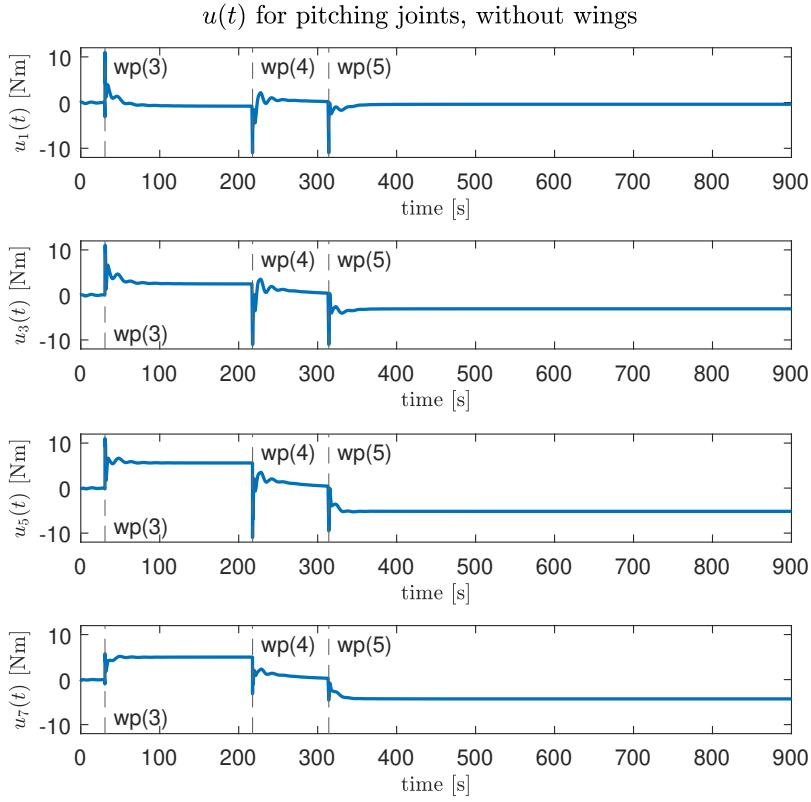

Figure 13.10: Commanded torque for pitching joints for unwinged AIAUV.

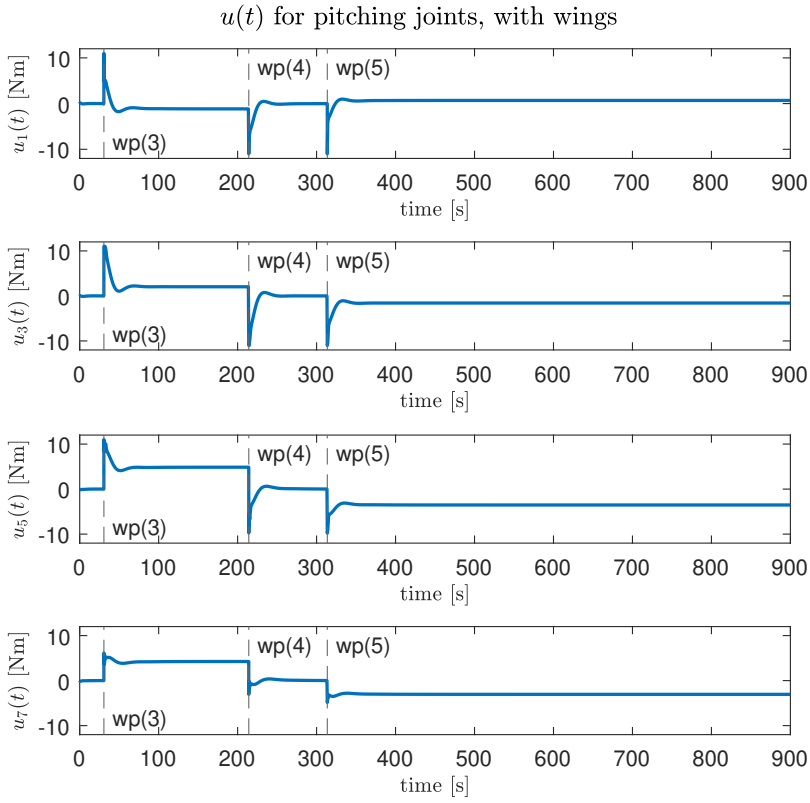

Figure 13.11: Commanded torque for pitching joints for winged AIAUV.

13.4 Power comparison

This section compares the energy demands of the two AIAUV configurations with the metrics developed in Chapter 8. But first a little reminder of our findings in Chapter

10. In that chapter we commented that across all tests the energy used by the thruster dominates that of the joints by orders of magnitude. With that in mind, we would like to show some more data pertaining to the speed-controller subsystem of our control hierarchy.

The control objective of the speed controller is to keep the forward speed of the base link at a constant magnitude at $U_d = 0.5\text{ms}^{-1}$ by commanding the appropriate thrust force produced by the aft thruster. The control law for the speed controller can be found in Chapter 9, Equation 9.10. To better understand the difference in performance in thruster utilization between the winged and unwinged AIAUV we created the plots in Figures 13.12 and 13.13. Where Figure 13.12 plot the forward speed, control input to aft thruster, and instantanous power from the aft thruster for the unwinged AIAUV, and Figure 13.13 plot the same data for the unwinged AIAUV.

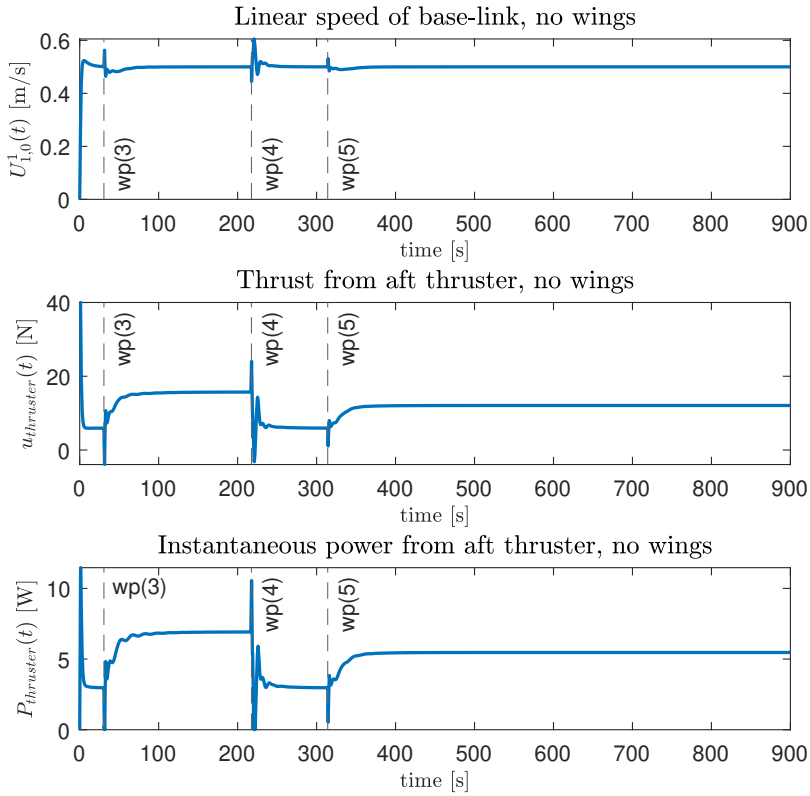

Figure 13.12: Forward speed, thrust, and instantanous power from aft thruster for unwinged AIAUV.

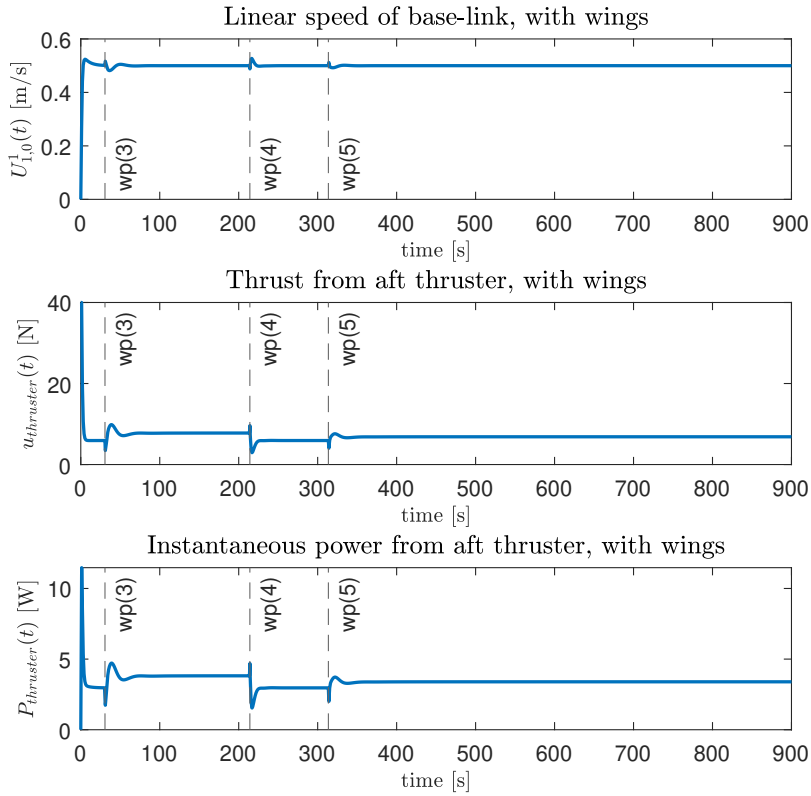

Figure 13.13: Forward speed, thrust, and instantanous power from aft thruster for unwinged AIAUV.

When comparing the linear speed of the base link for the winged and unwinged AIAUV we see that the transient behaviour after switching waypoints fluctuates more for the unwinged configuration than the winged one. This together with the difference in heave velocites of the base link seen in Figure 13.5 might imply that the wings have

a stabilising effect on the AIAUV which in itself is beneficial.

We can now compare the thrust use for the unwinged and winged AIAUV, again shown in Figure 13.12 and Figure 13.13 respectively. We see that the unwinged AIAUV requires more thrust at all times than it winged counterpart. From the increased control input and relatively similar velocities it also follows that the instantaneous power from the thrusters is also greater at every time for the unwinged AIAUV, shown in the bottom plots of the aforementioned figures. The increased thrust is, as explained in Section 13.2, due to the increased curvature, θ_0, that the unwinged AIAUV has to maintain in order to keep its desired mean pitch angle. The increased curvature makes the AIAUV experience more drag and require more energy to maintain the desired speed.

As first encountered during initial testing in Chapter 10, we observed that the energy demand of joint actuators were orders of magnitude less than that of the aft thruster, the same is true for the instantaneous power. The instantanous power from pitching joint actuators for the unwinged and winged AIAUV are presented in Figures 13.14 and 13.15 respectively. We observe the same trend in these simulations also, and our recommendation is to focus effort on reducing the thrust forces necessary if one wish to further reduce the energy demands of this system in future work.

We conclude this section by showing the total energy demand for thrusters and joints for both AIAUV's in Table 13.1. The total energy is found by integrating the respective instantaneous power over the whole simulation time as detailed in Chapter 8. E_T denotes the total energy from the aft thruster, and E_τ denotes the total energy from all joints. From the table we have clearly saved energy by equipping the AIAUV with wings. With wings the total thruster energy is 3084.33J , while without wings it is 4798.54J. In this experiment, with these AIAUVs, under this control hierarchy the winged AIAUV used 64.3% of the energy of its unwinged counterpart.

Configuration	E_T	E_τ
Unwinged AIAUV	4798.54J	8.63J
Winged AIAUV	3084.33J	9.26J

Table 13.1: Total energy from joints and thruster supplied for the whole simulation for unwinged and unwinged AIAUV.

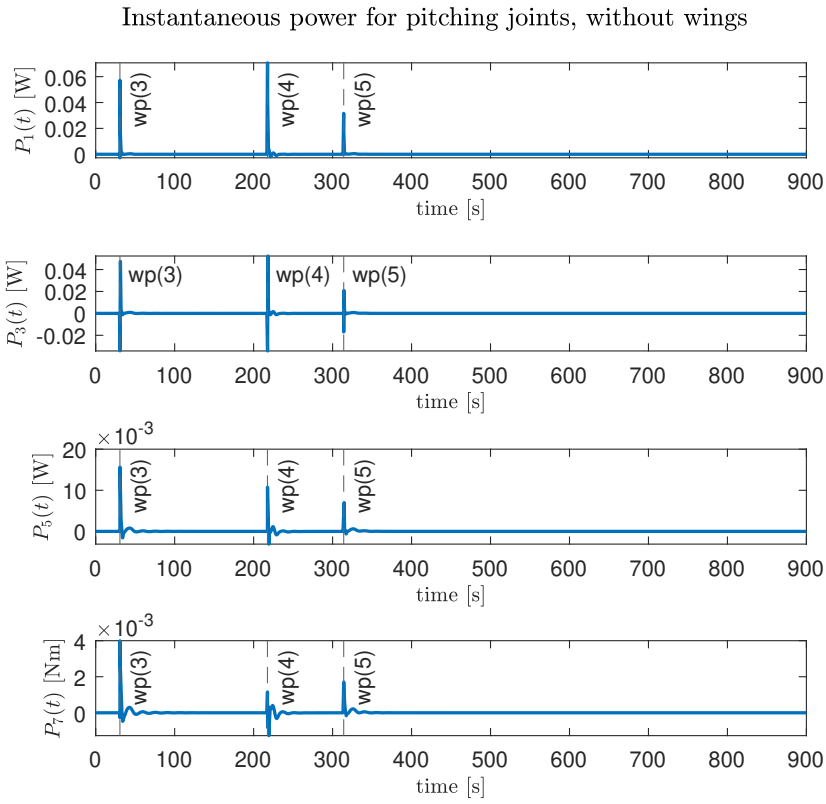

Figure 13.14: Instantaneous power from pitching joints for AIAUV without wings.

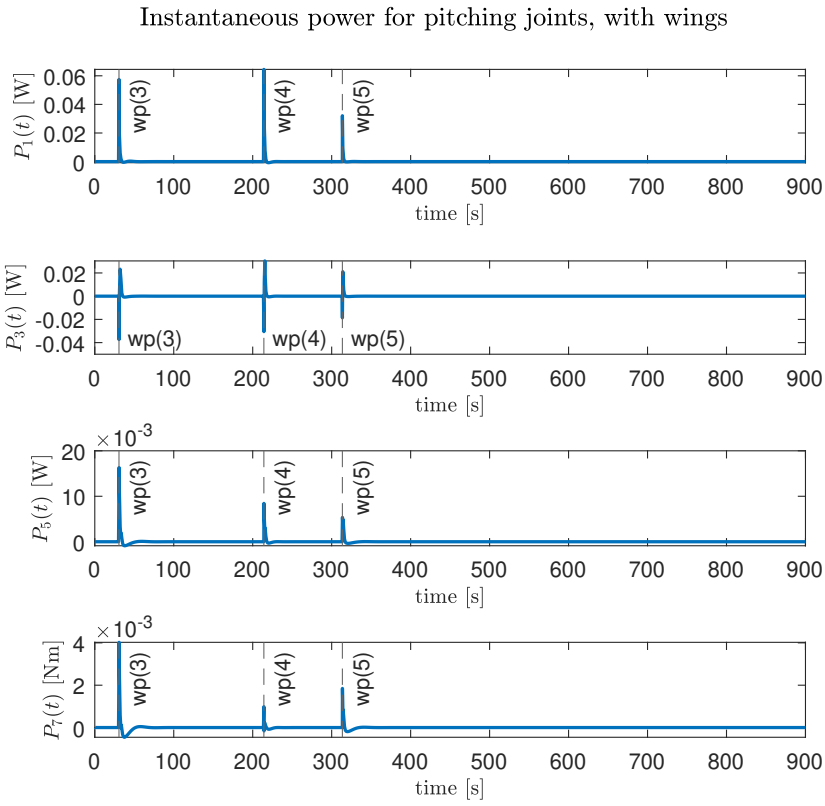

Figure 13.15: Instantaneous power from pitching joints for AIAUV without wings.

13.5 Wing forces

We conclude the chapter by including plots for the angle of attack and hydrodynamic forces generated by each wing on the winged AIAUV, shown in Figures 13.16 - 13.20.

The hydrodynamic forces of each wing, lift and drag, are decomposed into their normal and axial components, N and A respectively. As the wings have zero trim this means that positive direction for A is along the $x - axis$ of the link frame the wing is attached to, while positive direction for N is along the $z - axis$. This allow us to see the local effect that each individual wing has on its parent link. We also note that all wings experience angle of attacks within our modeling threshold of $\pm 20^{\circ}$.

We observe the general trend during the dive portion of our simulations of wings at the aft of the AIAUV, at link #1 and #3, generate a positive normal force, while wings towards the head, at links #7 and #9, generate a negative normal force. The effect of this is to force the tail up and the head down. Similarly the behaviour is mirrored for the rise occuring when tracking "wp(5)" where the wings push the tail down and the head up. We use this to support our argument that the wings do in fact aid in pitching the AIAUV towards it desired course.

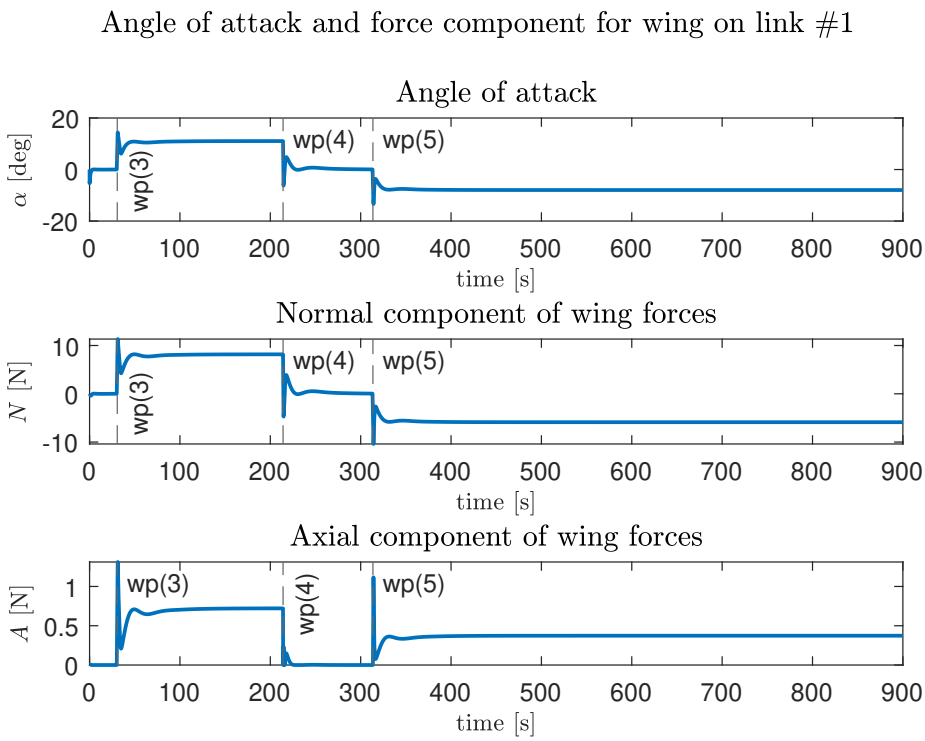

Figure 13.16: Angle of attack, and lift and drag decomposed into normal and axial forces for wing 1.

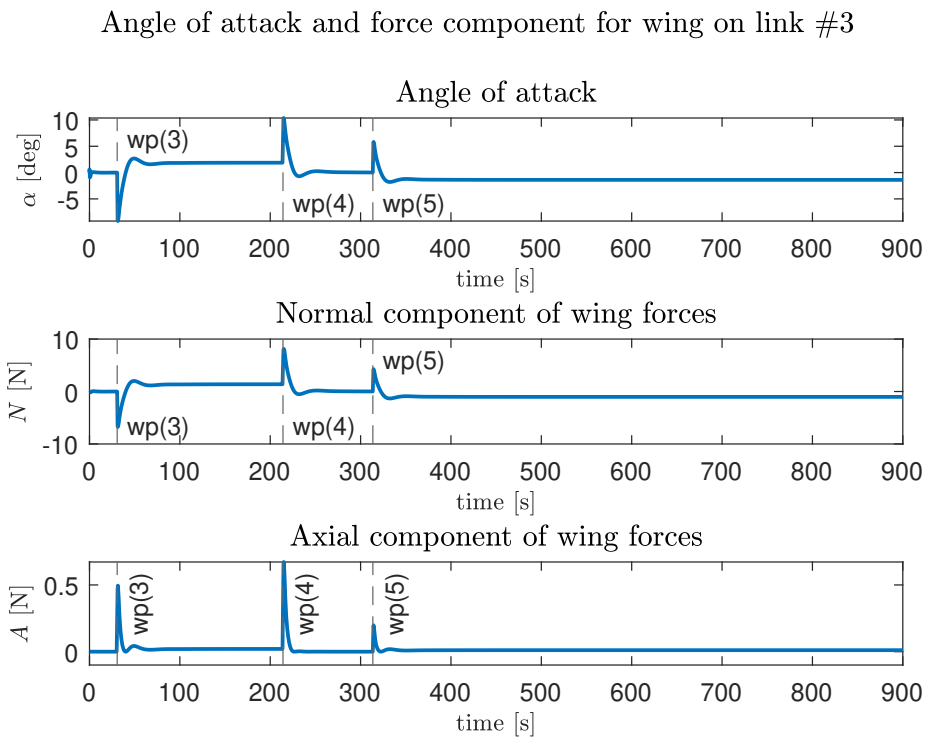

Figure 13.17: Angle of attack, and lift and drag decomposed into normal and axial forces for wing 2.

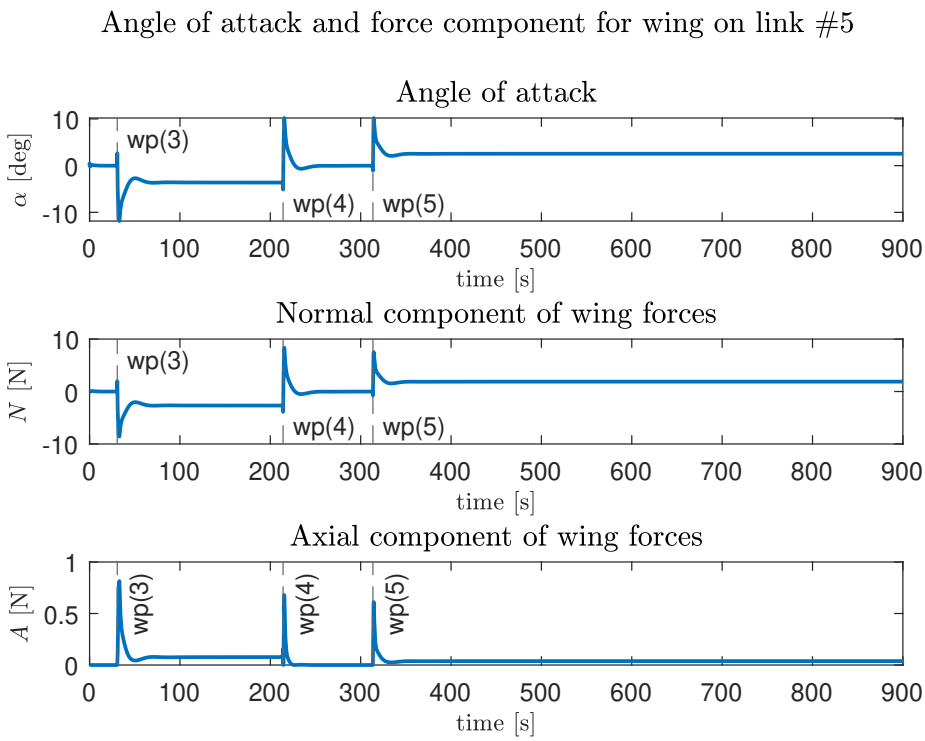

Figure 13.18: Angle of attack, and lift and drag decomposed into normal and axial forces for wing 3.

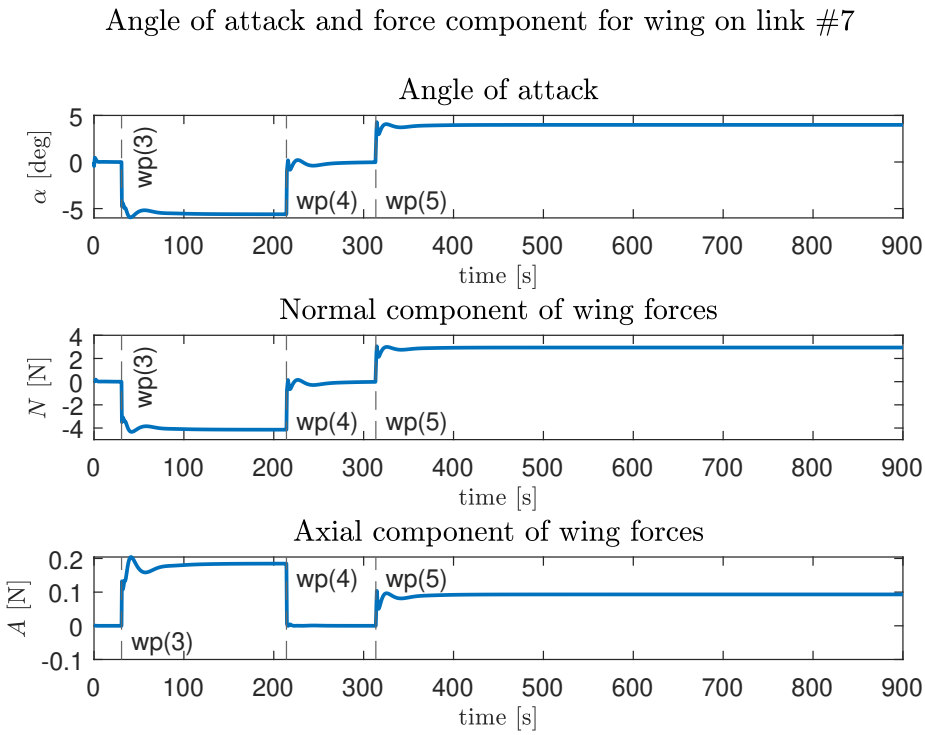

Figure 13.19: Angle of attack, and lift and drag decomposed into normal and axial forces for wing 4.

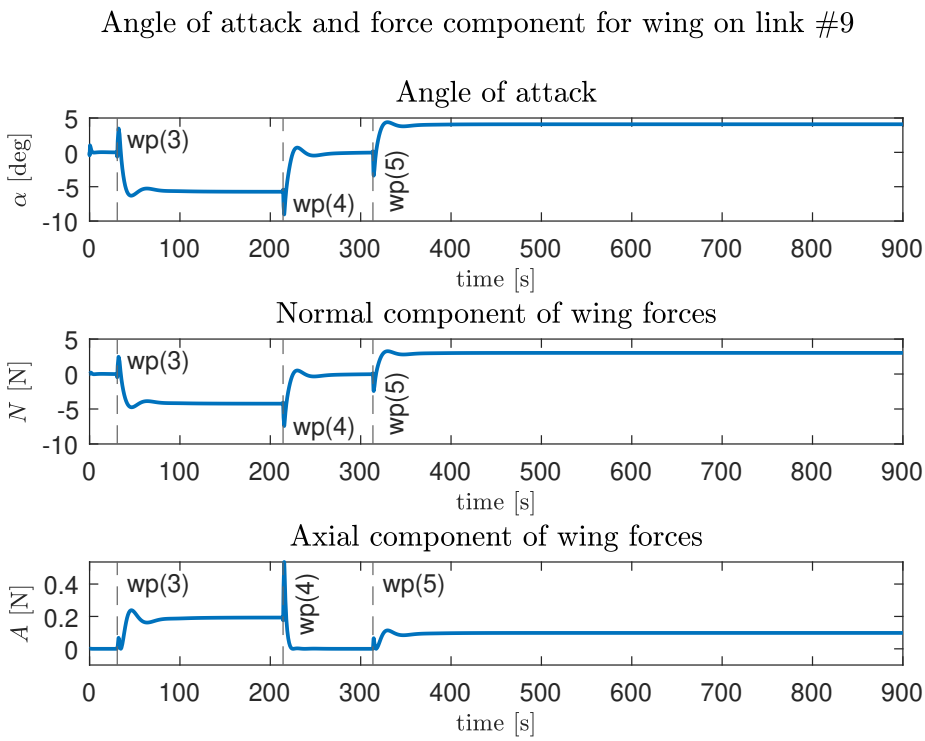

Figure 13.20: Angle of attack, and lift and drag decomposed into normal and axial forces for wing 5.

Chapter 14

Conclusions and future work

To summarise the main findings from this thesis we would like to divide the summary into three parts. One part devoted to the discussion of our work on extending the AIAUV model, one part discussing the control architecture we constructed, and lastly one part devoted to our conclusion on the energy saving potential for this class of Wing-Augmented Articulated Intervention Autonomous Underwater Vehicle.

14.1 Model extension

With the method for including wings into the AIAUV dynamics developed in Chapter 7 we have created the foundations for a solid framework that support future extensions due to its modular nature. The framework allows individually placed and trimmed wings and the approximations for lift- and drag- coefficients can be easily swapped, allowing a change of wing characteristics either with experimental coefficients or ones for other wing geometries. We also feel that this model can be extended without much work to allow actuation of the wings through a simple rotation of the link frame. A rotation of the link frame have the effect of rotating the orientation of the wings and will enable them to be used as control surfaces.

The motion of the AIAUV was constrained to the $X - Z$-plane for all simulations

in this thesis, and as such the wings developed for this thesis only support motion in the same plane. Much of the fundamental work needed to support unconstrained motion is already in place, but new research on hydrodynamic modeling should be performed as we constrained our research to 2D-flow models. The computation for the wing wrench acting on the link must also be extended to account for sway-velocities.

Lastly we would reiterate that the winged AIAUV model developed in this thesis is most suited for system design and analysis. Due to the choice of incorporating the hydrodynamics into a rigid-body model we have no explicit formulation of the ocean flow-field surrounding the AIAUV. The hydrodynamic forces from the wings are higly dependent on the exact flow conditions in its proximity and we fear that the assumption of hydrodynamic decoupling between links and ideal flow is to naive to give a realistic simulation. This means that the quantification of energy-savings from wings is accompanied by a high degree of uncertainty. We do however feel that this model can be beneficial for system design and testing purposes, as well as its use as for example a reference-model in a control-architecture.

14.2 Controller architecture and performance

As the notion of an AUV with articulated body and wings is quite novel we were not able to find existing research on control for this class of AUV. As a first step towards creating a control-architecture for the winged AIAUV we opted to piece together parts from different sources, the result is the control-architecture found in Chapter 9. With this architecture we accomplished waypoint-tracking in the $X-Z$-plane and were able to compare the AIAUV with and without wings as shown in Chapter 13.

14.2.1 Guidance law

In our presentation of the simulation results in Chapter 13 we remarked that the transient behaviour of the AIAUV's would be expanded upon in this Chapter, its now time to address that. At every switch of active waypoints we observe sharp jumps and oscillating motions before the system eventually settles to a steady configuration and

velocities. Part of the blame might lie on poorly tuned control gains or the rudimentary implmentation of integrating terms, but we argue that its mostly attributed the choice of guidance law.

As the guidance system connects waypoints with straight lines we will have a sharp jump in desired course angle, α_k at every switch. This jump in desired pitch then propagates throughout the whole control system and leads to poor tracking of objectives right after the switch. This problem could be alleviated by either padding the user-requested waypoints with more waypoints on either side to give a step-wise smoother transition, or one could introduce some rate-limitation α_k.

We would also propose that the lookahead based LOS algorithm we have chosen is ideal for the task of energy-efficient locomotion. Artificially constraining the winged AIAUV to follow an arbitrary line through space by controlling the cross-track error to zero is likely to not be energy optimal for the system. This is a topic worth exploring in further work.

14.2.2 Pitch controller

The pitch controller is adapted from a paper on spiral path planning for docking an AUV, the paper in question is Sans-Muntadas et al. (2017). In that paper they present a heading controller for the MAMBA snake robot that we have repurposed as a pitch controller. From our findings in the work on this thesis we would argue that this is the key component for proper utilisation of the wings. We recommend a deeper study on this topic in particular. Testing different algorithms for distributing the joint angles along the body of the AIAUV instead of the algorithm in this thesis could likely increase the pitching potential of the wings.

14.2.3 Joint and thruster controllers

The low level joint and thruster controller seems to perform their purpose adequately for the simulations in this thesis. However, the poor tracking of joints towards the head should be investigated. Furthermore the choice of U as the body-velocity of the tail-link for the speed controller might not have been wise. The linear speed of the

middle link of the AIAUV or some other composite metric of speed might be a better choice to characterise the total forward speed of the system. The choice of using the linear speed of the base-link was simply due to it being easily accessible as part of the state-space of the model.

14.3 Drawbacks of MATLAB implementation

As we mentioned in Chapter 11, the framework required to implement the simulations for this thesis in MATLAB was quite extensive. Requiring us to make custom functions for datalogging, a rather intrusive and basic method for integrating terms in control laws, and spreading the tunable parameters for the controllers all across the code-base. In hindsight we understand that much of this work could have been avoided if the dynamics were imported into a more suitable framework such as Simulink, where many of these features already exist and our code-base could have been more modular.

14.4 Energy saving potential of wings

The main purpose of this thesis was to investigate if an AIAUV could save energy if it was equipped with wings. All work presented in this thesis has been in pursuit of an answer to this question. With the simulations in Chapter 13 that compares a winged AIAUV with an unwinged AIAUV we have shown at least one scenario where we save energy. As a reminder we restate the total energy required for both systems below in Table 14.1.

Configuration	E_T	E_τ
Unwinged AIAUV	4798.54J	8.63J
Winged AIAUV	3084.33J	9.26J

Table 14.1: Total energy from joints and thruster supplied for the whole simulation for unwinged and unwinged AIAUV.

From our simulation experiment we see a clear difference in the energy required by the aft thruster for the winged and unwinged AIAUV. We attribute this to the fact that the wings aid in pitching the AIAUV to the desired pitch course and that we can maintain a greater mean pitch, $\bar{\theta}$, with a more elongated body with wings. The joint references generated by the pitch controller orients the links so that in a dive the wings push the head down and the tail up, and vice versa for a rise.

In contrast the unwinged AIAUV has to rely on its aft thruster both to maintain the forward speed and keep its mean pitch. With our control method this is achieved with a larger body curvature for both the dive and rise. With a larger curvature the aft thruster is oriented such that it can maintain both the forward speed and desired mean pitch. However, this increases both the drag experienced by the AIAUV and requires more thrust, ultimately requiring more energy. A question for further work on this topic is whether a control architecture more suited for the unwinged AIAUV can close the gap in energy demands and mitigate some of the advantage the winged AIAUV has shown to have in this study.

We feel that we have adequately showed the potential of this new platform, but would like to end with a final reminder of the limitations of our results. Due to the simplified model for both the AIAUV dynamics, the wing forces, and thruster dynamics one would not expect to see these exact energy demands in real world experiments, but reality should show the same trend for a winged AIAUV.

Appendix A

AIAUV link parameters

A.1 Short link

Name	Short link
Length $[m]$	0.75
Radius $[m]$	0.1
Mass $[kg]$	23.562
c_g $[m]$	$[0.375, 0, -0.05]^{\mathrm{T}}$
c_b $[m]$	$[0.375, 0, 0]^{\mathrm{T}}$
$c_{F,i}$ $[m]$	None
$F_{dir,i}$	None

Table A.1: Parameters for short link

$$M_{RB} = \begin{bmatrix} 23.5619 & 0 & 0 & 0 & -1.1781 & 0 \\ 0 & 23.5619 & 0 & 1.1781 & 0 & 8.8357 \\ 0 & 0 & 23.5619 & 0 & 8.8357 & 0 \\ 0 & 1.1781 & 0 & 0.884 & 0 & 0.2209 \\ -1.1781 & 0 & -8.8357 & 0 & 4.4768 & 0 \\ 0 & 8.8357 & 0 & 0.2209 & 0 & 4.4473 \end{bmatrix} \tag{A.1}$$

$$M_A = \begin{bmatrix} 4.7124 & 0 & 0 & 0 & 0 & 0 \\ 0 & 23.5620 & 0 & 0 & 0 & 8.8357 \\ 0 & 0 & 23.5620 & 0 & -8.8357 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -8.8357 & 0 & 4.4179 & 0 \\ 0 & 8.8357 & 0 & 0 & 0 & 4.4179 \end{bmatrix} \tag{A.2}$$

A.2 Long link

Name	Long link
Length $[m]$	1.0
Radius $[m]$	0.1
Mass $[kg]$	31.416
c_g $[m]$	$[0.5, 0, -0.05]^{\mathrm{T}}$
c_b $[m]$	$[0.5, 0, 0]^{\mathrm{T}}$
$c_{F,i}$ $[m]$	None
$F_{dir,i}$	None

Table A.2: Parameters for long link

$$M_{RB} = \begin{bmatrix} 31.4160 & 0 & 0 & 0 & -1.5708 & 0 \\ 0 & 31.4160 & 0 & 1.5708 & 0 & 15.7080 \\ 0 & 0 & 31.4160 & 0 & -15.7080 & 0 \\ 0 & 1.5708 & 0 & 0.1178 & 0 & 0.3927 \\ -1.5708 & 0 & -15.7080 & 0 & 10.5505 & 0 \\ 0 & 15.7080 & 0 & 0.3927 & 0 & 10.5112 \end{bmatrix} \tag{A.3}$$

$$M_A = \begin{bmatrix} 6.2832 & 0 & 0 & 0 & 0 & 0 \\ 0 & 31.4160 & 0 & 0 & 0 & 15.7080 \\ 0 & 0 & 31.4160 & 0 & -15.7080 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -15.7080 & 0 & 10.4720 & 0 \\ 0 & 15.7080 & 0 & 0 & 0 & 10.4720 \end{bmatrix} \tag{A.4}$$

A.3 Short link with aft thruster

Name	Short link with aft thruster
Length $[m]$	0.75
Radius $[m]$	0.1
Mass $[kg]$	23.562
c_g $[m]$	$[0.375, 0, -0.05]^{\mathrm{T}}$
c_b $[m]$	$[0.375, 0, 0]^{\mathrm{T}}$
$c_{F,i}$ $[m]$	$[0, 0, 0]^{\mathrm{T}}$
$F_{dir,i}$	$[1, 0, 0]^{\mathrm{T}}$

Table A.3: Parameters for short link with aft thruster

$$M_{RB} = \begin{bmatrix} 23.5619 & 0 & 0 & 0 & -1.1781 & 0 \\ 0 & 23.5619 & 0 & 1.1781 & 0 & 8.8357 \\ 0 & 0 & 23.5619 & 0 & 8.8357 & 0 \\ 0 & 1.1781 & 0 & 0.884 & 0 & 0.2209 \\ -1.1781 & 0 & -8.8357 & 0 & 4.4768 & 0 \\ 0 & 8.8357 & 0 & 0.2209 & 0 & 4.4473 \end{bmatrix} \tag{A.5}$$

$$M_A = \begin{bmatrix} 4.7124 & 0 & 0 & 0 & 0 & 0 \\ 0 & 23.5620 & 0 & 0 & 0 & 8.8357 \\ 0 & 0 & 23.5620 & 0 & -8.8357 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -8.8357 & 0 & 4.4179 & 0 \\ 0 & 8.8357 & 0 & 0 & 0 & 4.4179 \end{bmatrix} \tag{A.6}$$

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}} \tag{A.7}$$

A.4 Cardan joint

The Cardan-joint serves as an ideal connection between two links and have trivial dynamics and no hydrodynamic properties. it's purpose is purely to separate the other links in space. And allow the implementation of AIAUV dynamics to assign one simple rotation as the joint coordinate for each link.

$$\mathbf{M}_{RB} = \mathbf{M}_A = \mathbf{B} = \mathbf{0} \tag{A.8}$$

Name	Generic link
Length $[m]$	0.02
Radius $[m]$	0.1
Mass $[kg]$	0
c_g $[m]$	$[0, 0, 0]^{\mathrm{T}}$
c_b $[m]$	$[0, 0, 0]^{\mathrm{T}}$
$c_{F,i}$ $[m]$	None
$F_{dir,i}$	None

Table A.4: Parameters for Cardan-joint link

Appendix B

Simulation results in Y, roll, yaw

The results in this appendix accompany the simulations in Chapter 13. As the AIAUV is modeled with full degrees of freedom its motion is not constrained to the $X-Z$ plane. We constrain it to the $X-Z$ plane artificially by having no actuation in the degrees of freedom corresponding to the $Y-axis$ or rotations in roll, ϕ, and yaw ψ.

Below are the simulation results for positions and velocities in the states corresponding that can give motion not contained in the $X-Z$ plane for both the winged and unwinged AIAUV. As expected they are all trivial for the whole duration of simulation.

Remark. *The titles of Figure B.5 and Figure B.6 are wrong, they are plots for the winged AIAUV, not the one without.*

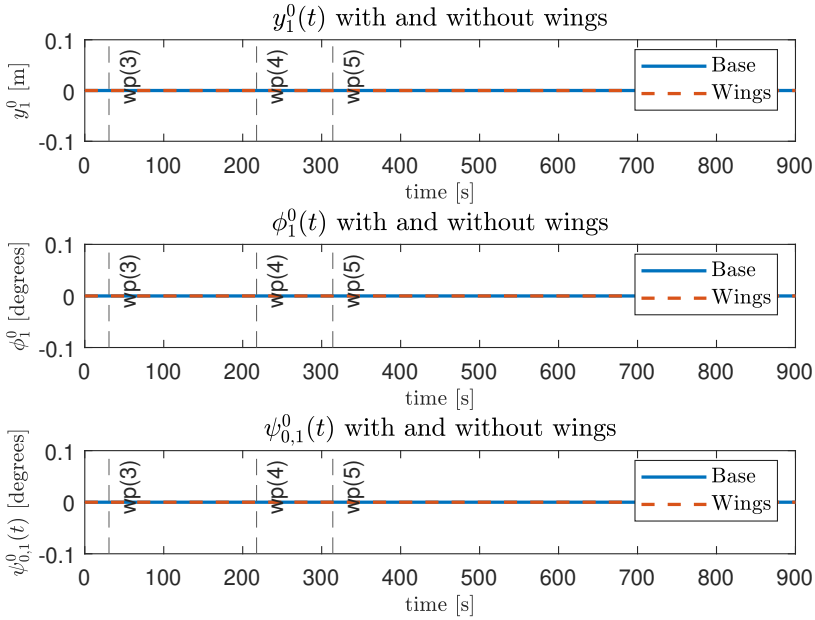

Figure B.1: Base positions in y, ϕ, ψ for both winged and unwinged AIAUV.

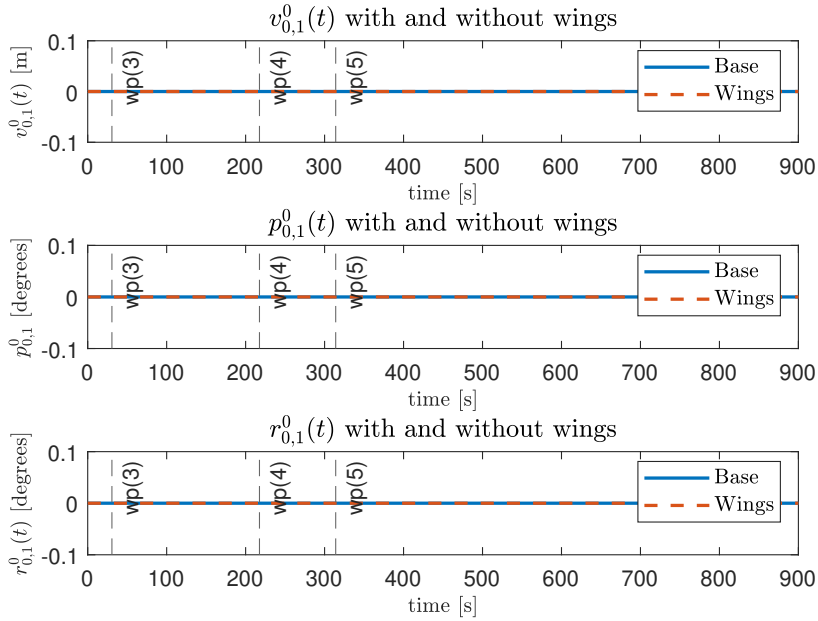

Figure B.2: Base velocities in y, ϕ, ψ for both winged and unwinged AIAUV.

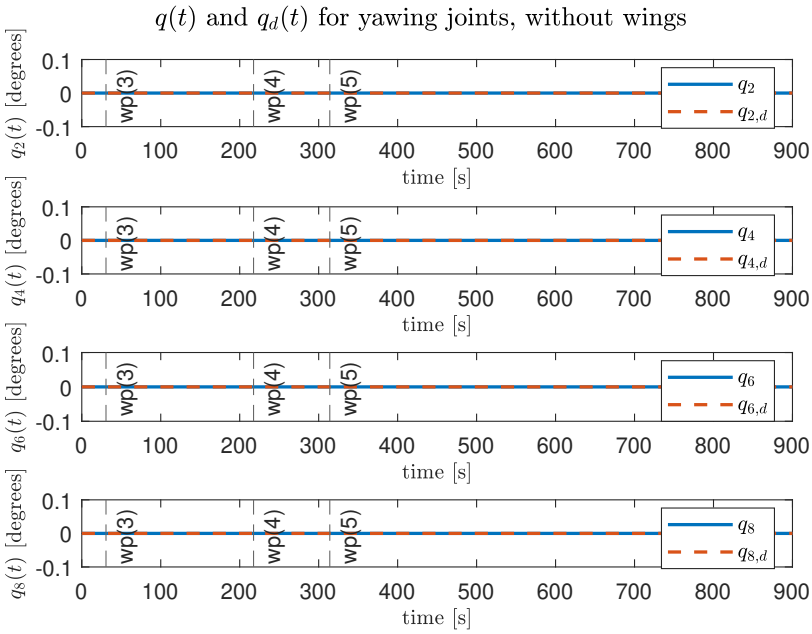

Figure B.3: Yawing joint angles for base config of AIAUV.

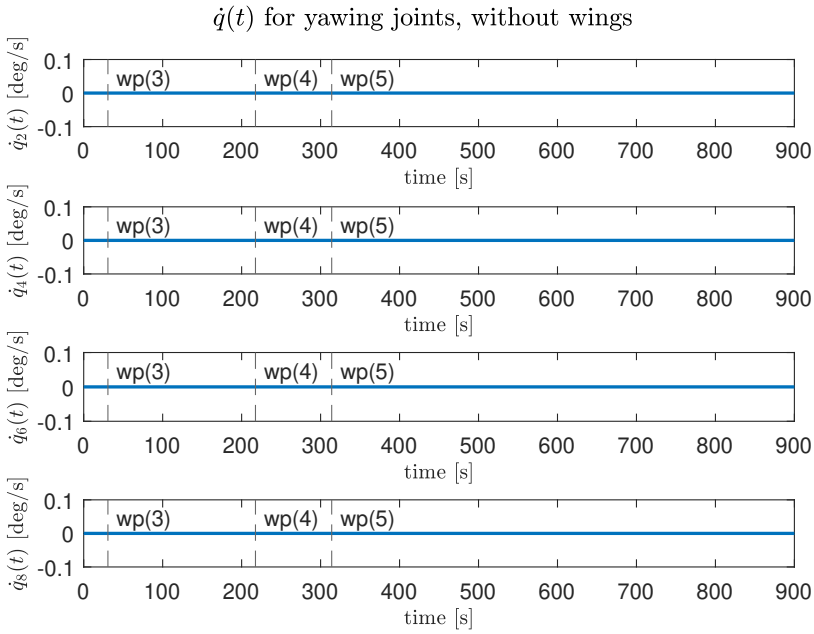

Figure B.4: Yawing joint angular velocities for base config of AIAUV.

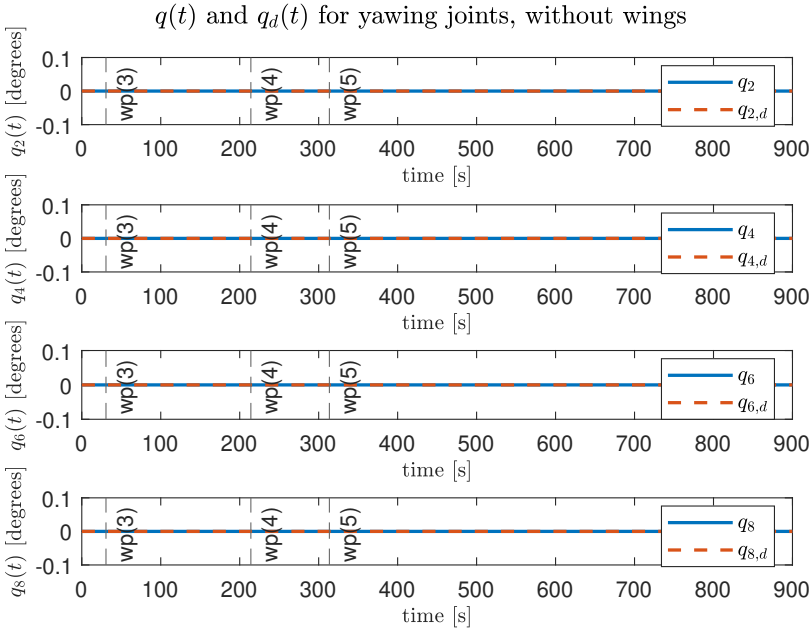

Figure B.5: Yawing joint angles for winged config of AIAUV.

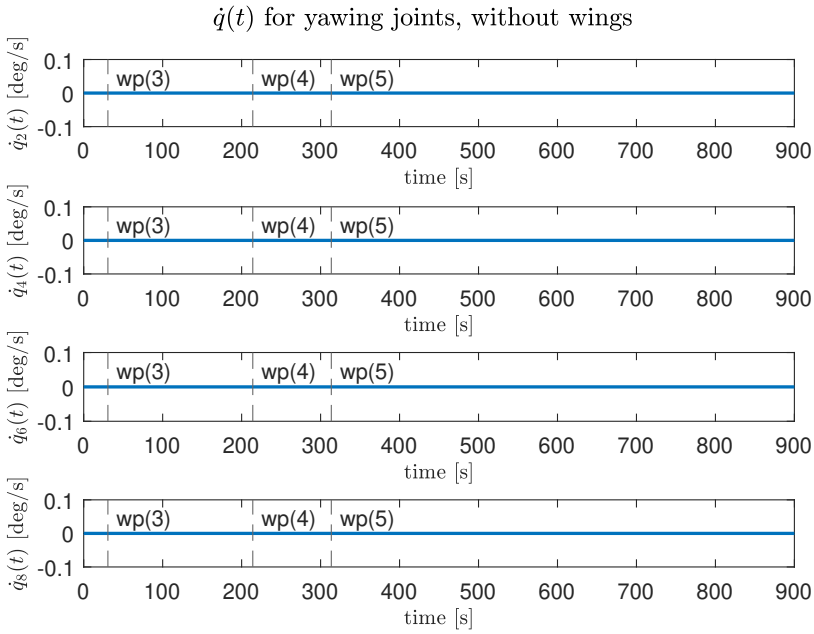

Figure B.6: Yawing joint angular velocities for winged config of AIAUV.

References

Antonelli, G. (2018). *Underwater Robots*, Vol. 123 of *Springer Tracts in Advanced Robotics*, Springer International Publishing, Cham.

Cengel, Y. and Cimbala, J. (2013). *Fluid Mechanics Fundamentals and Applications: Third Edition*, McGraw-Hill Higher Education.
URL: *https://books.google.no/books?id=QZIjAAAAQBAJ*

Chyba, M., Leonard, N. and Sontag, E. (2001). Optimality for underwater vehicle's, Vol. 5, pp. 4204 – 4209 vol.5.

Claus, B., Bachmayer, R. and Williams, C. D. (2010). Development of an auxiliary propulsion module for an autonomous underwater glider, *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment* **224**(4): 255–266.

Egeland, O. and Gravdahl, J. T. (2002). *Modeling and simulation for automatic control*, Vol. 76.

Faltinsen, O. M. (2006). *Hydrodynamics of High-Speed Marine Vehicles*, Cambridge University Press.

Fossen, T. I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*, John Wiley & Sons, Ltd.

From, P. J., Gravdahl, J. T. and Pettersen, K. Y. (2016). *Vehicle-manipulator systems*, Springer.

Graver, J. G. and Leonard, N. E. (2001). Underwater glider dynamics and control, *12th international symposium on unmanned untethered submersible technology*, Citeseer, pp. 1742–1710.

Henry Stommel | Woods Hole Oceanographic institution, Woods Hole, M. U. (1989). The slocum mission, *Oceanography* **2**.
URL: *https://doi.org/10.5670/oceanog.1989.26*

Inanc, T., Shadden, S. C. and Marsden, J. E. (2005). Optimal trajectory generation in ocean flows, *Proceedings of the 2005, American Control Conference, 2005.*, IEEE, pp. 674–679.

Javaid, Muhammad Yasar, Ovinis, Mark, Nagarajan, T and Hashim, Fakhruldin B M (2014). Underwater gliders: A review, *MATEC Web of Conferences* **13**: 02020.
URL: *https://doi.org/10.1051/matecconf/20141302020*

Kelasidi, E., Jesmani, M., Pettersen, K. Y. and Gravdahl, J. T. (2018). Locomotion efficiency optimization of biologically inspired snake robots, *Applied Sciences* **8**(1): 80.

Lee, T., Leok, M. and McClamroch, N. H. (2017). *Global formulations of Lagrangian and Hamiltonian dynamics on manifolds: a geometric approach to modeling and analysis*, Springer.
URL: *https://cds.cern.ch/record/2271407*

Sans-Muntadas, A., Kelasidi, E., Pettersen, K. Y. and Brekke, E. (2017). Spiral path planning for docking of underactuated vehicles with limited fov, *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 732–739.

Schmidt-Didlaukies, H. M., Sørensen, A. J. and Pettersen, K. Y. (2018). Modeling of articulated underwater robots for simulation and control, *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pp. 1–7.

Ullah, B., Ovinis, M., Baharom, M. B., Javaid, M. Y. and Izhar, S. S. (2015). Underwater gliders control strategies: A review, *2015 10th Asian Control Conference (ASCC)* pp. 1–6.