AeroDima: A Cheetah-Inspired Aerodynamic Tail for Rapid Manoeuvrability

Daryn Bright
Thesis presented for the degree of Masters of Science in Engineering
In the Department of Electrical Engineering
University of Cape Town

Supervisor:
A/Prof. A. Patel

September 2023

DECLARATION

I know the meaning of plagiarism and declare that all the work in the document, save for what is properly acknowledged, is my own.

Signed by candidate
Signature

27/05/2024
Date

ABSTRACT

The cheetah, the fastest land animal has been hypothesised to use its tail for manoeuvrability and has been the source for many bio-inspired robotic tails. However, the use of a lightweight tail to achieve the same effects has not be studied. This paper goes into the study of using a lightweight, aerodynamic tail to increase manoeuvrability on a wheeled platform, *AeroDima*.

This is achieved by studying the aerodynamics of a cheetah's tail in a wind tunnel to develop a quasi-steady state model. A bio-inspired aerodynamic tail was then designed to maximise drag forces. This bio-inspired tail was also studied in a wind tunnel and compared to the cheetah tail's model. The tail required a platform to operate on, so *AeroDima* was designed and manufactured. The tail, designed to be a 3DOF underactuated system, was designed to test the effects of the tail on high-speed manoeuvres through a roll motion swing of the tail. The system was tested both in simulation and physical experiments. The simulation, developed with MATLAB's Simscape Multibody toolbox, was designed to be a comprehensive model of *AeroDima*.

Testing of the simulation model showed that the addition of the tail enabled faster turning movements, the fastest successful turn occurring at 4.36 m/s, by increasing the stability during the turning motion. The Kohut manoeuvrability metric, being defined by $v\dot{\theta}$ [1] and used in previous robotic studies[2], however decreased with the addition of the tail. The physical experiments showed a more significant result with *AeroDima* increasing its speed with the addition of the tail by 81.01% to 6.1 m/s. Not only did the tail increase the capable forward velocity but it also increased the stability of *AeroDima* by decreasing the total roll displacement in the roll axis by 37.16%. The instantaneous Kohut manoeuvrability increase by 88.8%.

These results show that an aerodynamic tail is more beneficial for instantaneous high-speed manoeuvres than for drawn out manoeuvres as shown in previous robotic tail studies. It was also shown in simulation that at higher speed runs having the tail to be more passive (actuating less) was more beneficial. These results demonstrate that an aerodynamic tail can aid manoeuvrability at high-speed, but that tail actuation is more complex than originally thought necessitating future work to investigate more advanced control algorithms to leverage its full benefits.

ACKNOWLEDGEMENTS

I would like to acknowledged and thank my God for giving me the strength and endurance to finish this paper. I would like to thank my family for their continued support and encouragement.

I would like to thank my life group community that have been there for me through the good and the bad. To the same extent, I would like to thank my friends for their support and encouragement.

To my supervisor, Dr. Amir Patel, I want to thank you for the privilege of working under your guidance and supervision. You have shown me what it means to be a good and meticulous engineer. I have learnt so much from you, and I am grateful for the opportunity to work with you.

To the members of my research unit, the African Robots Unit, thank you for being part of the journey with me. Mr. Chris Mailer, you have been a sounding board that I've been able to bounce ideas off, and I am grateful for your input. To Mr. Matthew Church and Mr. Michael Katsoulis, thanks for all the good times you brought. You never seemed to fail to bring a cheery environment to the workspace.

To everyone else, I thank you for your loving support and encouragement. You have helped me get through this and have encouraged me to be the best I can be, and I am grateful for that.

Contents

List of Figures

List of Tables

Chapter 1: Introduction

This dissertation at a masters level documents a research and development research focusing on the study of the use of aerodynamics in cheetah tails and the implementation of a bio-inspired tail to aid with high-speed manoeuvres of a wheeled Remote Controlled (RC) Vehicle.

1.1 Background

Cheetahs are widely known to be the fastest living land animal and have been recorded of reaching running speeds of up to 29 m/s [3]. These straight line speeds are impressive alone, however these speeds occurred in special test conditions. When research was done to determine the actual hunting speeds of cheetahs, they were recorded to attain speeds of up to 18.94 m/s [4]. In the same research paper, cheetahs were seen to achieve acceleration of up to 7.5 m/s^2 which shows an inspiring agility. While the raw power of the cheetah's muscles play a large role in reaching these speeds, it is impossible for the cheetah to be able to turn at such high-speeds without the aid of some mechanism. A prey of the cheetah, the gazelle and other similar buck, gallop in their escape of being prey [5], allowing them to be manoeuvrable, which means that for the cheetah to be able to keep up with its prey, it needs to be equally manoeuvrable.

When studying a cheetah chase, it is evident that the cheetah uses its tail significantly in these chases. While not being as evident as to what the use of this movement is, it can clearly be seen that the movement of the tail aids the cheetah in their actions. This movement, as can be seen in Figure 1.1, shows the cheetah swinging its tail during the take-down of a wildebeest. The tail, in the original video [6], can be seen to be doing a full rotation. It not only swings in the roll axis, but also in the pitch axis where, at times, the tail moves in a conical motion which has led to questions being asked about what the purpose of this motion is.

(a) Cheetah with tail in near upright position

(b) Cheetah with tail angled to the right

Figure 1.1: A cheetah chase showing clear movement of its tail. The tail is shown in a (a) vertical position and a (b) horizontal position [6].

This question was asked by a researcher at the University of Cape Town in a PhD thesis [7]. In this paper, the researcher studied the motion of the cheetah's tail and showed that the tail did in fact aid in the stability of various metrics in high-speed turns as well as the stabilisation of rapid acceleration. This was based on the assumption that the tail acted as an inertial counterweight due to the weight of the tail. Using this assumption, it was found that, with the aid of an inertial tail, moving in the roll axis the robot upon which the tail was mounted, *Dima*, was able to go from having a lateral acceleration of 8.3 m/s^2 to having a lateral acceleration of 11.8 m/s^2 [7].

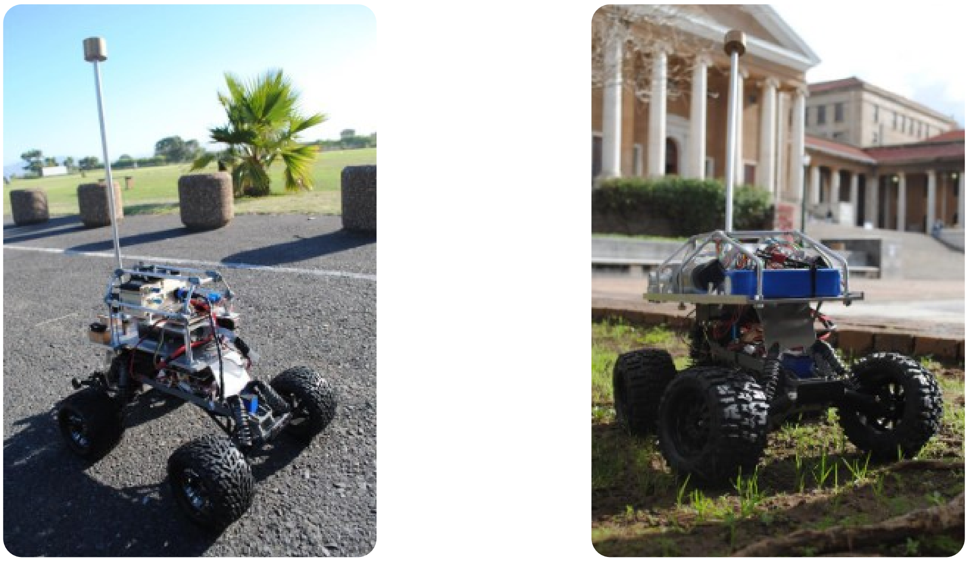

(a) *Dima* I **(b)** *Dima* II

Figure 1.2: The *Dima* platforms that were developed to determine the effect of an inertial tail on a robot. The platforms are shown in (a) *Dima* I which utilized a single degree of freedom tail and (b) *Dima* II which utilized a two degree of freedom tail [7].

However, this experiment did not end up showing the full story as after a routine necropsy, the tail was shown to have a lower inertia than originally assumed. Upon this discovery, new research was done where the tail then was put into a wind tunnel. Through this experiment, it was seen that the tail was able to produce large aerodynamics forces without having a have a large weight [8]. The fur on the tail allowed the tail to have a larger surface area which allowed for larger forces with less mass.

Based on this discovery, the question was asked as to what would happen if a bio-inspired aerodynamic tail was designed and put upon the *Dima* platform. Would the aerodynamics aid it to the same extend that the inertial tail did or would it show that the cheetah uses further unknown mechanisms to aid in its high-speed manoeuvres?

1.2 Research Scope and Outcomes

Based on the background of the research, the scope of the research can be determined. The research aims to further determine the effects of aerodynamics on a cheetah tail and create a mathematical model of these effects and using that information to determine the effects of a bio-inspired robotic aerodynamic tail on a wheeled platform, the redesigned *Dima* to be called *AeroDima*. This research will also aim to develop this wheeled platform and aerodynamic tail which includes the development of the physical robot and a simulated robot. The research can be split into three main focuses being:

focusing on the tail characteristics of the cheetah tail; focusing on the mechatronic design and build of *AeroDima* where this step will include the development of the simulation of the platform; and focusing on the testing of the platform and the analysis of the data.

Using this structure, the outcomes of this research are as follows:

- The acquisition of an aerodynamic model of a cheetah tail, mainly the coefficient of drag and lift, through the use of a wind tunnel and Euler Lagrangian Dynamics Model.
- The development of *AeroDima*, which includes the development of a three degree of freedom underactuated aerodynamic tail that can be mounted on a wheeled platform.
- The testing of the aerodynamic tail on the *AeroDima* platform, both in simulation and with the physical robot, to determine the effects of the tail on the platform.
- Prove that the use of an aerodynamic tail on the *AeroDima* platform increases the manoeuvrability by allowing faster high-speed short-term manoeuvres to be performed while increasing the stability at those speeds.

The key assumptions of this research are:

- The cheetah uses only inertial and aerodynamic effects with its tail to aid in its high-speed manoeuvres and no other forces need to be accounted for.
- Aerodynamics will have a significant effect on the stabilisation of the platform at high-speed manoeuvres.

1.3 Research Outline

The structure of the research can be split into in four sections. The first section is the preamble which includes the introduction, literature review and methodology. The second section is the development of the aerodynamic model of the cheetah tail. The third section is the development of *AeroDima* which includes the design, the build, the simulation, the testing, and the results' analysis. The final section of the research is the conclusion and ending. This outline of the research' sections can be seen in Figure 1.3.

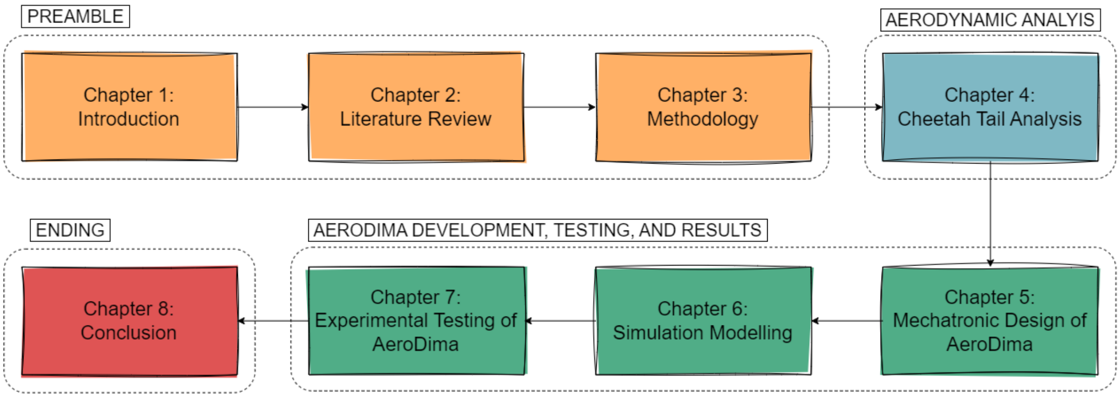


Figure 1.3: The outline of the research.

In **Chapter 2**, a study into the literature pertaining to this topic will be done. This will serve as a foundation of knowledge for the rest of the research.

In **Chapter 3**, a methodology will be developed as to how and why this research is taking place. The research process is described in detail in this chapter.

In **Chapter 4**, the cheetah tail will be analysed to determine an aerodynamic model that is fitting for the tail. This will be done through the use of a wind tunnel and an Euler Lagrangian Dynamics Model.

In **Chapter 5**, a mechatronic design of *AeroDima* will be done. This includes the design of the body of the robot, the design of the gearbox and tail, and the design of the embedded systems which include the electronics and communication systems.

In **Chapter 6**, the simulation of *AeroDima* is developed in MATLAB's Simscape Multibody toolbox. This simulation will be used to test the control of the tail before the implementation of this on the physical robot.

In **Chapter 7**, the testing of the physical robot will be done. This includes the simulating and physical testing of *AeroDima*, both with and without the tail. The results of these tests will be analysed and discussed.

Finally, **Chapter 8** concludes the paper by giving a holistic summary of the paper and the findings of the paper.

Chapter 2: Literature Review

Before getting into the research of this thesis, one needs to know about the fundamentals behind the topic. This chapter will cover the basics of aerodynamic forces, various tails (biological and robotic), and the metrics that will be used to evaluate the performance of *AeroDima* and its bio-inspired aerodynamic tail.

2.1 Aerodynamic Drag Forces

With a robotic tail designed to be lightweight, the inertial forces that will be exerted through the tail will be kept to a minimum. The lack of inertial forces will be countered by the use of aerodynamic forces on the tail and for this reason, it is important to study various possible aerodynamic forces that could be applied.

2.1.1 Drag and Lift Forces

One of the fundamental forces in fluid dynamics is the drag force. The drag force is the force experienced by the body due to relative motion of the body within a fluid which is experienced normal to the direction of the fluid flow. The relative motion does not only cause a force normal to the direction of flow, but also causes a force perpendicular to the direction of the flow, known as the lift force. To help understand where these forces come from, they must first be split into two separate components, the drag force due to pressure and the drag force due to shear [9], see Figure 2.1.

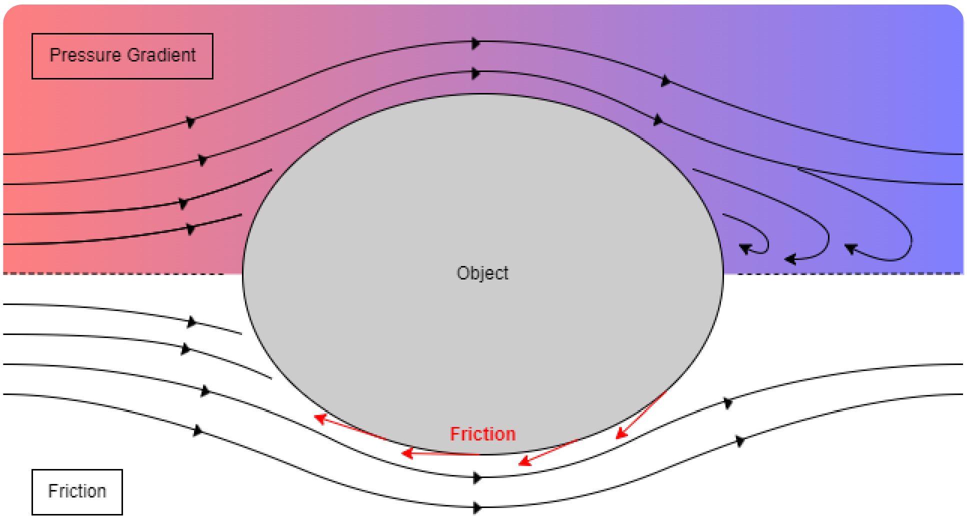


Figure 2.1: The drag force is the sum of the pressure drag and shear drag forces.

These forces have been extensively studied and used in various engineering applications from keeping aeroplanes in the sky to the racing of formula 1 cars to the design of row boats. The standard formula for the drag force is represented by Equation 2.1 [9].

$$F_D = \frac{1}{2}\rho C_D A u_r^2 \tag{2.1}$$

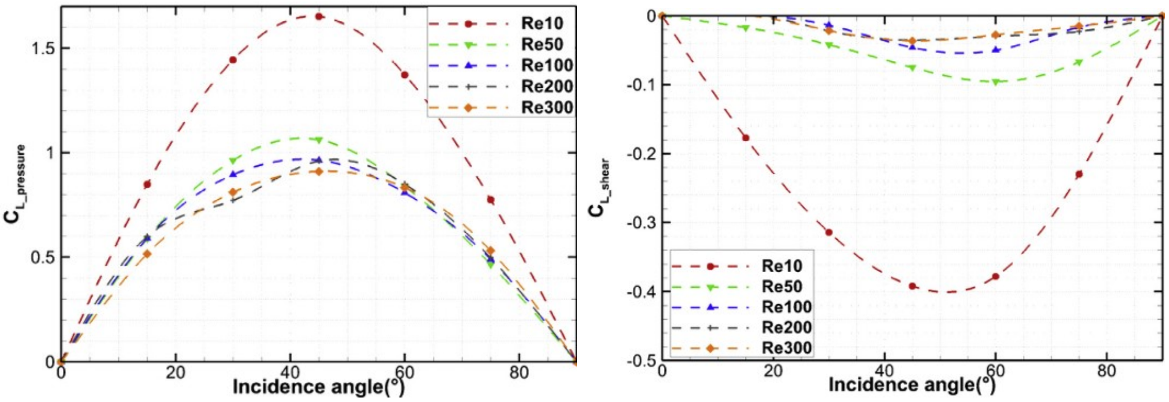
In the equation above, ρ is given as the fluid density, C_D is the coefficient of drag, A is the cross-sectional area of the object and u_r is the relative fluid velocity between the body and the fluid. This relative fluid velocity is given by Equation 2.2.

$$u_r = u_{body} - u_{fluid} \tag{2.2}$$

These formulae are the same for the lift force, however, the drag coefficient, C_D, becomes the lift coefficient, C_L, and the lift force becomes F_L. These equations hold true for steady-state flow where the velocity of both the fluid and body is constant, whether zero or non-zero. To experimentally determine the coefficient of drag, the equation can be rearranged to show C_D as the desired variable. This can be seen as follows.

$$C_D = \frac{F_D}{\frac{1}{2}\rho A u_r^2} \tag{2.3}$$

The coefficient of drag, and lift, are unfortunately non-constant values for non-steady-states and vary depending on two factors: the Reynolds number, and the angle of attack as seen in [8], [9], to name a few papers. Cao's representation of this can be seen in Figure 2.2.



(a) Pressure lift coefficient

(b) Shear lift coefficient

Figure 2.2: Lift coefficients from pressure and shear drag forces for various angles of attack and Reynolds numbers. Image from [9].

The Reynolds number, which is given as Equation 2.4 is dependent on ρ, the fluid density, u, the fluid velocity, L, the object's characteristic length, and μ, the fluid's dynamic viscosity.

$$Re = \frac{\rho u L}{\mu} \tag{2.4}$$

The angle of attack of the object is the orientation of the object to the direction of the fluid's motion. With this term in mind, equation 2.1 can therefore be rewritten as

$$F_D = \frac{1}{2}\rho C_D(t) A u_r (u_r \cdot A) \tag{2.5}$$

In this equation, C_D is now a function of time and $u_r \cdot A$ represents the velocity multiplied by the cross-sectional area of the object normal to the direction of the flow. It was also seen in [9] that the angle of attack effects the ratio of pressure drag to shear drag. At low angles, shear drag does most of the work, nearly 75% at $\theta = 0°$ while at high angles, the pressure drag does most of the work, nearly 95% at $\theta = 90°$. To determine the function of coefficient of drag, various experiments will need to be performed. By varying the fluid velocity and the angle of attack separately, a surface of coefficients can be determined. This will allow a quasi-steady-state analysis of a system. It has been unsaid thus far, but all this assumes the fluid is incompressible.

2.1.2 Added Mass Effect

While steady-state flow is relatively simple to analyse, the difficulties in fluid mechanics come in when acceleration is introduced. Not only does the object need to accelerate, but the fluid also experiences acceleration which introduces an extra term in the force equation. This concept can be seen in Figure 2.3 where the object acceleration in a fluid interacts with the fluid and causes the fluid to move alongside it, effectively increasing the mass of the object with the movement of the fluid. According to [10], it is therefore imperative to analysis a system in the unsteady-state. The concept of the acceleration of fluid is called the added mass effect. In the simplest view, added mass refers to the necessary work required to change the kinetic energy associated with the fluid [11].

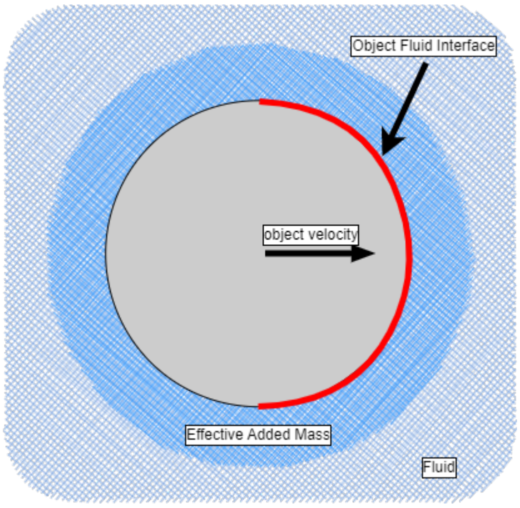


Figure 2.3: The added mass effect is the work required to change the kinetic energy of the fluid.

The added mass for simple objects is well known, however it can also vary for various reasons. For complex geometries, the added mass is more difficult to analyse, but a crude approximation for the added mass of an object for a given acceleration is to take the cross-sectional area of the object. For a sphere, the volume with the same cross-sectional area a non-sphere should then be halved [11]. While there are some better approximations and formulas associated with calculating the added mass [11], it has been seen that for a cylinder, the added mass is given as follows [11],

$$M = \pi R^2 \rho \, per \; unit \; length \tag{2.6}$$

This added mass for the cylinder is for a rectilinear motion undertaken by the cylinder. To the author's knowledge, the added mass of a cylinder rotating along its edge has not been investigated and published. This may be due to how complicated it becomes and so it is assumed that for rotational movements, rectilinear added mass values will be used.

2.1.3 Morison's Equation

These two concepts, the drag force and added mass can be combined to form the Morison's equation [12]. While this equation is traditionally used in hydrodynamics, it can be written in as follows as Equation 2.7 [13],

$$F_x(t) = F_D(t) + F_{vm}(t) = \frac{1}{2}\rho C_D A U(t)^2 + (m_b + m_f)a(t) \tag{2.7}$$

In this equation, F_D is the steady-phase drag, $F_v m$ is the virtual mass force which is the sum of the virtual mass of the body, m_b, and the fluid, m_f, multiplied by the acceleration of the objects over time, $a(t)$. This equation, however, does not account for the change in drag coefficient. This will have to be included in the final equation.

2.1.4 Basset Force

It has been seen that the history of a moving object also plays a role on the forces experienced by the object. This was first proposed by Boussinesq in 1885 [14] and Basset in 1889 [15]. A simple way to understand this force is by thinking of running your hand through water in a pool. At first, the wake is small but as time goes and your hand accelerates, the wake gets larger. This wake has additional forces depending on the size of it. It is therefore seen as a history force [13]. It was seen experimentally by [13] that the force experienced by their test plate did not immediately dissipate when the plate stopped accelerating but that it had a force that exhibited a damping factor. This Basset force was expressed as the term given below which is only experienced after the body has stopped accelerating at t_{sr} and can be seen in the full equation in Equation 2.8.

$$F_{mh} = \frac{dm_h}{dt} a(t)(t - t_{sr}) \tag{2.8}$$

This force is mainly used in hydrodynamics, though it is a general fluid dynamics equation. The force is equal to the product of the entrainment rate, $\frac{dm_h}{dt}$, the acceleration of the object, $a(t)$, and the time since the object stopped accelerating, $(t - t_{sr})$. The entrainment rate is the mean rate of increase of turbulent fluid in the flow direction [16].

2.1.5 Final Aerodynamic Force Equation

Combining the Morison equation, the Basset force, the change in coefficient of drag, and the added mass terms, the full aerodynamic force equation for a cylinder can be written as Equation 2.9.

$$F_x(t) = \frac{1}{2}\rho C_D(t) A u_r(t)(u_r(t) \cdot A) + (m_b + m_f)a(t) + \frac{dm_h}{dt}a(t)(t - t_{sr}) \tag{2.9}$$

While this may be the full equation, it is not the equation that this research will be using to model the characteristics of drag of the cheetah tail. The Basset Force, or the history force, will be removed when modelling the aerodynamics of the tail as it falls outside the scope of the research, and if it had to be included, the direction of the research would change. The final equation that will be used will be a representation of Equation 2.7 which is the Morison's equation.

2.2 Tails in Animals and Robotics

Robotic designs are often inspired by nature, trying to simulate and enhance what nature has used for millennia. This inspiration from nature which influences the design of robotics has been deemed bio-inspired robotics. One field of bio-inspired robotics has been that of the addition of appendages, such as the tail. An animal that has inspired this large field of study has been the flat-tailed house gecko, the Hemidactylus platypus. The gecko has been seen to use its tail to climb, glide, and to control its aerial attitude mid-air [17], [18]. Other studies have seen similar effects with lizards [19], [20]. The kangaroo's tail has also been seen to be used for active control in balancing and forward locomotion [21]. The basilisk lizard, known for being able to run on water, has been known to use its tail to balance by dragging the tail through the water [22].

2.2.1 Animal Tails

Tails in nature have developed for various reasons. Some tails have developed to be prehensile, such as the pangolin and most monkeys, while other tails have developed, not to grasp, but to aid in whatever the environment calls for. Arboreal species have developed large tails to help navigate the rough terrain found within the uneven canopy tops [23]. Terrestrial species' tails have also been found to be used in navigating the terrain by providing stabilisation in rapid manoeuvres [24].

Tails, spanning most species, are used for stabilisation effects. Where species differ is in the method of stabilisation. Even within an animal group, different effects are used [23]. Black-capped squirrel monkeys were found to use their tails as a static counterweight due to their long digits and high clasping ability. In the same study, cotton-top tamarins, a close relative primate, used their tails as a dynamic stabiliser rather than a pure static counterweight due to their short digits and their low clasping ability [23]. The kangaroo rat, as seen by [25], use their tail, not only as the third leg to balance on when reaching for objects above them, but also as both a static and dynamic stabiliser.

Geckos and lizards, two well studied animals, also use their tails for inertial stabilisation, both dynamic and static [17], [18], [20], [26]. An example of this can be seen in Figure 2.4 where the gecko uses its tail in a self righting aerial manoeuvre to rotate its body by 180°. These animals have all used inertial stabilisation relying on the mass and length of the tail however not all animal tails are long and heavy. This must mean that the animals use their tails for something other than inertial stabilisation. It has been postulated that they use aerodynamic effects, or a combination of the two for stabilisation. In a recent study on common grey squirrels, inspired by a viral YouTube video [27], it was seen that the squirrels, when launched at a non-zero angular momentum, stabilised their bodies using both inertial effects and aerodynamic effects induced by the bushiness of the tail. It was estimated, using a simplified model, that the aerodynamic torque accounted for 15% of the total torque [28]. This has also been seen in the animals like the cheetah [29] whose tail is around 2% of the body mass with the majority of the mass being at the base of the tail near the hip [24]. [8] proposed that the cheetah rather uses aerodynamic effects to enhance rapid manoeuvres rather than the purely inertial stabilisation. Another group of animals that uses the combination of inertial and aerodynamic effects are flying snakes [30]. While the snake does not have a tail per se, the shape and movement of the body is very similar to that of a prehensile tail which makes it interesting to see another animal use the same effects with different methods.

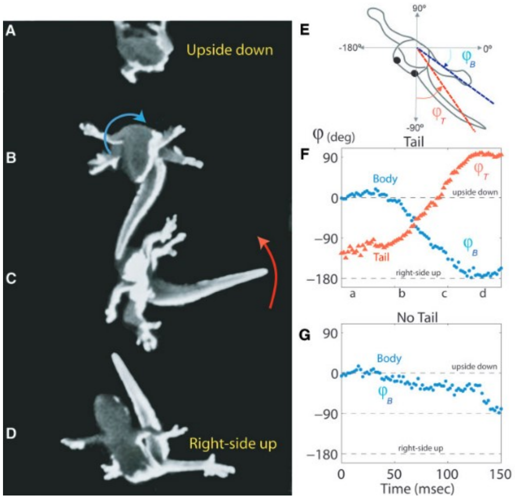
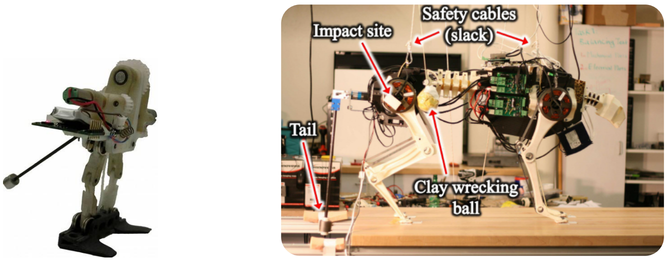


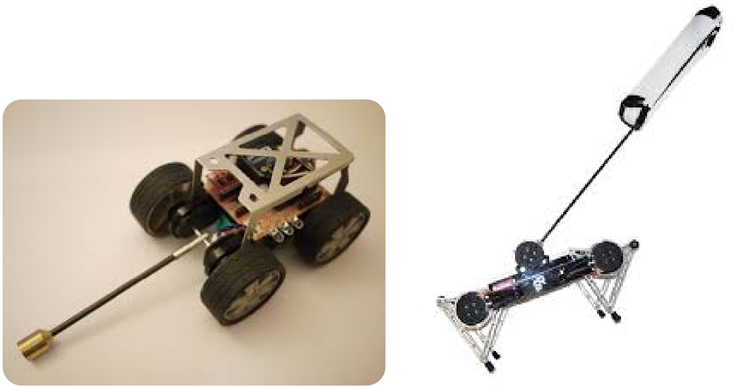
Figure 2.4: The gecko using its tail in aerial righting manoeuvres. Without the tail, the gecko can only turn 90° while with the tail, the gecko can turn its body by 180°. Image from [31]

2.2.2 Robotic Tails

These animal tails have been the inspiration of many robotic studies. Some studies have looked at and inspired robots in the following areas: hydrodynamic drag-induced balancing [22], hydrodynamic drag-induced steering control [26], robotic hopping control [21](*MSU Tailbot*) [32], inertial impulsive disturbance (*MIT Cheetah*) [33], aerial righting (*Berkeley Tailbot*) ([19]),([17])(*Ghost Robotics Minitaur*) [29], steering control (*SailRoACH*) [34], [2], and enhanced high-speed manoeuvrability (*Dima* I & II)[8], [35]. These study the addition of robotic tails, active or passive, to enhance performance of the robot. Various tails designs have been used from point masses on massless rods for inertial effects to static plates and sails to tails designed to maximise aerodynamic drag.

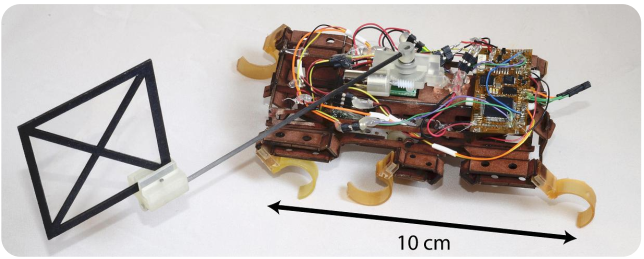


(a) *MSU Tailbot* **(b)** *MIT Cheetah*



(c) *Berkeley Tailbot* **(d)** *Ghost Robotics Minitaur*

(e) *Dima* I



(f) *SailRoACH*

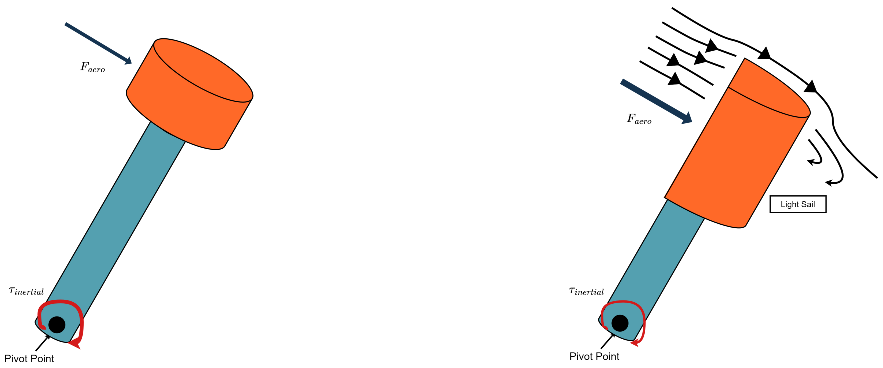
Figure 2.5: Various robots with inertial and aerodynamic tails. Images taken from [32], [33], [19], [29], [8], [2] respectively.

2.2.3 Steering Mechanisms

While this research aims to increase the manoeuvrability of the robot using a tail, this, as will be seen later, is done through the using of being able to steer well. The majority of terrestrial robots have used three different methods to enable steering. These methods are the differential drive, the addition of flywheels, and the addition of tails. It was seen, when comparing inertial tail steering to differential steering, that differential steering is more effective for sustained turning on high friction surfaces while tails are effective for high speed turning or low friction surfaces [34]. The turning range with a tail was limited to the motor range and contact points. This was also seen in aerial righting manoeuvres performed in [29]. The other steering method are momentum wheels or flywheels. While these are not limited by contact points or motor range, they have been seen to be ideal for applications for slow and precise steering adjustments. For dynamic steering, a tail is more compact, lighter and has a faster response time that a flywheel [19]. For high-speed manoeuvres, tails are more effective turning mechanisms than differential steering or flywheels. This effect is verified by nature's use of tails rather than the other methods.

2.2.4 Inertial versus Aerodynamic Tail

While it has been shown that, for the purpose of high-speed manoeuvres, tails are the most effective, there are various tails that could be incorporated. These tails range between active and passive tails, and inertial and aerodynamic tails. Passive, or static, tails have mainly been used for balancing [22], but that have also been used as the basis for aerodynamic tails as is seen in the SailRoACH [36]. The inertial tail by nature requires an active tail. As is seen in [29], inertial torque does not continue to act on the body when the tail is stationary. The inertial tail, unlike differential steering and flywheel steering, can only make a single consecutive turn in the same direction without resetting the tail [2]. This problem is either solved by having a slow reset of the tail, by having an unlimited range of motion or by having some sort of constant torque applied to it. This is made possible with the use of an aerodynamic tail. Aerodynamic tails are inspired by creatures such as the cheetah, the giant Indian squirrel, and the jerboa whose tails have low masses and yet use their tails in controlling their manoeuvres. Due to aerodynamic drag and how it is quadratically proportional to the relative velocity between the air and tail, a tail designed to maximise aerodynamic torque can apply a constant turning torque. This alleviates the issue of an active inertial tail not being able to provide a constant turning torque. The drag force applied to the body is not dependent on the mass of the system as seen in Equation 1 and developed in [29] which means that the tail can be optimised to be as light as possible without effecting the performance of the design unlike the inertial tail where the torque is proportional to mass. It was seen in [29] that having an active aerodynamic tail helped accelerate the robot 12% faster than without the tail, despite an increase in mass of the system. In [36], a passive aerodynamic tail helped the robot accelerate 9% faster than without an aerodynamic tail when positioned rearwards. This was supposedly due to the tail acting as a stabiliser which mitigated the yaw disturbances it would have otherwise experienced. It was seen that aerodynamic tails work best by making use of inertial and aerodynamic torques. The robot in [29] experienced an increase in linear acceleration of 2% when incorporating aerodynamic and inertial effects as opposed to pure inertial effects. The difference in concepts between an inertial tail and an aerodynamic tail can be seen in Figure 2.6. In the figure, the inertial tail has a large torque due to the inertia of the effective point mass, while the aerodynamic tail has a large torque due to the high aerodynamic forces acting on the tail.



(a) Inertial Tail **(b)** Aerodynamic Tail

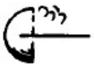
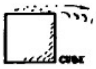
Figure 2.6: Representation of inertial and aerodynamic tails respectively. The inertial tail has a large inertial torque and, being an object passing through a fluid, has a small aerodynamic torque. The aerodynamic tail has a large aerodynamic torque and, being a light mass, has a small inertial torque.

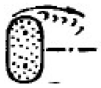
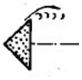
2.2.5 Aerodynamic Tail Design

When trying to design for a tail that uses both inertial and aerodynamic effects, there are a few things to consider. The first is the mass. To design the tail to have a large mass at the end of the tail would be designing an inertial tail. The design should therefore minimise the mass so to conjoin the aerodynamic and inertial effects. This will also reduce the total mass of the system, increasing maximum velocity that the robot would be able to attain. As was seen with *SailRoACH*, to maximise aerodynamic turning, the yaw moment coefficient should be maximised while minimising the drag coefficient [36]. While it may seem counter-intuitive to minimise the drag coefficient, it essentially describes minimising the reaction force in the direction opposite to the forward velocity. This force would reduce the ability for the robot to accelerate quickly. Maximising the yaw moment coefficient describes maximising the turning moment of the robot. These two considerations fall in a tricky situation where increasing one means increasing the other, to the detriment of itself. The design will need to be optimised to find the correct balance. The size of the robot also plays a large role in the ability to use aerodynamic steering. Two studies have looked at this and have stated complementary but different results. One paper showed that the steering's effectiveness scales inversely with robot size squared, meaning that smaller robots are better equipped to use this method [36]. This effect is seen in nature by smaller stick insects using aerodynamic steering methods [18]. The other paper suggested that aerodynamic drag tails can be effective at any scale, although other effects may be more useful at smaller scales [29]. This effectiveness is seen, or rather still hypothesised, in the cheetah and the use of their tail at high speeds [8]. It was seen in [29] that the tail length is the most important aspect of the tail design. If the tail width scales with length, then the torque is proportional to L^4. When comparing a long tail compared to a short tail case, the long tail was almost four times as effective. Along with length, the aerodynamic torque scales linearly with width and the drag coefficient which means that the optimal design has a maximised width of the tail. Due to physical constraints, a design call will have to be made by the researcher regarding the specifications of the tail. It was also seen in [29] that while the optimal distance between the body and the start of the sail portion of the tail, the shaft length, is 0.6 m. Due to the quartic scaling, the torque is only significantly reduced for shaft lengths close to the total tail length. Therefore, the shaft length can be chosen to be any value so long as the sail produces enough torque to achieve the purpose of the tail. While it has not been done, according to the author's knowledge, it may be a design consideration to be able to control the tail's normal direction relative to the shaft. To maximise the torque, the sail should face into the direction of motion of the tail, assuming that the body is stationary. At high velocities, the sail should be angled at an optimal angle to the direction of the body and the direction of the tail. This has all the makings of a difficult control problem to overcome.

Once the metrics of the tail are decided upon, the shape of the sail will need to be chosen. This field of study, the study of coefficients of various shapes, has been saturated with all the papers written about the coefficients of various shapes. An example of different drag coefficients can be seen in Table 2.1

Table 2.1: Various drag coefficients from different shapes. The section on the left illustrates the drag coefficients of 3D shapes, while the section on the right illustrates the drag coefficients of 2D shapes. This table's values and figures were taken from [37].

3D SHAPE	C_D
	0.47
	0.38
	0.42
	0.59
	0.80
	0.50
	1.17
	1.17
	1.42
	1.38
	1.05

2D SHAPE	C_D
	1.17
	1.20
	1.16
	1.60
	1.55
	1.55
	1.98
	2.00
	2.30
	2.20
	2.05

2.2.6 Manoeuvrability

According to [38], to manoeuvre is to make a decisive change in direction and position of the body. [39] adds to that definition to include the acceleration and deceleration required for locomotion, turning, discontinuous movements, and attitude transitions. Manoeuvrability is seen in all animals, some to a greater extent than others. It has been seen that cheetahs, the fastest recorded land animal, are not successful in their hunts because of their speed but because of their ability to manoeuvre effectively [4], [40]. While cheetah's have been recorded to run at 29 m/s [3], the fastest successful hunting speed was recorded at only 18.94 m/s [4]. The ability to measure manoeuvrability, however, can be a bit more difficult to get a grasp on as various manoeuvrability metrics can be used to quantify the manoeuvrability of something. It will be looked at, in the research content, what manoeuvrability metrics will be looked at when analysing the manoeuvrability of *AeroDima*.

Chapter 3: Methodology

This chapter looks at the research process taken throughout this research. The holistic purpose of this research topic is to develop a robotic platform that uses a robotic aerodynamic tail to aid it in high-speed manoeuvres. The purpose, then, of this chapter is to provide the how and why behind the research steps taken. When testing the robotic tail, the focus will be on the validity of the use of a tail rather than the design of a traditional control system. This means that this research will aim to prove that the tail, even with only a basic control system, will be able to benefit said robot in terms of certain metrics of manoeuvrability. For this to occur, the research is split into three main sections; that of the analysis of the cheetah tail, the design and build of *AeroDima*, and the physical and simulation testing and analysis of *AeroDima*.

3.1 Analysis of the cheetah tail

This section looks to prove two main concepts. The first is that the cheetah tail experiences significant enough aerodynamic forces to verify the use of aerodynamic forces on a cheetah tail. This will show the validity of an aerodynamic tail design over the use of an inertial tail design, as done in the previous *Dima* iterations. The second is to determine a suitable aerodynamic model that the tail uses for which to use for the modelling of the tail in a simulation state.

This is achieved through experimentation using three cheetah tails to produce a range of data that will aggregate to the final results, and a wind tunnel. These tails are obtained through the means of a necropsy which meant that no cheetahs were harmed in the process of this research. The fur of the tails were removed and attached to a steel rod with a rubber type lining to allow for adhesion. This process was done for previous research studies [41] and so that process will not be described in this paper. With the preparation of the tails done, an experimental rig will need to be designed to perform the research. The wind tunnel, a low-speed closed return devices situated in the Duncan McMillan Laboratory in UCT, was used here.

To get a full model of the tail, both static and dynamic testing procedures will be performed. The static procedures will allow for the development of a quasi-steady-state model (QSS), while the dynamic procedures will allow for the verification of the QSS model under movement and will allow for testing of additional components to the drag forces. An additional dynamic component will be looked at, that being the added mass effect. This is term in the equations describes the forces on a body in a fluid that is due to the acceleration of the fluid around the body. This effect is not modelled in the QSS model due to the static nature of it and the aim of the dynamic tests is to see whether the addition of this effect is necessary for the modelling of the tail. These models and results will be developed using MATLAB and their Simulink package. Due to the mathematical aspect of these models, the full models will not be shown in this paper, but the results will be shown and discussed [*]. The model will,

---

[*] The model developed can be found here: https://github.com/Darynbright/AeroDima-Added-Info

however, be developed using the Euler-Lagrangian dynamics methods as this allows for the system to be modelled as physical bodies using the equations of motion. The final model produced at the end of the analysis will guide the design of the aerodynamic model for the simulation of the robotic tail.

3.2 Design and build of AeroDima

This part of the research encompasses the actual design of *AeroDima*, both the physical and the simulated models. Included in this section will be the design of the tail, the design of the robot, the design of the control system, and the design of the simulation. The rough flow of the design can be seen in Figure 3.1. The skeletal structure of the *AeroDima* will be taken from and inspired by *Dima* II. The *Dima* platform used a store bought remote controlled (RC) vehicle and was adapted to fit its purposes and in the same vein, *AeroDima* will do the same thing by disassembling *Dima* to make room for *AeroDima* to be built off. The modelling of the platform will be done in SolidWorks, a commonly used CAD modelling software. The design will need to incorporate enough space and mounting points for the electronics, the batteries, the motors, and the tail. They will need to be housed in a safety structure, namely a roll cage, to prevent damage to the vehicle when the vehicle ends up flipping for various reasons.

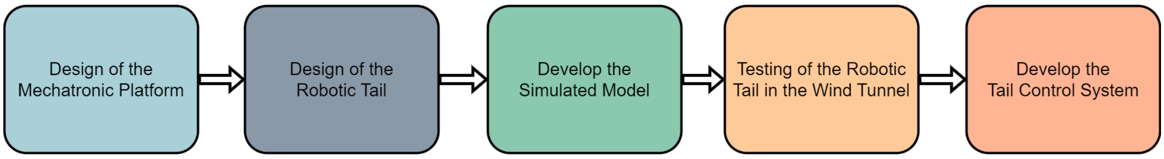


Figure 3.1: Flow diagram of the design and build of *AeroDima*.

The tail too will need be designed to maximise the aerodynamic forces experienced by the tail. Numerous designs will be tested against one another to choose the design with the largest aerodynamic coefficients by using SolidWorks' Computational Fluid Dynamics (CFD) Flow Simulation package. The whole design process will be done to allow the tail to exert as high an aerodynamic torque on the vehicle as possible. Once the design is complete, it will have to be manufactured using either the on-site workshop where required or having to have the manufacture of some parts outsourced.

With the hardware designed and manufactured, the software side of the platform will need to be done. This will involve the development of communication protocols to communicate between the various microcontrollers and motors, as well as the development of the logging of the data from the sensors and the user interface for the control of the vehicle.

After the platform becomes operational, the simulated platform will be modelled in MATLAB's Simscape Multibody package. Using the .STL 3D files derived from the SolidWorks models, the platform will be modelled in a 3D environment with each component's mass and inertia taken from the SolidWorks model. At this stage, an aerodynamic force model associated with the robotic tail will need to be developed to ensure a realistic simulation. To do this, the robotic tail will be put into the wind tunnel used for the cheetah tail analysis, the tail will be swung in various scenarios, and the results will be logged. These data sets will be imported into the Simscape tail model and the model will be tuned to match the data. This allows for the drag coefficients to be determined and the model to be validated.

Finally, the control system of the tail will be designed, and while the control system is not the main aspect of this research as the main focus of this research is the tail and its effect on the platform, it is an important aspect . Therefore, the control system will be designed to be a simple state-based controller that alternates between a PD position and PD velocity controller to allow for the tail to be swung in a controlled manner and to test the performance of the tail on *AeroDima*.

3.3 Testing and Simulation

With everything designed, manufactured, and modelled, the physical testing and running of the simulation remained. The following discussed methods of testing will be performed on both the platform and the simulation to verify the simulation against the platform, but also so that the simulation can be used to test multiple scenarios in a shorter, and safer, time frame. The simulations, taking only computational intensity, provide no danger to the real robot and so can be used to test the limits of the robot. Therefore, the simulations were run first, followed by the physical testing of the platform.

The first of the two experimental methods will be looking at setting the base performance of the vehicle by testing and logging the tests of *AeroDima* without the added tail. The test will be done by driving the vehicle in a straight line and then performing a turn where a successful test will look something like Figure 3.2.

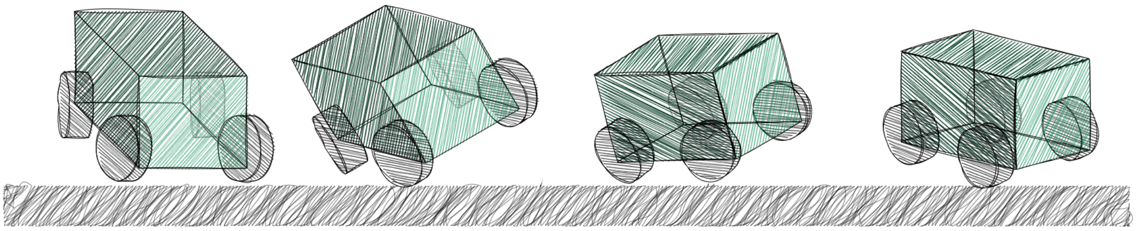


Figure 3.2: Illustration of a successful turn of the tailless *AeroDima*.

This manoeuvre will be repeated until the vehicle flips over to allow for the base performance to be measured. An illustration of a failed turn is shown in Figure 3.3. and this procedure will be done for both the physical platform and the simulation.

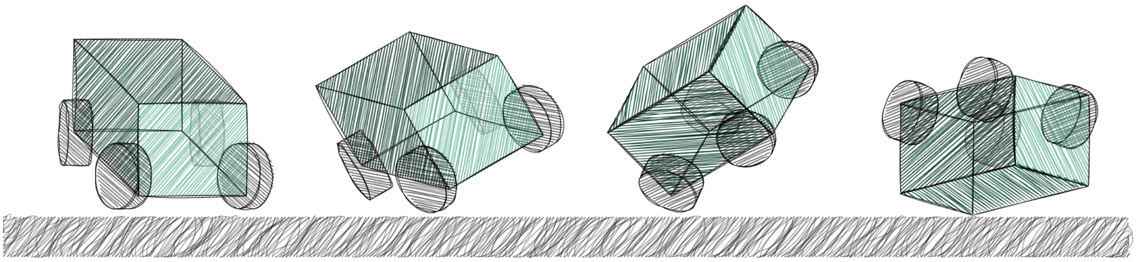


Figure 3.3: Illustration of a failed turn of the tailless *AeroDima*.

With the aforementioned tests performed, the tail will be added to it and the same tests will be performed, except this time with the tail being swung during the motion. An illustration of this is shown in Figure 3.4.

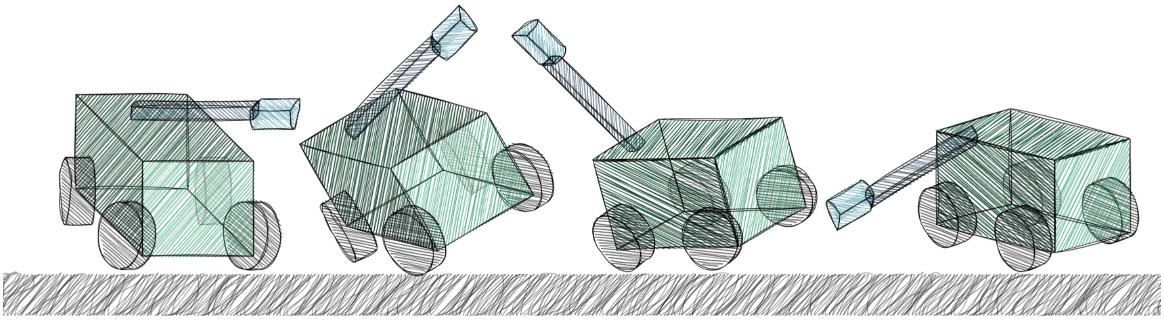


Figure 3.4: Illustration of a successful turn of a tailed *AeroDima*.

3.3.1 Manoeuvrability Metrics

With these tests, eight metrics will be looked at to measure the performance. Those eight metrics being: maximum and the average linear acceleration, the maximum and the average yaw rate of the vehicle, the maximum and the average Kohut manoeuvrability metric, the integral of the roll rate, and the top speed of the vehicle. The configuration for these metrics can be seen in Figure 3.5 along with the orientation of the axis upon which they're measured.

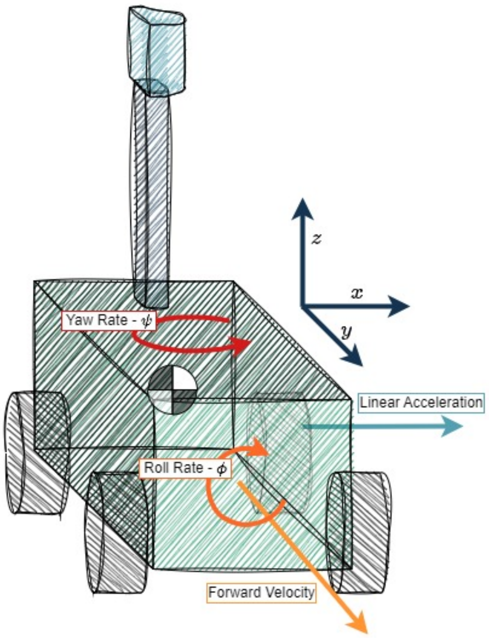


Figure 3.5: Configuration of the metrics used to evaluate the performance of the vehicle.

For both the maximum and the average metrics, the data set was sampled during the turning motion. The linear acceleration was chosen as a metric due to the centrifugal forces that will be exerted on the vehicle during the turn. The centrifugal forces are dependent on both the mass of the vehicle, the speed of the vehicle, and the radius of the turn. This relationship can be seen in Equation 3.1 where the m, v, and r are the body's mass, forward velocity, and the radius of the turn.

$$F_{centrifugal} = \frac{mv^2}{r} \tag{3.1}$$

The yaw rate was chosen as a metric due to the fact that the vehicle will be turning. This means that the vehicle will be rotating about its vertical axis. The yaw rate will show how much the vehicle turns in the turning motion. The Kohut manoeuvrability metric was chosen as a metric due to the fact that it is a metric that is used to measure the manoeuvrability of other robotic platforms as seen in [2] and [36]. This was actually initially developed by McClung [1], but will be named in this paper as the Kohut manoeuvrability metric due to the frequency at which it is used in Kohut's papers. The metric is the product of the angular velocity ($\dot{\theta}$ in $^{\circ}/s$) and the forward velocity (v in m/s) and is given by the constant, K. From here onward, it shall be referred to as the Kohut metric and is defined by the Equation 3.2.

$$K = \dot{\theta} v \tag{3.2}$$

The same paper where the Kohut metric was used as a defining metric defined compared the manoeuvrability of several legged robots. A few of the robots used non-tail methods for steering purposes and so were omitted from the comparison table shown below. The robots performed turning manoeuvres using inertial tails except for the SailRoACH (Aerodynamic) which employed aerodynamic means of turning. The metrics can be seen in Table 3.1.

Table 3.1: Kohut Metric of various robots. Table adapted from [36]

Robot	K = $\dot{\theta}v$ ($\frac{°m}{s^2}$)
OctoRoACH ([34])	36
TAYLRoACH ([2])	108
SailRoACH ([36])	360
SailRoACH(Aerodynamic) ([36])	134

Two other metrics that will be looked at will be the roll rate stability and the capable forward velocity as these metrics are useful in determining the performance of the turning capabilities of the robot during high-speed manoeuvres. All of these metrics, apart from the stability and the forward velocity can be recorded in two ways. The first is an average reading which looks at the average of the metric over a period of time. The period of time for a desired reading would be the period at which a turning manoeuvre is performed. The second way of recording the metric is a maximum instantaneous metric that looks at the maximum value that the metric experiences during the turning manoeuvre. The roll rate stability metric is a metric that looks at keeping the robot level during the manoeuvre and is calculated by taking the integral of the roll rate of the robot. The metric is calculated as can be seen in Equation 3.3 where $\dot{\phi}$ is the roll rate of the body.

$$Stability = \int_{t_0}^{t_f} \dot{\phi} dt \tag{3.3}$$

The last metric is the forward velocity that the vehicle can experience when performing the manoeuvre without failing the manoeuvre.

3.3.2 Summary

In summation, the following was done in this chapter. The methodology that the research took was laid out in three main sections: the analysis of the cheetah tail, the design and build of *AeroDima*, and the testing and simulation of *AeroDima*. The analysis of the cheetah tail section elaborates on the plans on performing wind tunnel tests on real cheetah tails to develop a quasi-steady-state aerodynamic model using the Euler-Lagrangian dynamics method. The design and build section looks at the procedures that will be taken place to do the full design and development of *AeroDima*, including the simulated model done in MATLAB's Simscape Multibody package. Lastly, the testing and simulation section looks into the means of testing both the robotic and simulated platform as well as giving a brief overview on the metrics that will be used to analyse the performance of the platform.

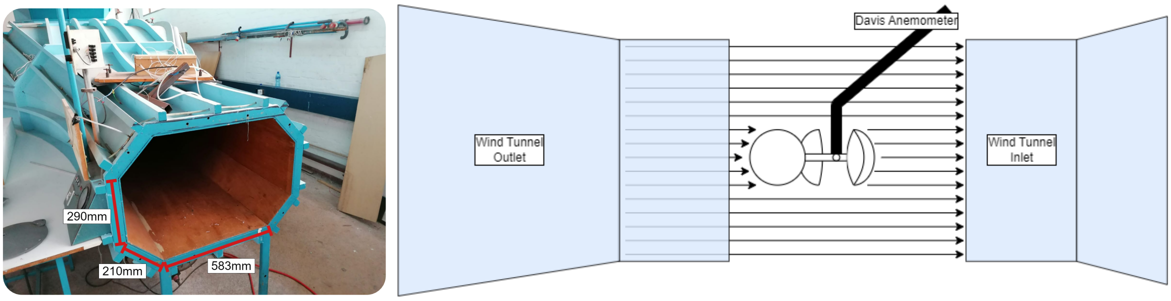
Chapter 4: Cheetah Tail Analysis

This chapter details the analysis of the cheetah tail, looking from the experimentation process till the data analysis. The detailed methodology used to conduct the research employed in this chapter is described as well as the results of the analysis. The aim of this chapter is to provide clear insight as to how the research went about answering the question of what the aerodynamic characteristics of a cheetah tail are while proposing a mathematical model than can be used for both static (and dynamic) motions. The chapter is broken down into three main sections. The first section is the experimental setup and data collection which details the design of the experimental setup and the data collection process. The second section is the detailed development of the mathematical model of the tail while the third section is the data analysis which details the analysis of the data obtained from the experimental setup.

The aerodynamic properties of the cheetah's tail has been studied to some degree, however an accurate dynamic model was not obtained due to the various assumptions used by the authors. In that research, it was assumed that the aerodynamics of the tail produced an effective area of a truncated cone where the fur of the tail acted as an increased surface area with no mass penalty [8]. This research analyses the aerodynamics of the cheetah's tail to produce a more accurate model of the tail, aiming to reduce some assumptions used in the original study. This study also looks into the dynamic movement of the body of the tail producing a relative acceleration term. A quasi-steady-state model (QSS) will be implemented in which values for the aerodynamic coefficients will be obtained with no relative fluid acceleration. Upon the model being developed, a dynamic model will be implemented in which the aerodynamic forces associated with acceleration, the added mass effect, will be included. This aerodynamic model, a quasi-steady-state added mass model (QSS-AM) will be developed with the use of a wind tunnel and the simulation of the model done in MATLAB's Simulink using Euler-Lagrangian dynamics. A suitable aerodynamic model will be chosen based on the results of the analysis.

4.1 The Wind Tunnel Setup

To experimentally determine the aerodynamic characteristics of a shape, a wind tunnel can be used. In the following experiments, UCT's Duncan McMillan Laboratory was used which houses a low-speed closed return wind tunnel as can be seen in Figure 4.1a which was capable of reaching wind speeds of up to 30 m/s. The wind speed at the outlet of the tunnel was measured through the use of a Davis anemometer as the internal anemometer was not functioning. The wind tunnel was used to pump air across the body of the tail to determine the aerodynamic characteristics of three cheetah tails.



(a) Wind Tunnel

(b) Anemometer and Wind Tunnel setup.

Figure 4.1: The figure shows the wind tunnel equipment. Subfigure (a) shows the low-speed closed return wind tunnel in the Duncan McMillan Laboratory while subfigure (b) shows the method of obtaining the wind velocity with a Davis anemometer. The anemometer is not drawn to scale.

4.2 The Tails

The tails used in this experiment are made from taxidermied fur taken from tails of cheetahs obtain from a necropsy of three cheetahs that had died of natural causes. The fur, wrapped around a threaded steel rod to form a stiff tail, for each of the tails can be seen in Figure 4.2. Each tail was given a number to identify it from 1 to 3 and the masses of each tail were obtained by weighing them on a force plate. Each tail was seen to have an elliptical shape, which was assumed uniform across the length of the tail with a larger diameter in one axis, the *major axis*, and a smaller diameter in the other, the *minor axis*. The tails were measured using a Vernier Caliper at four places split equally along the length of the tail in each axis. These measurements can be seen in Table 4.1.

Figure 4.2: The three tails used in the wind tunnel experiment to determine the aerodynamic characteristics of the cheetah's tail. The tails from left to right are tails 1, 2, and 3, respectively.

Table 4.1: Widths of the tails at various points along the length of the tails.

Tail	**Reading 1** (mm)	**Reading 2** (mm)	**Reading 3** (mm)	**Reading 4** (mm)
Major Axis 1	64.0	63.0	66.4	62.3
Minor Axis 1	49.3	45.2	36.6	49.6
Major Axis 2	56.9	48.4	50.7	50.0
Minor Axis 2	41.3	34.1	34.0	34.0
Major Axis 3	39.2	39.2	39.1	39.1
Minor Axis 3	48.9	33.5	33.5	33.5

Using these measurements, the tails were then assumed to be uniform ellipses with the diameters taken to be the mean of the four readings as represented in the Figure 4.3. The final dimensions, masses, and inertias of each tail are given in Table 4.2.

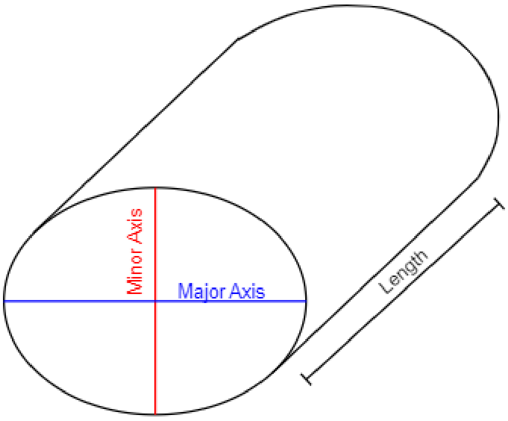


Figure 4.3: A representation of a tail as a uniform ellipse with the major and minor axes and the length illustrated.

Table 4.2: The properties of the three tails used in the wind tunnel experiment.

Tail	**Length**(mm)	**Major Axis** (mm)	**Minor Axis** (mm)	**Mass** (g)	**Inertia** (kgm^2)
1	390.0	63.9	45.2	98.2	0.01203
2	365.0	51.5	35.9	97.5	0.01262
3	365.0	39.2	37.4	93.7	0.01273

4.3 Experiment Rig Design

The design of the testing rig was twofold, a mechanical design and an electrical design. The mechanical design was done with the aid of SolidWorks 2020. Included in the design was a 6-axis force sensor, an Axia80 M20, that can read forces up to 500 $N(F_{xy})$, 900 $N(F_z)$, and 20 $Nm(T_{xyz})$. Mounted on the force sensor was a baseplate that the motor, a Maxon DCX35L Motor with a 26:1 gear ratio, was mounted on. Attached to the motor was a steel bar that extended to the tail where inside that bar was a 3D-printed interface that allowed the tail to be mounted to the bar. Attached to the motor mount was a 3D-printed protractor that aided in setting the pitch angle of the tail. The setup of the mechanical design is shown in Figure 4.4.

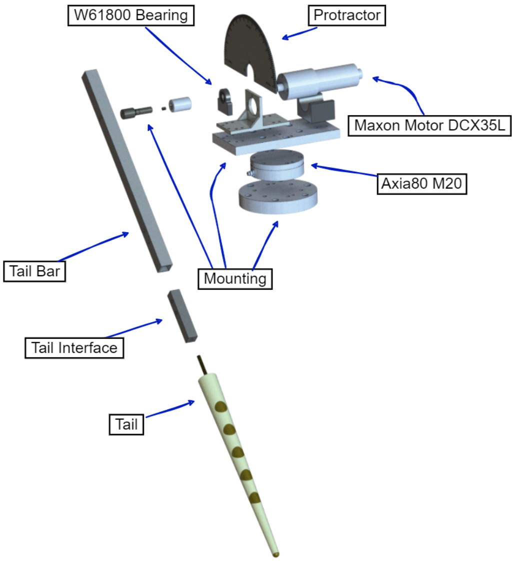


Figure 4.4: Exploded view of the experimental rig to be used with the wind tunnel.

The embedded design made use of multiple components that are listed and shown in Table 4.3 and in Figure 4.5. In the system, the Speedgoat made use of a Simulink Real-Time model that used it an external target machine to log the data obtained through the force sensor and the Teensy. An Ethercat communications protocol was used between the force sensor and the Speedgoat, while serial communication was used between the Speedgoat and the Teensy. The Teensy acted as a buffer microcontroller and was used to read the current and position of the motor, obtained through the Escon 50/5 motor controller module, as well as to read the digital I/O pins of the STM32F0 UCT Dev Board which was then sent to the motor controller to control the tail's motion. The STM32F0 UCT Dev Board was used to read and display the wind speed measured from the Davis Anemometer. The STM32F0 UCT Dev Board could not inherently communicate with the Speedgoat while the Teensy had no visual display or physical I/O buttons to manually operate the system. Therefore, both boards were used to ensure the system operated the way it was intended. The setup of the embedded design is shown in Figure 4.5.

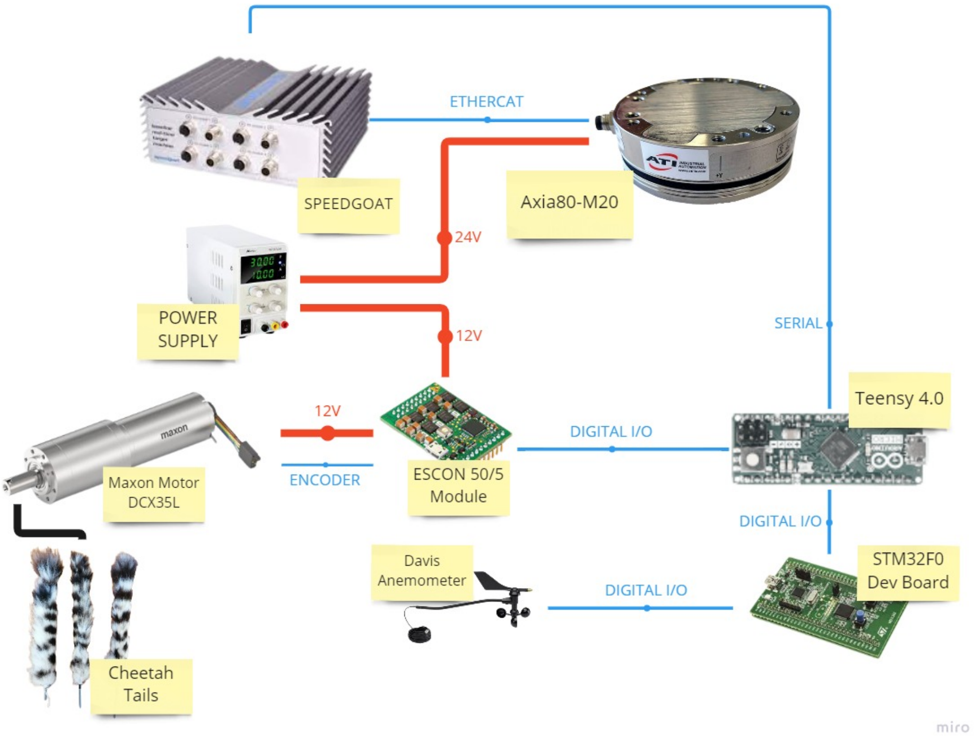


Figure 4.5: Wind Tunnel Test Rig

A comprehensive list of equipment used in embedded design of the wind tunnel experiment is shown in Table 4.3.

Table 4.3: Wind tunnel test embedded equipment list.

Component	**Details**
Force Sensor	Axia80 M20 6 Axis Force Sensor
Power Supply	12V, 24V
Data Logger	Speedgoat
Motor Controller	Maxon ESCON Module 50/5
Motor Encoder	Maxon ENX16 EASY 512IMP
Planetary Gearhead	Maxon GPX42 26:1
12V Brushed DC Motor	Maxon DCX35L GB KL
Anemometer	Davis Anemometer
Buffer Microcontroller	Teensy 4.0
Embedded Microcontroller	STM32F0 UCT Dev Board

4.4 Methodology

The experiment was divided into two procedures, the static and the dynamic procedure. The static procedure was observed in the following steps:

1 Set up the test rig so that the tail's major axis is facing into the wind tunnel with its rotation axis in the pitch axis.
2 Calibrate the readings to account for any offsets from the sensors.
3 Set the wind speed to 10 m/s.
4 Set the pitch angle of the tail to 0°.
5 Get 5 readings of a 20-second duration each.
6 Set the pitch angle of the tail to 15° and repeat step 5. Repeat for the following angles: 30°, 45°, 60°, and 75°.
7 Set the wind speed to 20 m/s and repeat steps 5 and 6. Repeat for the following wind speeds: 30 m/s.
8 Rotate the tail such that its minor axis is facing into the wind tunnel and repeat steps 2 to 7.
9 Repeat steps 1-8 for the other two tails.

The result of this static procedure was to determine the aerodynamic coefficients of drag and lift for the tails at incrementing angles of attack and wind speeds. By obtaining a grid of coefficients, a QSS model could be obtained through linearising the nodes. The more nodes the data set has, the more accurate the QSS model, but it was chosen that 30 nodes would be sufficient to produce a reliable QSS model. An example of the tail at 90° is shown below in Figure 4.6.

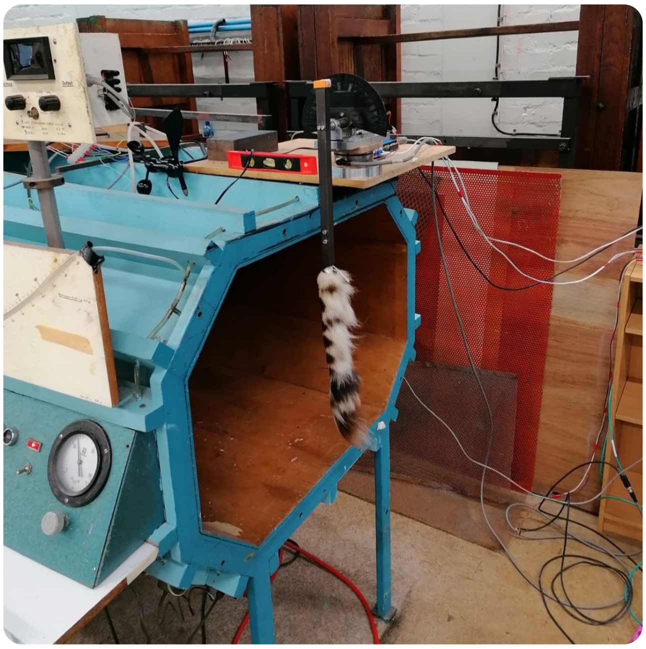


Figure 4.6: Wind tunnel test rig setup for the static procedures

%

With the static procedure previously outlined, the dynamic procedure can be outlined as well in the following steps:

1 Set up the test rig so that the tail's major axis is facing into the wind tunnel with its rotation axis in the pitch axis.

2 Calibrate the readings to account for any offsets from the sensors.

3 Set the wind speed to 10 m/s.

4 Set the pitch angle of the tail to 90°.

5 Get 5 readings of a 90° flick.

6 Set the wind speed to 20 m/s and repeat the previous step. Repeat for the following wind speeds: 30 m/s.

7 Rotate the tail such that its minor axis is facing into the wind tunnel and repeat steps 2 to 6.

8 Repeat steps 1-7 for the other two tails.

The result of this dynamic procedure was to determine the effects of relative acceleration and the added mass effect on the QSS aerodynamic model of the tail. For the dynamic tests, the tail was swung through a 90° angle in the pitch axis, directly into the wind. The results of the dynamic tests will be compared to the results of the static tests to determine the effects of the added mass effect on the QSS aerodynamic model of the tail to create a QSS-AM model. The motion of the tail can be seen in Figure 4.7.

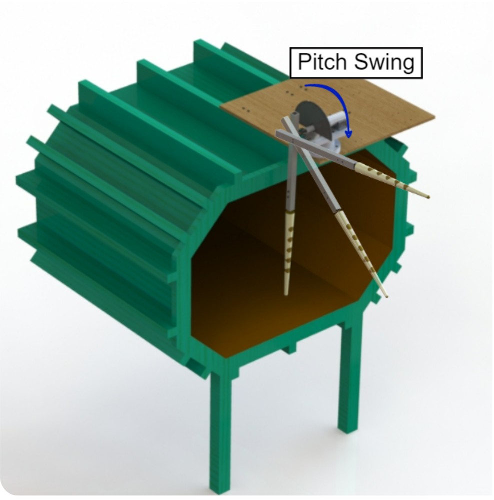
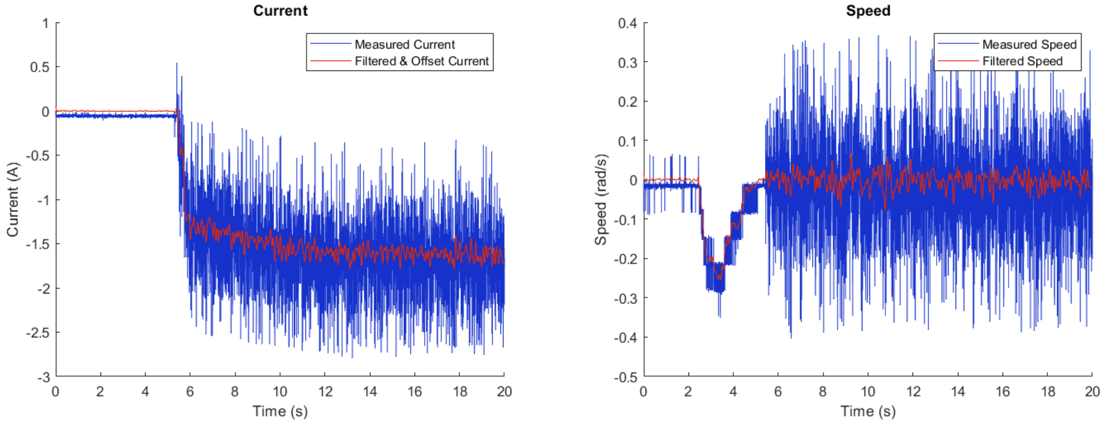


Figure 4.7: Render of the wind tunnel test rig showing a pitch swing of the tail.

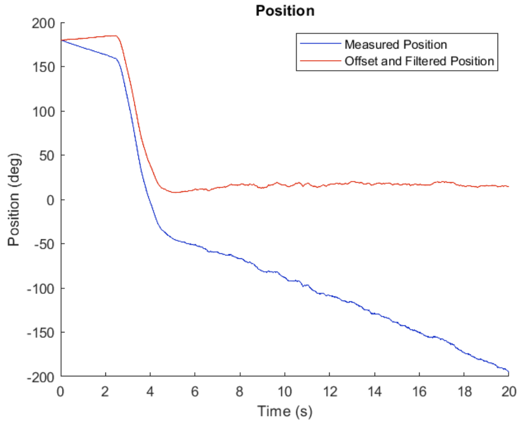
4.5 Logging and Filtering Experimental Data

The data was obtained and logged through the use of the Speedgoat target machine. The data recorded included the motor speed and the motor position from the motor encoder, the motor current from the motor controller, and the three forces and three torques from the force sensor. Each sensor was sampled at 1000 Hz. The data was then filtered using a 6th order Butterworth lowpass digital filter with a cutoff frequency of 10 Hz. The data had an initial offset that had to be zeroed before data analysis. Due to sensor error, there was a large drift that occurred in the measurement of the motor position data. Prevention of this drift was attempted in various manners. It was seen that the drift decreased as the sensor reached standard operation conditions, however this did not completely remove the drift of the measurements. Other attempts involved mounted it with different methods for thermal insulation and mechanical obtrusion. However, this did not fix the issue of the large drift that occurred in the sensing of the data. Therefore, this drift was accounted for and removed as best as possible in the final data based off of the average final position held by the static tail. This was achieved with the use of a line of best fit through the data taken from a time after the tail had been stationary for a few seconds. The data can be seen in Figure 4.8 below. Looking at the graphs, there is an evident movement of the tail at, in this example, around 2 seconds. This movement is induced by the researcher who moved the tail from a static stationary position where the tail was held vertically up to its respective angle. This can be seen by the fact that the current measured only indicated a current around the 5-second mark where the speed of the motor had already stopped. Due to this reason, the data analysed was only taken from the time after the tail had been stationary for a few seconds.



(a) Raw and Filtered Static Current Readings from the ESCON 50/5 Motor Controller

(b) Raw and Filtered Static Speed Readings from the Motor Encoder



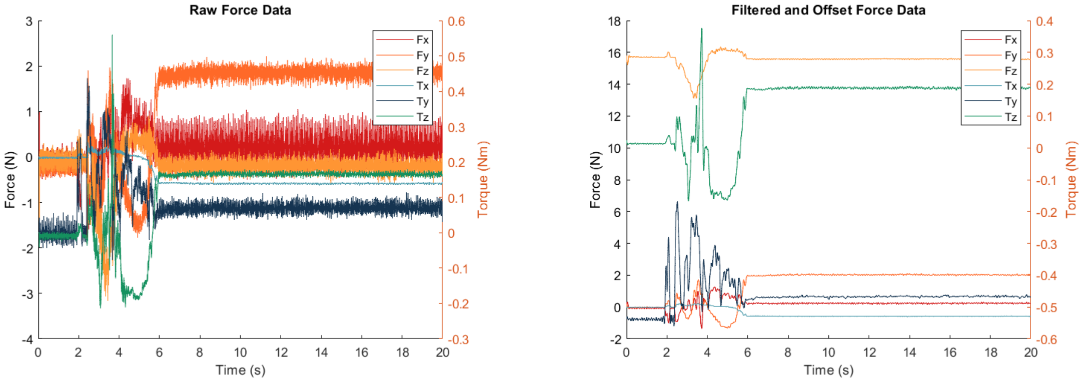
(c) Raw and Filtered Static Position Readings from the Motor Encoder

Figure 4.8: Raw Data and Filtered data obtained through the static tests. The data was filtered using a 6th order Butterworth lowpass digital filter with a cutoff frequency of 10 Hz. The figure shows the current, speed, and position data from (a) to (c) respectively.

The force sensor also had an offset zeroing at the beginning of each logging and so the initial offset, based on the dimensions and the mass of the system, was not accounted for as the readings were zeroed after every run. The offset for each reading was later added depending on the offsets calculated per tail. These offsets were obtained using static tests where the rig was only mounted onto the force sensor after the internal offset was dealt with. The numerical values of the measured offsets added can be seen in Table 4.4. The raw force data obtained can be seen in (a) of Figure 4.9 while the filtered and offset data can be seen in (b) of the same figure. The forces measured are plotted against the left y-axis while the torques measured are plotted against the right y-axis.

Table 4.4: Force sensor offsets for each tail for each force and torque reading

Tail	**Offsets**					
	$F_x(N)$	$F_y(N)$	$F_z(N)$	$T_x(Nm)$	$T_y(Nm)$	$T_z(Nm)$
1	-0.064	-0.019	15.695	-0.011	-0.538	0.013
2	-0.118	0.005	15.689	0.011	-0.537	0.016
3	-0.097	0.006	15.652	-0.001	-0.523	0.015



(a) Raw Static Force Readings

(b) Filtered and Offset Static Force Readings

Figure 4.9: Raw Data and Filtered data obtained through the static tests. The data was filtered using a 6th order Butterworth lowpass digital filter with a cutoff frequency of 10 Hz. The figure shows the three forces and 3 torques in their raw format in (a) and in their filtered and offset format in (b).

4.6 Simulink Model Development

A mathematical simulation was designed to be a 3D multibody representation of the test rig. It was developed as a model to be compared to the experimental data so to aid in determining the desired parameters. Due to the complexity and size of matrices used in the model, this section will use the generalised matrix or formula unless the formula is simple enough to include numerical values and/or symbols. The simulation was developed using MATLAB's Simulink using a Euler-Lagrangian 3D Multibody Dynamics model. A 3D multibody dynamics model has two frames of reference that that can be worked with. The first is that of the body and the second is that of the world, or inertial, frame. Attention is crucial when doing rotations and translations to make sure that the correct frame is being used. It can be seen in Figure 4.10 that the body frame is situated some translation and rotation away from the world frame.

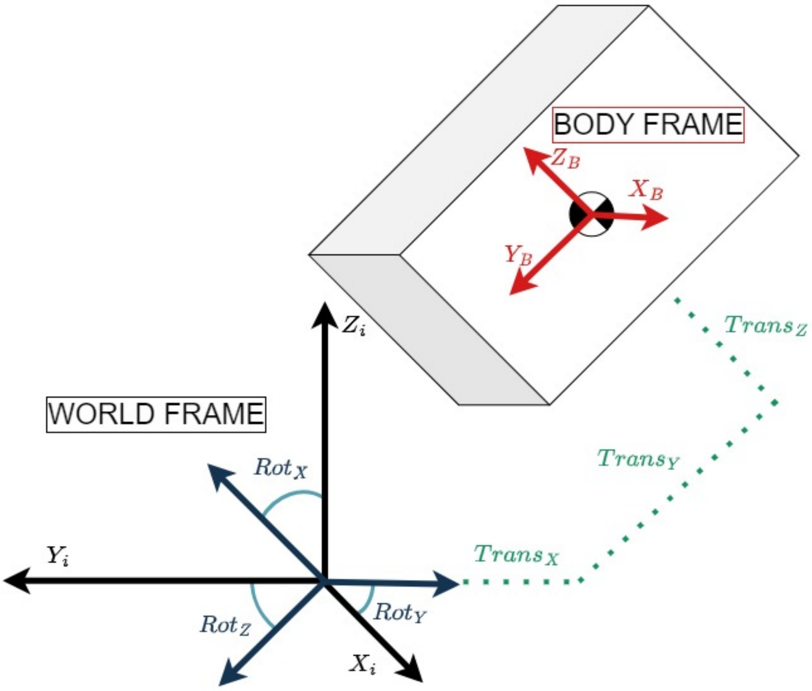


Figure 4.10: An example of the world frame and the body frame. The body frame is situated some translation and rotation away from the world frame.

To get from the world frame to the body frame, rotation matrices are used and while they can be defined in any order, for this model a Euler 3-2-1 rotation was used which is first a rotation about the z-axis, then the y-axis, then the x-axis. The matrices for the rotations can be seen in Equation 4.1.

$$\mathbf{Rot_X} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}, \mathbf{Rot_Y} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \mathbf{Rot_Z} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

Before the model development, the variables and multibodies needed to be defined. There are eight state variables that were used in this model. Six of them were defined and constrained to be stationary to be used to calculate the lambda forces experienced by the model. These lambda forces are the reaction forces that are the effective measurements made through the force sensor in the test rig. The seventh variable was the rotation about the point of rotation of the tail while the last variable was an error variable that was included as, after studying the built test rig, it was seen that the tail was slightly off kilter in the roll axis. This was therefore included as a variable to be able to be adjusted to get the best fit to the experimental data. These variables can be seen in Equation 4.2and can be seen in a simplified illustration of the model in Figure 4.11. Due to the nature of constraint forces, the first six variables were constrained to be zero.

$$\mathbf{q} = \begin{bmatrix} x, & y, & z, & \phi_1, & \theta_1, & \psi_1, & \phi_2, & \theta_2 \end{bmatrix} \tag{4.2}$$

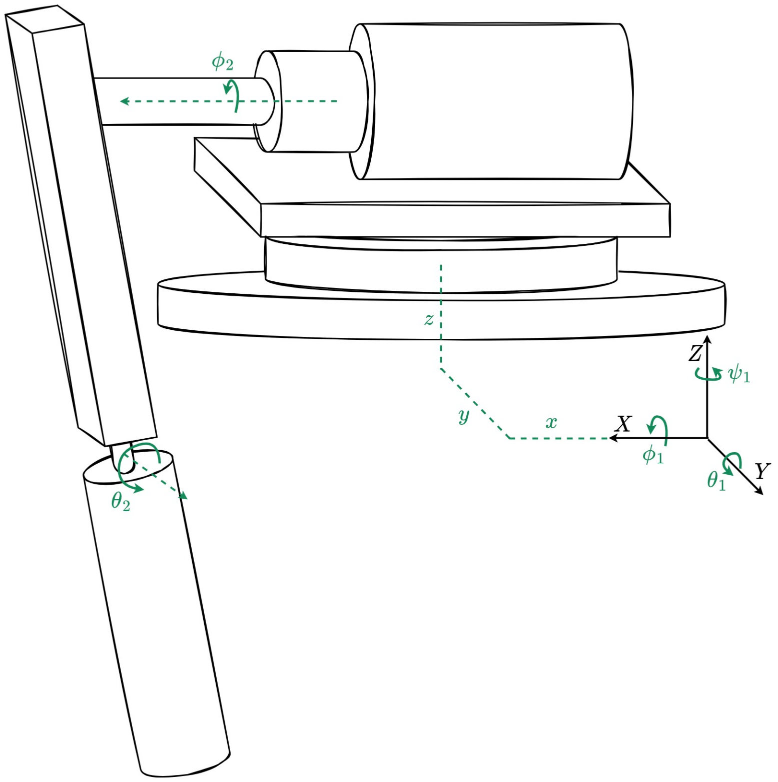


Figure 4.11: An illustration of the variables used in the dynamics model. The first six variables are constrained to be zero. The last two variables, ϕ_2 and θ_2, are the rotation of the tail and the error variable.

The model was split into three bodies: the stationary mounting structure which included the stator of motor, the steel bar and connecting shafts which included the motor rotor, and the cheetah tail. Due to three bodies being used with three different frames, three rotation sequences were required. These were from the inertial frame to the first mounting structure, Rot_{0-1}, from the mounting structure to the steel bar, Rot_{1-2}, and from the bar to the tail, Rot_{2-3}. While each rotation sequence was defined from one frame to the next, any rotation sequence can be defined from any frame to any other frame and this is done by multiplying the rotation sequences together. The rotation sequences can be seen in Equation 4.3 and the rotation combinations in Equation 4.4.

$$\mathbf{Rot_{0-1}} = \mathbf{Rot_X} \cdot \mathbf{Rot_Y} \cdot \mathbf{Rot_Z}, \quad \mathbf{Rot_{1-2}} = \mathbf{Rot_X} \cdot \mathbf{Rot_Y}, \quad \mathbf{Rot_{2-3}} = \mathbf{Rot_Z} \tag{4.3}$$

$$\mathbf{Rot_{0-2}} = \mathbf{Rot_{1-2}} \cdot \mathbf{Rot_{0-1}}, \quad \mathbf{Rot_{0-3}} = \mathbf{Rot_{2-3}} \cdot \mathbf{Rot_{0-2}} \tag{4.4}$$

To make sure the dynamics of the model were incorporated, a Euler-Lagrangian model was used which, for 3D dynamics, means that the manipulator equation needs to be used which can be seen in Equation 4.5. In this equation there are six terms. The M is the mass matrix, the C is the Coriolis matrix, the G is the gravitational matrix, the τ is the input torque matrix, the Q is the external force matrix, and the $A^T\lambda$ is the constraint matrix.

$$\mathbf{M(q)\ddot{q}} + \mathbf{C(q,\ddot{q})} + \mathbf{G(q)} = \tau + \mathbf{Q} + \mathbf{A^T}\lambda \tag{4.5}$$

Starting off with the mass matrix, the kinetic energies of each body were required where the total kinetic energy is made up of translational and rotational energy. The mass matrix requires the sum of all kinetic energies of the various bodies. Each body's kinetic energy is made of the form of Equation 4.6 which shows the general formula as well as the 3D vector format. The total kinetic energy in Equation 4.7 shows the sum of the translation and rotational energy where, for the translation energy, the velocity is replaced by the velocity of the bodies ($\dot{r}$) and where, for the rotational energy, the velocity is replaced by the rotational velocity (ω_B) and the mass is replaced by inertia (I_B). The masses and inertias used for these formulae, excluding the tails, were obtained from the SolidWorks 3D model of the test rig. The tail's metrics can be seen back in Table 4.2 while the parameters of the rest of the model can be seen in Table 4.5. The bar's inertia is made up of the sum of all the inertias that are dependent on the rotation of the motor. This includes the bar, the couplers, as well as the inertia of the motor itself. The apparent inertia of the motor with a gearbox is the inertia of the rotor multiplied by the square of the gear ratio as seen in Equation 4.8.

$$\mathbf{E} = \frac{1}{2}m\mathbf{v}^2 = \frac{1}{2}\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \cdot m \cdot \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \tag{4.6}$$

$$T = \sum_{i=1}^{n}(T_{T_i} + T_{R_i}) = \frac{1}{2}\dot{\mathbf{r}}^{\mathbf{T}} m \dot{\mathbf{r}} + \frac{\mathbf{1}}{\mathbf{2}}\omega_{\mathbf{B}}^{\mathbf{T}}\mathbf{I_B}\omega_{\mathbf{B}} \tag{4.7}$$

$$I_{motor} = N^2 I_{rotor} \tag{4.8}$$

Table 4.5: The position of each body with respect to their body's frame.

Body	**Mass (kg)**	**Inertia** ($g.mm^2$)
Mounting Structure	1.189	$\begin{bmatrix} 7848671.46 & 46.99 & 371.26 \\ 46.99 & 7898265.25 & -204161.94 \\ 371.26 & -204161.94 & 2396927.46 \end{bmatrix}$
Steel Bar to Motor	0.313	$\begin{bmatrix} 673108019198 & 673108000000 & 673108000000 \\ 673108000000 & 673108019206 & 673108000000 \\ 673108000000 & 673108000000 & 673108000217 \end{bmatrix}$

To get the velocity vectors for each body, the position vectors are first defined with respect to each body frame. The position of each body was chosen as the origin of the world frame offset by values obtained by the SolidWorks 3D model of the test rig. The general formula for the position, r_i, of each body is seen in Equation 4.9 where the transform of some rotation is multiplied by the translation of the body in that frame. Each body's respective position is given in the Table 4.6. For values that are constant, a numerical value is given and for values that are variables, the variable is given. Due to the only difference in tests between the tails being the tails themselves, only variables associated with the tail were given.

$$\mathbf{r}_i = Rot^T \cdot \begin{bmatrix} x_i & y_i & z_i \end{bmatrix} \tag{4.9}$$

Table 4.6: The position of each body with respect to their body's frame.

Body	Variable	Position (numerical values in mm)
Mounting Structure	r_m	$Rot_{0-1}^T \cdot \begin{bmatrix} x-5.69 & y-0.11 & z+31.49 \end{bmatrix}$
Steel Bar	r_s	$r_m + Rot_{0-1}^T \cdot \begin{bmatrix} 101.45 & 0 & 139.89 \end{bmatrix} + Rot_{0-2}^T \cdot \begin{bmatrix} 0 & 0 & -100 \end{bmatrix}$
Tail	r_t	$r_m + Rot_{0-1}^T \cdot \begin{bmatrix} 101.45 & 0 & 139.89 \end{bmatrix} + Rot_{0-2}^T \cdot \begin{bmatrix} 0 & 0 & -270 \end{bmatrix}$ $+ Rot_{0-3}^T \cdot \begin{bmatrix} 0 & 0 & -Tail_{CG} \end{bmatrix}$

The input torques was just the torque, T, applied to the motor in the direction of the state variable, $\vec{u}$. Equation 4.10 shows this relation for the torque applied by the Maxon multiplied by the direction applied, the state vector ϕ_2.

$$\tau = \sum_{i=1}^{n} (T_i \cdot \mathbf{u}) = T \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{4.10}$$

The external forces, forces applied to the body by the environment, were calculated by taking the partial differential of the position vector with respect to the state variables and having it dot multiply by the force vector and can be seen in 4.11.

$$\mathbf{Q_i} = \sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f_j} \cdot \frac{\partial \mathbf{r_j}}{\partial \mathbf{q_i}} \tag{4.11}$$

In the static model, seven external forces were used. The first was friction applied through the bearing and motor joints. This was applied along the ϕ_2 axis where the force applied was taken to be the combination of both coulomb and viscous friction. These friction models can be seen in [42]. Due to the nature of MATLAB not liking discontinuous systems in which the simulation compilation speed drastically decreased, the model was altered to produce a continuous system. This was done by using a tanh function, as seen in 4.12, to smooth out the discontinuity. While this assumption does not perfectly model the real system, it does allow for a faster simulation with little drawbacks. The issue about this friction model is what occurs at zero velocity where the function returns a zero friction where in reality there is the coulomb friction acting upon it at that time. This non-zero friction is a small issue that can be ignored for the purposes of this model, due to how the current was modelled for static swings and that this situation didn't arise for dynamic swings. The modelled friction can be seen in Figure 4.12. This figure shows coulomb friction, viscous friction, and the combined friction model. The coulomb friction is the constant friction that is applied to the system when it is not moving. The viscous friction is the friction that is applied to the system when it is moving.

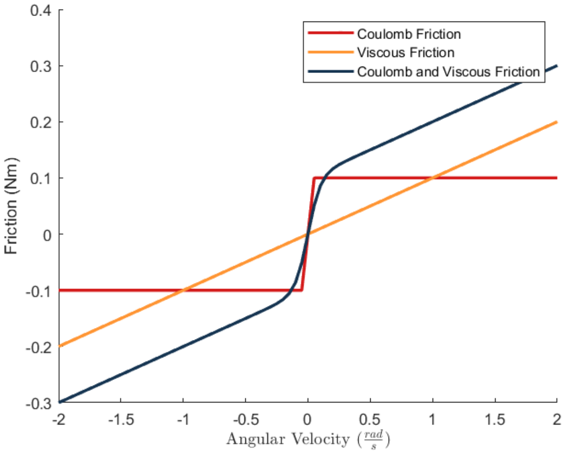
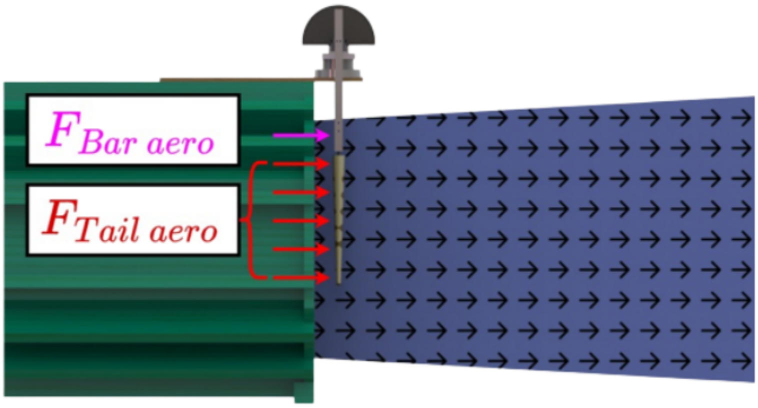


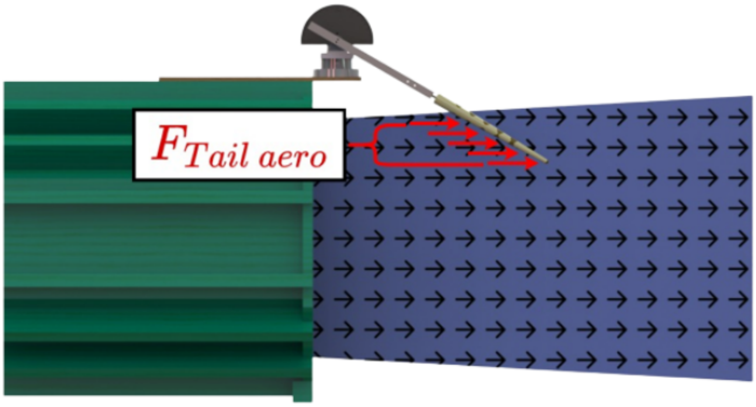
Figure 4.12: Three different friction models: coulomb friction, viscous friction, and a combined coulomb and viscous model. The combined friction model was used for the purpose of this project.

$$\mathbf{F_{fr}} = \mathbf{Rot^T_{0-1}} \cdot \left[C_f \tanh(10\dot{\phi}_2) + V_f \dot{\phi}_2; \quad 0; \quad 0 \right] \tag{4.12}$$

The second external force was the aerodynamic force applied on the section of the steel bar that was in the wind. When the bar was not intersecting the wind, it was assumed that the aerodynamic force was negligible and so no force was applied to it. The last five forces were the aerodynamic forces applied to the tail. To prevent oversimplification of this model, the part of the tail that was in the wind was split into five sections. For most angles of attack experienced, this was across the whole tail, but for the larger angles of attack, parts of the tail was not in the wind and so the section that was in the wind was split into five equal sections. This sectioning of the tail and forces can be seen in Figure 4.13. In the figure, it can be seen that the wind coming out of the wind tunnel expands its boundaries the further it travels. This unseen effect is due to the shape of the wind tunnel and the fact that it had an open outlet. The gradient of the boundary on either side of the tunnel was measured to be $\frac{1800}{100}$.



(a) Wind Tunnel Test Rig



(b) Wind Tunnel Test Rig

Figure 4.13: The figure shows the force placements for varying angles of attack. While the tail is in the wind, the tail's area is split into five sections that are in the wind. The force on the bar is made into one section while it is in the wind. The subfigure (a) shows the force sections for 0 degrees while subfigure (b) shows the force sections for 60 degrees.

Each external force needed a force and a position to be dot multiplied. The position for each force is similar to that of that for the center of mass for the tail except extended along the z-axis by some other variable. The position for each force, $\vec{r_i}$ can be seen in Equation 4.13. In this case r_{f_i} is the position at which the force is being applied which is calculated by taking the length of the section of the tail in the wind and splitting it into five equal sections. The position is then situated in the middle of those sections.

$$\begin{aligned}\mathbf{r_i} = \mathbf{Rot_{0-1}^T} \cdot \begin{bmatrix}101.45; & 0; & 139.89\end{bmatrix} + \mathbf{Rot_{0-2}^T} \cdot \begin{bmatrix}0; & 0; & -270\end{bmatrix} \\ + \mathbf{Rot_{0-3}^T} \cdot \begin{bmatrix}0; & 0; & -Tail_{GAP}\end{bmatrix} + \mathbf{Rot_{0-3}^T} \cdot \begin{bmatrix}0; & 0; & -r_{f_i}\end{bmatrix}\end{aligned} \tag{4.13}$$

The force then associated with each aerodynamic drag force is given by the standard aerodynamic drag equation as mention in Equation 2.5. For means of implementing it in MATLAB, the force was separated into a scalar force which was then multiplied by the unit vector of the relative wind velocity. For the static tests, the relative wind velocity was just the wind velocity. However, for the dynamic tests, the relative wind velocity was the wind velocity vector minus the velocity of the tail and is illustrated in Figure 4.14. The tail's velocity was the derivative of the position, $\vec{r_i}$, giving a vector,

$\vec{v}_{tail_i}$ as can be seen in Equation 4.14. The relative velocity of the air is the difference between the velocity of the wind and the velocity of the tail. This relationship expanded in Equation 4.15.

$$\mathbf{v_{tail_i}} = \frac{\partial \mathbf{r_i}}{\partial \mathbf{q}} \dot{\mathbf{q}} \tag{4.14}$$

$$\mathbf{v_{rel}} = \mathbf{v_{wind}} - \mathbf{v_{tail_i}} \tag{4.15}$$

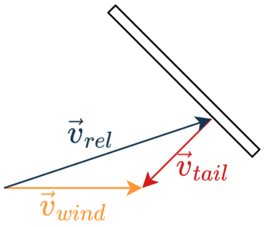


Figure 4.14: The figure shows the relative velocity of the tail. The relative velocity is the wind velocity vector minus the velocity of the tail vector.

The relative speed was then determined as the norm of the relative velocity, Equation 4.16, while the direction of the relative velocity was the unit vector, Equation 4.17.

$$v_{rel} = \sqrt{\mathbf{v_{rel}}^{\mathbf{T}} \cdot \mathbf{v_{rel}}} \tag{4.16}$$

$$\mathbf{v_{unit}} = \frac{\mathbf{v_i}}{v_{rel}} \tag{4.17}$$

This leads to the aerodynamic drag and lift force scalars being Equation 4.18 and Equation 4.19 respectively where ρ, C_D, C_L, A_{ref}, and v_{rel} are the density, drag and lift coefficients, the reference area and the relative velocity respectively. These forces were then multiplied by their unit vectors to give the final aerodynamic forces. For the lift forces, the unit vector was rotated by 90° about the x-axis to act vertically along the tail away from center of rotation. These vectors are seen in Equations 4.20 and 4.21.

$$F_{aero_D} = \frac{1}{2}\rho C_D \cdot A_{ref} v_{rel}^2 \tag{4.18}$$

$$F_{aero_L} = \frac{1}{2}\rho C_L \cdot A_{ref} v_{rel}^2 \tag{4.19}$$

$$\mathbf{F_{aero_D}} = F_{aero_D} * \mathbf{v_{unit}} \tag{4.20}$$

$$\mathbf{F_{aero_L}} = Rot_X(90^\circ) * F_{aero_L} * \mathbf{v_{unit}} \tag{4.21}$$

These drag and lift forces were then added to the external forces, Q, through the method used in Equation 4.11 and then added together to produce one force acting on each section. This method is done across each of the five sections of the tail, and the one section on the bar. The reference areas used in Equations 4.18 and 4.19 were obtained by taking the cross product of relative wind velocity and the tail's velocity. This angle is multiplied the area of the tail, the length by the relevant axis obtained from Table 4.2, to get the reference areas. The drag and lift coefficients for the tail were values that were changed to produce the best fit for the simulation to the real data. These coefficient changes are done through an optimisation process as described in Section 4.7. The drag coefficient used for the bar was 2, the standard drag coefficient of a rectangle.

Once the external forces had been calculated, the next thing to do was to generate the constraints. These constraints are parameters that the simulation has to hold true. In this case, the constraints were the initial, and constant, positions of the motor support. Through these constraints the simulation was able to generate the reaction forces which in turn will be compared to the real data recorded through the force sensor. The six reaction forces needed were the three forces and three torques from the force sensor, so the first six parameters in the state vector are constrained to be 0. The final constraint was the offset angle that the tail falls at. This gives a constraint vector as seen in Equation 4.22

$$\mathbf{H} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \phi_1 \\ \theta_1 \\ \psi_1 \\ \theta_2 \end{bmatrix} = 0 \tag{4.22}$$

Using these constraints, the constraint Jacobian, which is used in the manipulator equations, could be generated. This was done through Equation 4.23. The velocity and acceleration constraints Were calculated through the use of the constraint Jacobian. The acceleration constraint, Equation 4.24, could then be rearranged and plugged into the manipulator equation to solve for the acceleration of the system, $\ddot{q}$ and can be seen in Equation 4.25. Once the state acceleration was solved for, the reaction forces, Equation 4.26, could be calculated. This created a closed loop model that can be used to compare the real data to the simulation's data.

$$\mathbf{A} = \frac{\partial \mathbf{H}}{\partial \dot{\mathbf{q}}} \tag{4.23}$$

$$\mathbf{A}\ddot{\mathbf{q}} + \dot{\mathbf{A}}\dot{\mathbf{q}} = \mathbf{0} \tag{4.24}$$

$$\ddot{\mathbf{q}} = \mathbf{M^{-1}}(-\mathbf{C} - \mathbf{G} - \mathbf{Q} + \mathbf{B}\tau + \mathbf{A^T}\lambda) \tag{4.25}$$

$$\lambda = -(\mathbf{AM^{-1}A^T})^{-1}(\mathbf{AM^{-1}}(\mathbf{B}\tau - \mathbf{C} - \mathbf{G} - \mathbf{Q}) + \dot{\mathbf{A}}\dot{\mathbf{q}}) \tag{4.26}$$

This model was designed in MATLAB and once the matrices and functions were developed, they were imported into Simulink using the *matlabFunctionBlock* function which generates a MATLAB function block in the specified Simulink model.

4.7 Optimisation for Variable Collection

The optimisation of multiple variables was needed to determine the various model constants. These variables included the friction coefficients, as well as the drag coefficients and the added mass constant in the dynamic swings. The optimisation function used in this analysis was that of the *lsqnonlin* function found in MATLAB which solves nonlinear least-squares curve fitting problems in the form of Equation 4.27. The function can also take optional lower and upper bounds, *lb* and *ub*, as well as an initial guess or seed, x_0 [43].

$$\underset{x}{min}||f(x)||_2^2 = \underset{x}{min}\left(\sum_{i=1}^{n} f_i(x)^2\right) \tag{4.27}$$

Each of the $f_i(x)$ functions in the lsqnonlin function is a user defined function to minimise for, otherwise known as the cost function. A general cost function is given by the equation in Equation 4.28. In this case, σ_1 and σ_2 are the experimental and simulated data respectively, and ϵ is the uncertainty factor associated with the sensing of the experimental data.

$$f_i(x) = \frac{\sigma_1 - \sigma_2}{\epsilon} \tag{4.28}$$

The first set of variables to be optimised were the friction coefficients. To determine the friction coefficients, the position profile of an actuated swing was required. This profile would be compared to a simulated model, with the cost function being run chosen to minimise the position difference between the real and simulated data. A simple experiment was set up which took the wind tunnel rig, removed the rig from the wind tunnel, removed the tail and placed on a flat surface. The rig, which included the force sensor, motor and encoder, and the bar, was then held in a position such that the bar was held at a horizontal 90°. While the position was being logged, the bar was released from its horizontal position and allowed to swing down, and during this motion the data was recorded to compare to the simulation data. The only difference between the model used in this process and the model described in Section 4.6 was that the tail and aerodynamic forces were removed. This allowed for a localising of the friction coefficients. With this localisation, an optimisation could be run to determine the coefficients. This was described in Equation 4.28 where the uncertainty factor was 1.18e-4 as shown in Table 4.8. With the optimisation run, the profiles were obtained. To see an example of the two position profiles, see Figure 4.15. After five unactuated swings were analysed, the results were seen to be consistent with the results of the optimisation, as well as the seeds, can be seen in Table 4.7. The individual results were averaged with a weighted average model according to the inverse of the resnorm value. This method is described in Equations 4.29 - 4.31. These values were used in the model as the friction coefficients. It was seen that the C_f and V_f were 0.0435 and 0.0364 respectively.

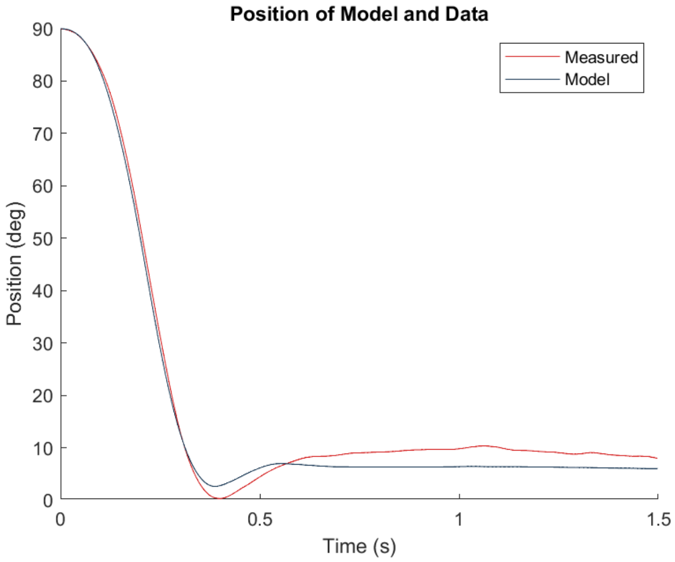


Figure 4.15: Position graph of simulated against experiment of an unactuated 90° swing of the bar to determine the coulomb and viscous friction coefficients.

Table 4.7: Results of the lsqnonlin function with their initial conditions. The coulomb and viscous coefficient as tabulated as C_f and V_f respectively. The resnorm is included as well as the weighted average of each coefficient.

Variable	**Seed**	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5	**Weighted Average**
C_f	0.0344	0.0272	0.0257	0.0389	0.0814	0.0364	**0.0435**
V_f	0.0381	0.0422	0.0446	0.0370	0.0230	0.0383	**0.0364**
Resnorm	*N/A*	0.0283	0.0442	0.0329	0.0309	0.0667	*N/A*

The method of finding the weighted average of the inverse of the norm is as follows. The sum of the resnorms needed to be divided per test condition by each individual resnorm to give a temporary variable, ϵ. Following that, each of these results where then normalised to make them all sum 1 by dividing each of them by the sum of the results. These weighting values, w, were then multiplied by the corresponding coefficient value to get the final coefficient value for each test condition. The sequence of equations to get the weighted average can be seen in Equations 4.29 - 4.31.

$$\epsilon_i = \frac{\sum_{k=1}^{5} res_{i_k}}{res_i} \tag{4.29}$$

$$w_i = \frac{\epsilon_i}{\sum_{k=1}^{5} \epsilon_{i_k}} \tag{4.30}$$

$$var = \sum_{k=1}^{5} w_{i_k} * var_k \tag{4.31}$$

With the friction coefficients calculated, the optimisation function can solve for the drag coefficients. The model used had seven variables with its cost function associated with it as it sought to adjust the data to get to the error terms of the six forces and torques and the speed minimised. The variables used in cost functions as well as their uncertainty factors for the lsqnonlin can be seen in Table 4.8. The uncertainty factors for each cost function come from the uncertainty in the experimental data due to the sensor used. For the force sensor, this uncertainty term for each axis is 2% of the readings based on the accuracy of the Axia80 added to the error values observed through the calibration process of the sensor. These guidelines are given by the manufacturer of the sensor. The uncertainty factor for the encoder comes from the encoder resolution of 512 Pulses Per Revolution (PPR) and the gear ratio of 26:1.

Table 4.8: Cost function variables used in the lsqnonlin function.

Cost Function	**Variable**	**Uncertainty Factor(ϵ)**
f_1	F_x	8.0067
f_2	F_y	7.9933
f_3	F_z	14.4600
f_4	T_x	0.4283
f_5	T_y	0.4450
f_6	T_z	0.3800
f_7	ω_{tail}	1.18e-4

The variables still needing to be solve for was the drag and lift coefficients, as well as the rotation of the tail due to tail structural errors given as ψ_3. This can be seen as a rotation in the direction of the wind by the same angle while keeping the tail straight. This was seen to be required due to the fact that a force in the x-axis was measured. The topic will be further expanded up in the section 4.8. The only input to the model, apart from the result analysis, was the current to set the input torque. This current was measured through the Maxon motor controller. Using this current however, it was seen that the model was not accurate enough, and the simulated tail position did not align with the measured tail position. This error could have been due to two facts with the first being that the friction model was not accurate at near-zero speeds. The friction model, as shown previously in Figure 4.12, took only into account the coulomb and the viscous friction. A realistic, discontinuous, model of friction includes the breakaway friction and the Stribeck friction which results in a friction graph looking more like Figure 4.16 [44].

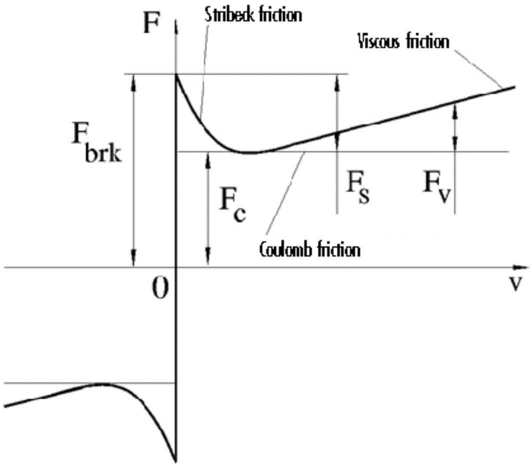


Figure 4.16: Full friction model between moving bodies. The model includes the coulomb, viscous, breakaway (F_{brk}), and Stribeck (F_{str}) friction. Image taken from [44].

With static friction being difficult to model, it is understandable for model error to creep in. To adjust for this, only during static tests, the input current was included in the variables as a constant for the lsqnonlin to solve. This method removed the difficulty of modelling static friction and allowed for the model to choose the optimal current to keep the tail stationary. The current was then used to calculate the input torque by multiplying it by the torque constant to ensure the model's position and speed accuracy. Due to the fact that the forces were the main part of the cost function, this was deemed acceptable as it would allow for the drag coefficients to be calculated while not worrying about external errors.

The optimisation required an initial guess for each variable to be solved for. These initial seeds were chosen based on what the researcher assumed to be near what the results would end up being. While lsqnonlin can solve for the variables without an initial guess, namely by having a zero guess which could have resulted in a local minimum while the optimal solution does not lie at that specific point. The lsqnonlin optimisation could also have used random seeds to get the global minimum, rather than potentially finding the solution in a deep local minimum. This method would have worked, but because of time and computing constraints, as well as a good general knowledge of the expected values, the seeds were chosen to be fair values according to the researchers estimation. Multiple initial seeds were tested but final initial seeds were found after testing the results. These seeds, for the 0°static test, can be seen in Table 4.9 while the bounds can be seen in Table 4.10 and these rest of the seeds chosen for the static tests can be found in Appendix 10.1.

Table 4.9: Initial conditions used in the lsqnonlin function for the three tail at a 0° static position for varying winds speed of 10, 20, and 30 m/s.

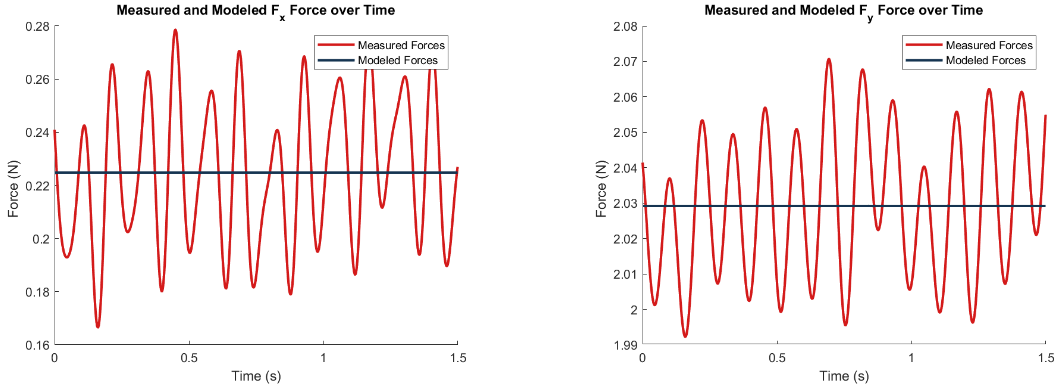
Variable	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
	10 $\frac{m}{s}$	20 $\frac{m}{s}$	30 $\frac{m}{s}$	10 $\frac{m}{s}$	20 $\frac{m}{s}$	30 $\frac{m}{s}$	10 $\frac{m}{s}$	20 $\frac{m}{s}$	30 $\frac{m}{s}$
C_D	1.3	1.1	1.1	1.6	1.4	1	1.7	1.4	1
C_L	0	0.03	0.09	0.04	0.04	0.1	0	0	0.1
ψ_3	0.01	0.08	0.045	0	0.08	0.07	0.01	0.08	0.07
I_{in}	2.75	9	18.5	2.4	9	15.6	2.75	9	15.6

Table 4.10: Upper and lower bounds used in the lsqnonlin function for the three tail at a 0° static position for varying winds speed of 10, 20, and 30 m/s.

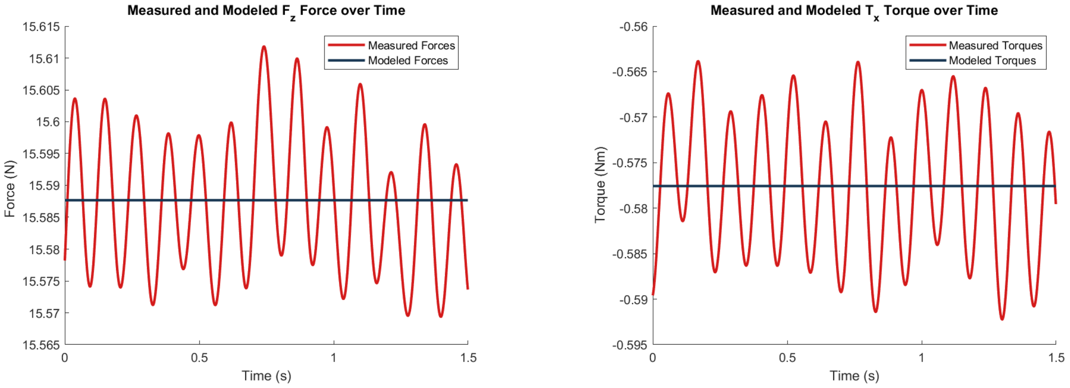
Variable	**Condition**	**Limit**
C_D	lb	0
	ub	3
C_L	lb	0
	ub	3
ψ_3	lb	-0.12
	ub	0.12
I_{in}	lb	0
	ub	30

4.8 Results of the Static Testing

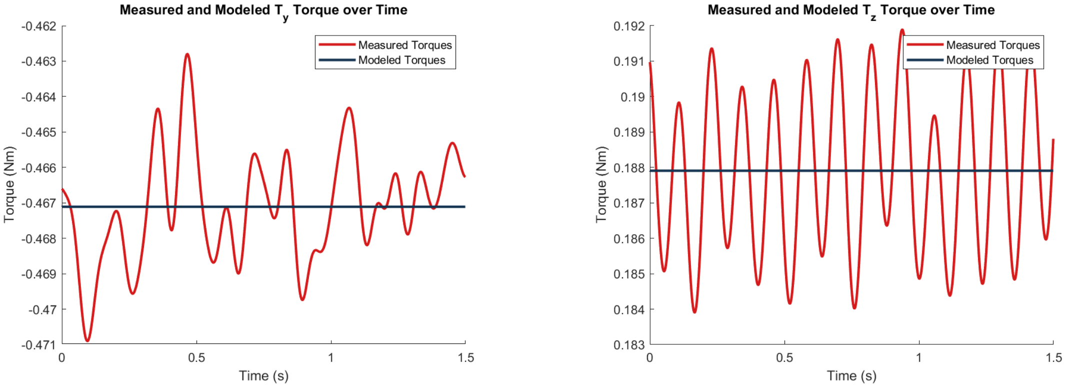
Due to the lack of motion in the static tests, the forces and moments measured were static as well which meant that the forces measured and simulated could be seen to be constant. While a real system could never produce a constant force as there will would always be noise depending on the sensitivity and accuracy of the sensor, the simulated data could be best fit to the average of the measured data. After running the optimisation through Simulink, the resulting forces can be plotted against the measured forces to test the accuracy of the process. While the resnorm, the squared norm of the residual, value from the optimisation result gave a numerical value for the accuracy of the model, the plotted forces could be seen to be a good illustrative representation of the results. The forces and torques measured and simulated can be seen in Figure 4.17. These graphs are from the static forces measured in one test at a 0° angle of attack. For the sake of brevity, these static force plots are the only static force plots in this paper. The other 539 tests run look similar, except with varying values. These results, however, show that the model is able to accurately predict the forces and torques produced by the system. In these plots, the position and speed plots are not shown as they too are just static values. What was noted was that, albeit small, a force was measured in the direction perpendicular to the wind direction in the x-axis. This could be due to abnormal wind flow, but it is assumed that the any drag forces associated in the perpendicular axis would be cancelled out on both sides due to the symmetry of the tails. For this reason it was assumed that the tail wasn't put accurately facing the wind direction but was angled by some angle chosen to be ϕ_3 as mentioned in the previous section. This angle could be seen as the tail angle offset, a rig angle offset, a wind direction offset, or even all three combined. In the model, the angle was chosen and optimised for to allow for any angle uncertainties to be removed by making the forces in the x-axis match the measured forces.



(a) Static F_x Force **(b)** Static F_y Force



(c) Static F_z Force **(d)** Static T_x Force



(e) Static T_y Force **(f)** Static T_z Force

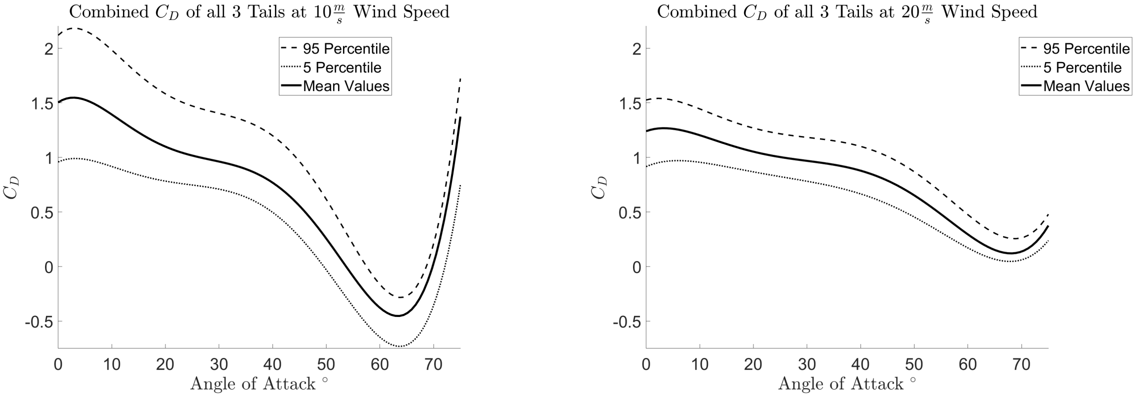
Figure 4.17: Static Forces at 0° Angle of Attack comparing the results of the modelled and measured data. The figures are for F_x, F_y, F_z, T_x, T_y and T_z respectively.

The resulting values obtained through the optimisation were tabulated and shown in Table 4.11. The table shows the wind speed the test was done at, the test number of each test per wind speed, the resulting drag and lift coefficients, the angle offset of the wind, the chosen current, and the resnorm. These results were the output of the optimisation function per test done on the various tail orientations. Any resnorm that were definite outliers are highlighted in red. To save space, only one table is shown here and the rest of the results for the static tests where put in Appendix 10.2.

Table 4.11: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's major axis.

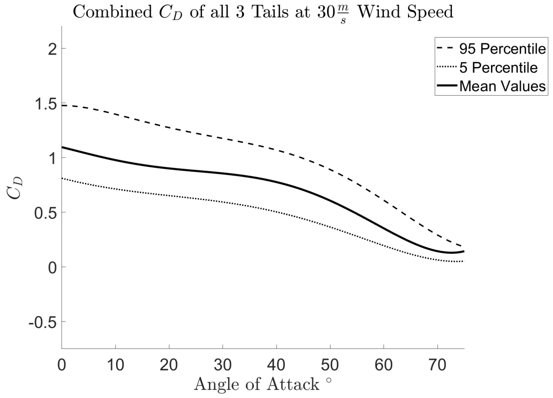
Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.316	0.062	-0.068	2.685	0.232
	2	1.316	0.062	-0.068	2.685	0.187
	3	1.316	0.062	-0.068	2.685	0.449
	4	1.316	0.062	-0.068	2.685	0.248
	5	1.316	0.062	-0.066	2.685	0.596
20 m/s	1	1.116	0.080	-0.056	9.143	23.005
	2	1.097	0.064	-0.057	8.989	9.809
	3	1.072	0.072	-0.056	8.787	62.734
	4	1.076	0.045	-0.058	8.820	7.761
	5	1.069	0.056	-0.054	8.772	24.604
30 m/s	1	1.011	0.063	-0.045	18.714	69.824
	2	1.006	0.082	-0.046	18.606	66.120
	3	0.986	0.061	-0.112	17.872	6280.690
	4	0.997	-0.050	-0.023	18.503	4698.952
	5	0.983	0.074	-0.112	17.817	6129.945

The results where then combined according to their resnorm weighting, as explained in Equation 4.31, to get the final results for the static test. Using this weighting method, which weights each result according to the lowest resnorm values, the final results for the coefficients of drag and lift for each total where then calculated, graphed and plotted. The results for the coefficients were combined from all three tails with both axes, with the angle of attack being on the x-axis and the coefficient being on the y-axis. The results for the coefficients of drag, as well as the 95th and 5th percentile for illustrative purposes, are shown in Figure 4.18 while results and percentiles for the coefficients of lift are shown in Figure 4.19.



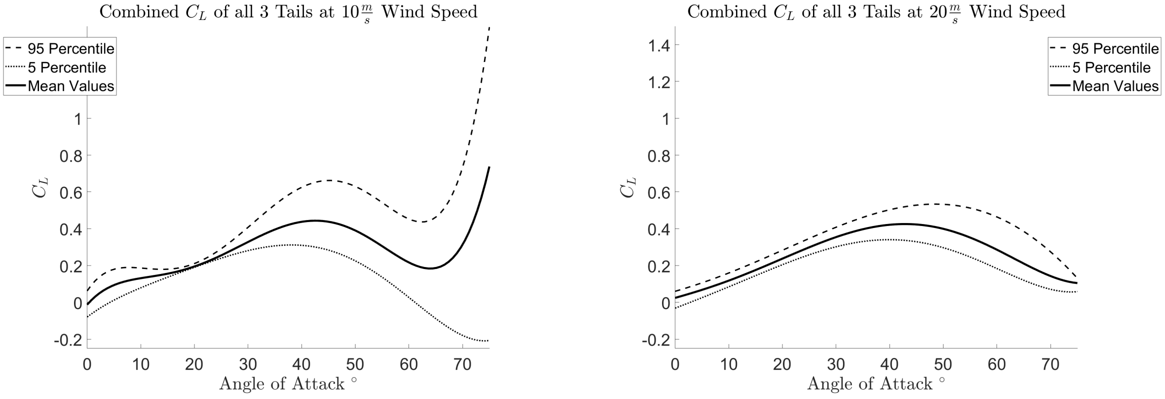
(a) 10 m/s Wind speed

(b) 20 m/s Wind speed



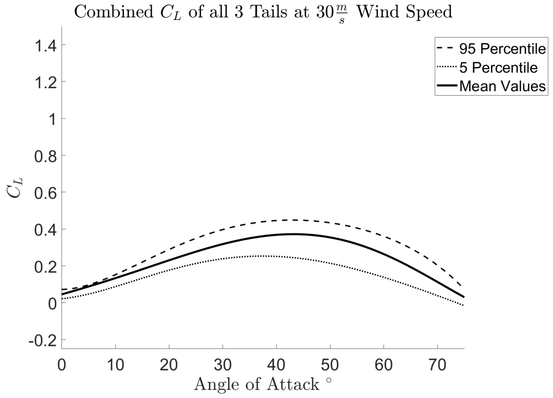
(c) 30 m/s Wind speed

Figure 4.18: The mean drag coefficients of all three tails for varying wind speeds across various angles of attack with the 95th and 5th percentiles are also shown. The results are shown for the 10 m/s, 20 m/s and 30 m/s wind speeds in their respective subfigures.



(a) 10 m/s Wind speed

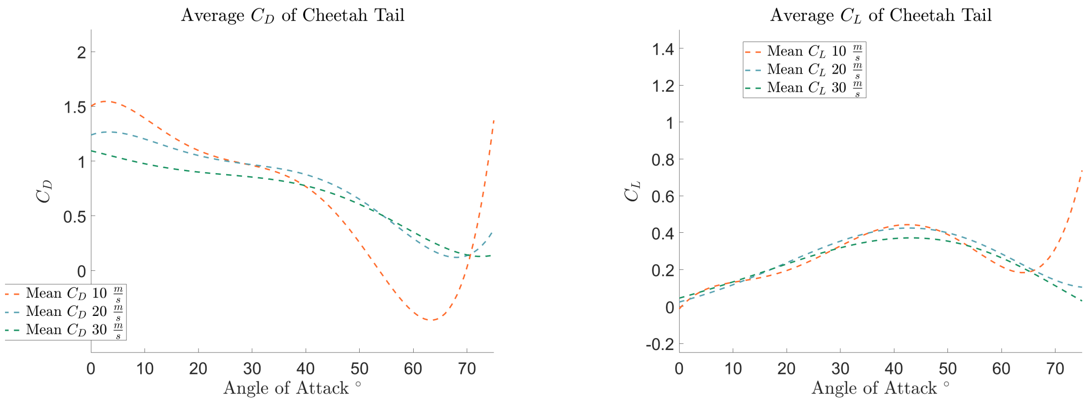
(b) 20 m/s Wind speed



(c) 30 m/s Wind speed

Figure 4.19: The mean lift coefficients of all three tails for varying wind speeds across various angles of attack with the 95th and 5th percentiles are also shown. The results are shown for the 10 m/s, 20 m/s and 30 m/s wind speeds in their respective subfigures.

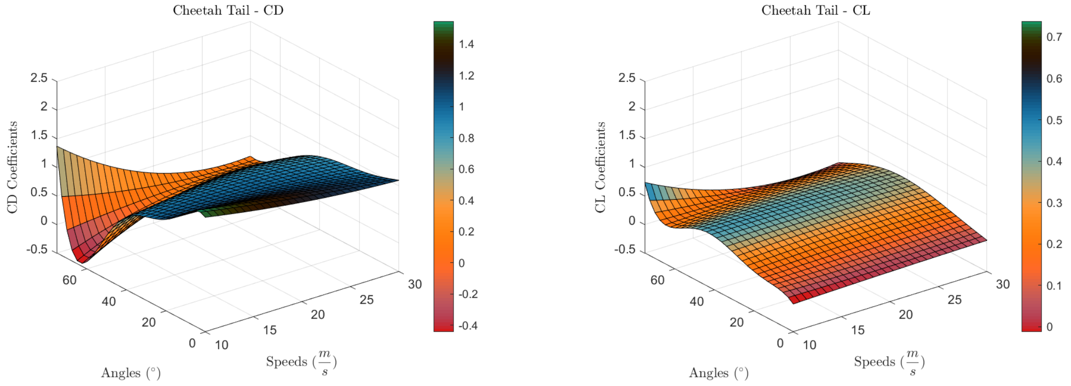
These graphs are then combined to show them all on a single plot, which can be seen in Figure 4.20. This graph shows the average coefficient values for each angle of attack for each of the three wind speeds.



(a) Average Coefficients CD **(b)** Average Coefficients CL

Figure 4.20: The various wind speeds tested combined onto a single figure showing average coefficients of drag and lift along each angle of attack for each wind speed.

Finally, all of these results, being nodes, were interpolated and then plotted on a surface plot to show the full effect of the wind speed on the coefficients of drag and lift. The values between the measured values were interpolated using a MATLAB's polyfit function with a 2nd degree polynomial for the speed axis and a 5th degree polynomial for the angle axis. These plots can be seen in Figure 4.21



(a) Surface Plot of the average Coefficient of Drag **(b)** Surface Plot of the average Coefficient of Lift

Figure 4.21: Surface Plots of the Coefficient of Drag and Lift of the cheetah tails tested in the wind tunnel at different wind speeds and angles of attack.

4.9 Results of Dynamic Testing

The previous section looked at modelling the aerodynamics of the cheetah tail using a static model. This quasi-steady-state model creates a dynamic model based on static points and interpolates between the points. This section looks to confirm whether that model holds true for dynamic motion and whether an external force can be added to produce a more accurate model. The external force looked at is a force produce by an apparent, or added, mass.

The means to determine the accuracy of the model relied on the experimental data of actuated swings of the cheetah tail in the wind tunnel and the simulated model of the system. The method of experimentation, as described in Section 4.4, involved the swinging of the cheetah tail, in their various orientations, in a pitch motion into the wind. These swings were recorded, and the results were then compared to the simulated model of the system. There were two differences between the static and dynamic analysis versions of the model. The first difference was the current input into the system. Due to the nature of static friction, the static tests involved the optimisation of the current input into the system and allowed the system to choose the optimal current input. With the swing tests, this was not an accurate means through which to solve the problem. While it was sound to allow the current to be chosen to allow the position to remain constant while optimising the drag coefficients, the motion of the swing tests removed this option. The current input that was used then, was the current measured from the motor during the swing. Using this method allowed for one value to be optimised when determining an accurate model. The value that could be optimised was the added mass constant. This was the second difference between the static and dynamic versions of the model: an added mass force was included. This force was modelled by having the acceleration of the tail ($a_1(t)$ multiplied by an added mass constant (m_{added}). This can be seen in Equation 4.32.

$$F_{added_1} = m_{added} * \mathbf{a_1}(t) \tag{4.32}$$

In this equation, the acceleration was realised in the same direction as the relative velocity of the tail, as previously seen in Equation 4.17. This force was transformed into the correct frame and added to the total force of the system. The added mass force was, like the drag forces, split into five components split equally along the length of the tail that was modelled to be in the wind according to the gradient of the boundary of the air flow coming out of the wind tunnel.

The drag and lift forces also needed to be adjusted to account for the quasi-steady-state nature of the model. The drag coefficients of the static model of the cheetah tail were found to be dependent on the relative wind velocity and the angle of attack of the tail. Using the static found drag and lift coefficients, these coefficients were interpolated using a linear 2D lookup table in the Simulink model. Due to the nature of the tail's pitch motion, the five tail forces each experienced different drag coefficients. While the angle of attack was the same for each point, the relative velocity was different and so each section of the tail were modelled to experience their own aerodynamic coefficients. Even though the added mass force was different across each section due to the varying accelerations at the varying points, a single added mass constant was used to determine the validity of the model. A brief figure showing these varied coefficients can be seen below in Figure 4.22.

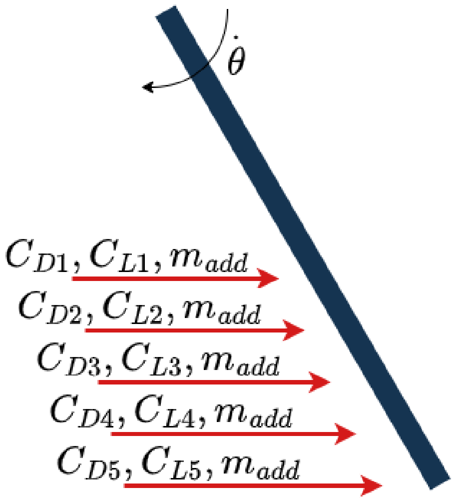


Figure 4.22: The figure showing the varying coefficients of drag and lift and the constant added mass constant across the force sections of the tail.

With the changes to the model to convert it from a static model to a dynamic model, it was time to analyse the model and determine the validity of a quasi-steady-state model with the addition of an added mass force which was done in a two-step process. The first step was to set the added mass constant to zero and see how close the quasi-steady-state model was to the experimental data and the second step was to optimise the added mass constant to see if the model could be improved. The optimisation of the added mass constant was done using MATLAB's lsqnonlin function and the cost function was chosen to minimise the error in position. The forces were not included in the cost function so to see how the added mass effect would affect the position of the tail. The forces were treated as a resultant of the model after running the optimisation function as the position profile was affected the most. The optimisation was run for each tail and each axis of the tail. While the actuated swing tests were run for wind speeds of 10, 20, and 30 m/s, the added mass constants were only optimised for the 10, and 20 m/s wind speeds. The reason for this will be explained further as the results are analysed, but it was found that the 30 m/s wind speed was too high for the added mass constant to be optimised. The optimisation seeds and bounds can be seen below in Table 4.12 and the seeds, or initial conditions, were chosen to allow for convergence of the optimisation function. Various seeds were tried which didn't let the function converge, but after finding seeds that allowed for convergence, the bounds were then set to allow for the optimisation to find the best value for the added mass constant. The bounds were chosen to allow for a large range of values to be tested. The reason for negative lower bound was to allow lsqnonlin to choose a negative constant, which would show that the addition of the added mass effect was rather detrimental to the model.

Table 4.12: The seed values used for the lsqnonlin optimisation of the added mass constant.

Variable	**Conditions**	**Wind Speeds**	
		10 $\frac{m}{s}$	20 $\frac{m}{s}$
M_{added}	x_0	-0.1	0.1
	lb	-0.5	
	ub	0.5	

The optimisation was done across all three tails and both axes of the tail. For each variance, there were five swings that were done and combining these results resulted in a set of 60 constants. The optimisation function also returned the residual value, or the resnorm, that was the sum of the values in the cost function. Using the results calculated, the benefits of adding an added mass force to the quasi-steady-state aerodynamic model can be discussed. The position profiles of the swings can be seen for the 10, and 20 m/s tests in Figure 4.23 and Figure 4.24 respectively.

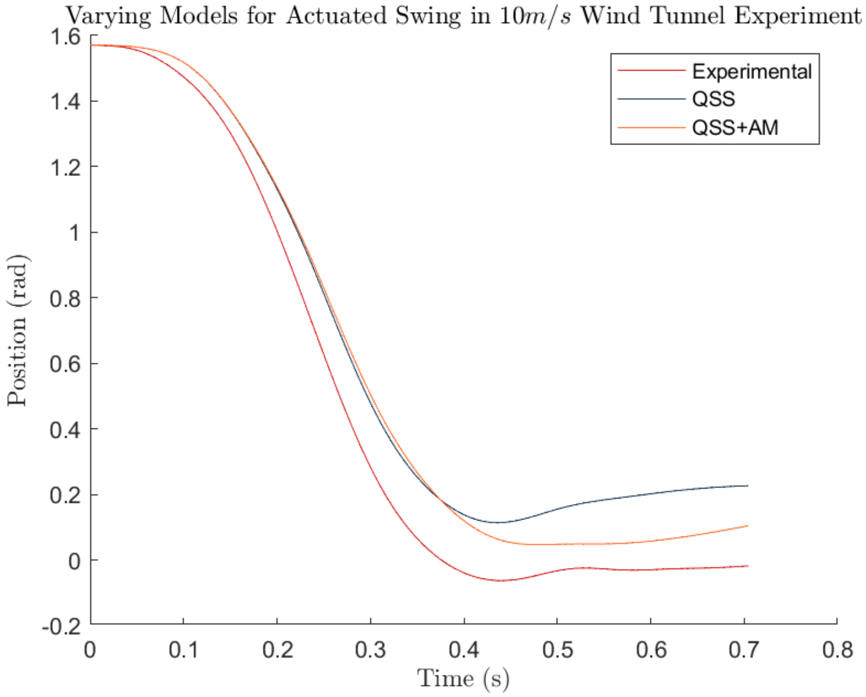


Figure 4.23: Position profiles of the experimental data, the QSS and the QSS with added mass for 10 m/s wind speeds for tail 1 in the major axis orientations.

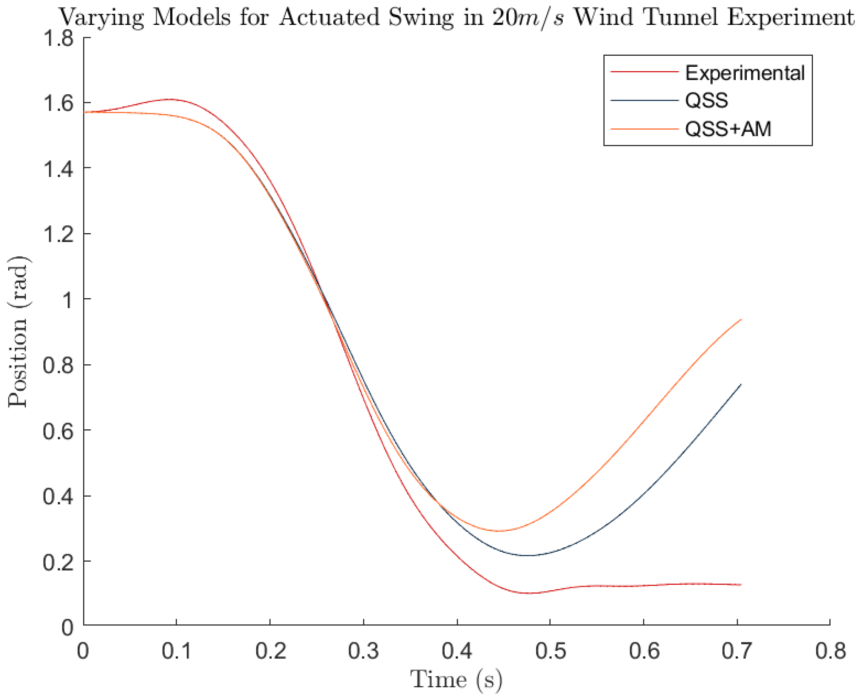


Figure 4.24: Position profiles of the experimental data, the QSS and the QSS with added mass for 20 m/s wind speeds for tail 1 in the major axis orientations.

These graphs show three different profiles: the experimental swing profile, the quasi-steady-state (QSS) model profile and the quasi-steady-state with added mass (QSS-AM) model profile. Even without looking at the quantitative results, it can be seen through the profiles that the models hold up better at lower wind speeds. It can also be seen that the QSS-AM model was the better model for wind speeds of 10 m/s while the QSS model was the better model for wind speeds of 20 m/s. In each of these profiles, it can be seen that the drag coefficients used need to be less than what was calculated. Based on the profiles, the higher the wind speed, the further away the calculated drag coefficients were from the coefficients needed.

In Figure 4.25, the weighted average added mass constants for the two wind speeds were plotted. It can be seen that, on average, the lower wind speed required a higher added mass constant than the higher wind speed tests. This, along with the fact that the 20 m/s tests required a negative constant, shows that the added mass force is more beneficial at lower relative wind speeds than higher relative wind speeds. The full set of results for the 10, and 20 m/s tests can be seen in Appendix 11.1.

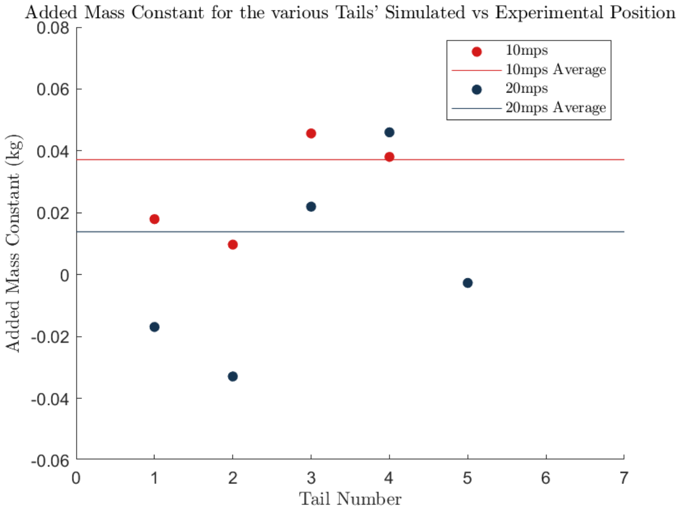


Figure 4.25: Weighted average added mass constants for 10, and 20 m/s wind speeds for the six tail orientations. The average added mass constant across the six tail orientations was also included.

To numerically see this, Figure 4.26 shows the resnorm of the QSS-AM model which shows that the 10 m/s model is more accurate than the 20 m/s model where the values of the model with the higher resnorm is a less accurate model.

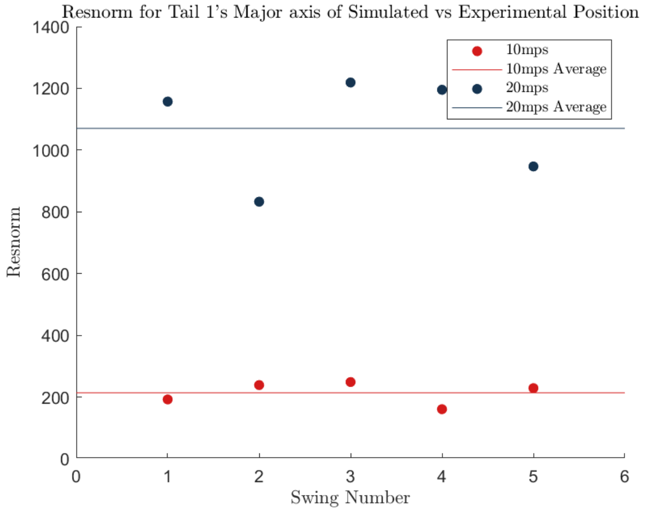


Figure 4.26: Resnorm for 10, and 20 m/s wind speeds for tail 1 in the major axis orientations. The average resnorm across the six tail orientations was also included.

With this trajectory, the 30 m/s tests were not optimised as at higher speeds the model loses validity. This can be explained by the relative wind speeds operating in various Reynolds numbers. Different shapes have different laminar to turbulent transition periods, and while the cheetah tail's Reynolds Number transition graph has not been explored, it is understandable that as the tail experiences a faster relative wind speed, the drag coefficient's change as the flow becomes more turbulent. This can be seen in Figure 4.27.

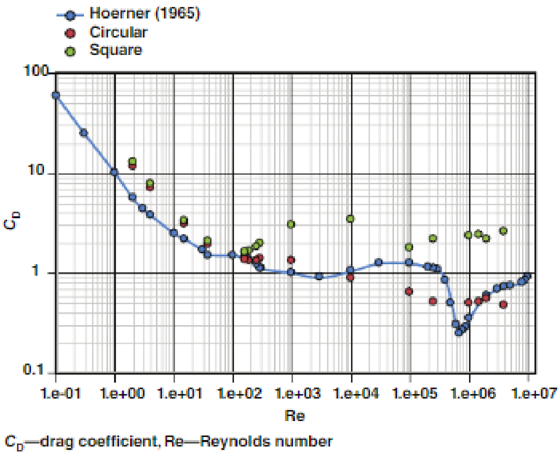
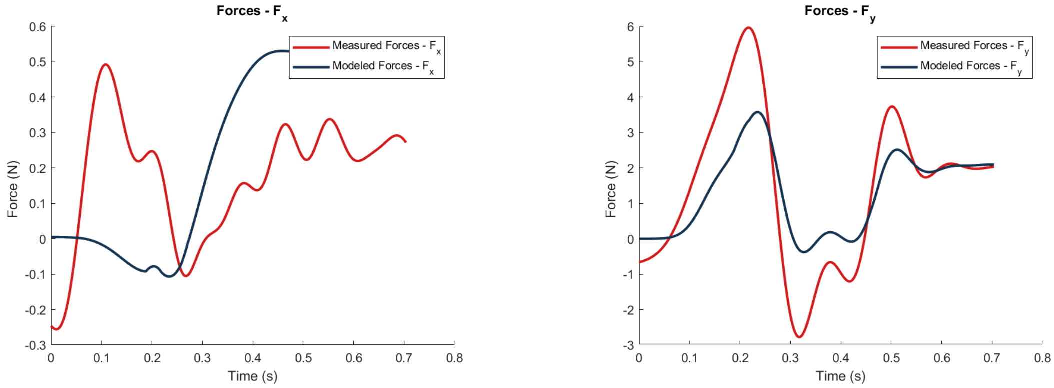


Figure 4.27: Figure showing the drag coefficients of various shapes at various Reynolds numbers. Image take from [45].

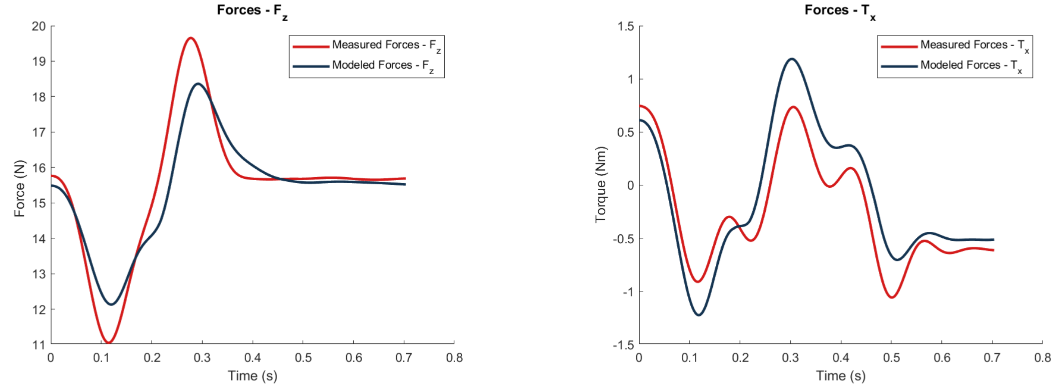
Through the varying of the swings in the wind tunnel, the relative wind velocity changed. At 10 m/s, the maximum speed that the tail experienced was 7.7047 rad/s. At 20 m/s, the maximum speed that the tail experienced was 7.6067 rad/s. At 30 m/s, the maximum speed that the tail experienced was 12.7987 rad/s. Using these radial speeds, the maximum speed at a point could be determined. Basing these values of the testing of tail 1, the radial distance between the center of rotation and the end point of the tail can be seen to be 0.66 mm. Using these two variables, the maximum point speed can be determined by multiplying the two. While the points at which the force was applied to the tail were not at the end of the tail, these calculations show the limits that the tail could experience. The maximum speeds experienced by the tail at the various wind speeds were 5.09 m/s, 5.02 m/s, and 8.45 m/s for the 10, 20, and 30 m/s wind speeds respectively. With these speeds, the air density with of 1.225 $\frac{kg}{m^3}$, the dynamic viscosity of air being 1.849e-5 $\frac{kg}{ms}$, and the tail diameter being 0.0639 m. The Reynolds Number for each of these speeds can be seen to be $Re_{10} = 6.391e4$, $Re_{20} = 1.060e5$, and $Re_{30} = 1.628e5$ for the 10, 20, and 30 m/s wind speeds respectively. These values were calculated from the formula for Reynolds Number as seen below in Equation 4.33 where ρ is the air density, v_{rel} is the relative velocity of the tail, D is the diameter of the tail and μ is the dynamic viscosity of air.

$$Re = \frac{\rho v_{rel} D}{\mu} \tag{4.33}$$

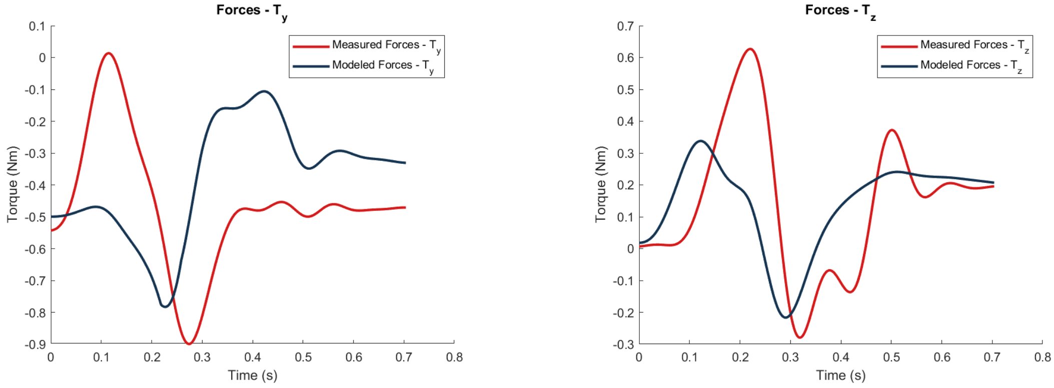
For these ranges of Reynolds Number, it makes sense, comparing it to regular shapes, that this range is a transition range where the drag coefficients change. Due to this, the QSS model is not a viable model to use for future analysis at higher relative wind speeds. However, due to the fact that the QSS and QSS-AM models were both more accurate the lower the Reynolds Number, both these models were seen as viable models to use for future analysis at lower wind speeds. While the QSS-AM model was more accurate for some tests, it was seen that there seemed to be some external factor that was not accounted for: likely that the added mass constant was not uniform across the tail, or that there was a time dependent variable unaccounted for which has been seen in numerous studies, such as by Grift [13]. Because this effect was not fully accounted for and seemed to show more sporadic accuracy across the tests done, the QSS model was chosen as a suitable standard model to use for future analysis at low relative wind speeds. A collection of the results for the QSS model's reactive forces compared to the measured reactive forces can be seen in the Figure 4.28.



(a) Dynamic F_x Force **(b)** Dynamic F_y Force



(c) Dynamic F_z Force **(d)** Dynamic T_x Force



(e) Dynamic T_y Force **(f)** Dynamic T_z Force

Figure 4.28: Dynamic reactive forces with experimental data and the QSS model for a tail flick at 10 m/s. The swing was performed by tail 1 in the major axis. The figures are for F_x, F_y, F_z, T_x, T_y and T_z respectively.

The three main force profiles that are looked at are F_x, F_y, and T_x. While the other three profiles are also relevant, these can be quickly skewed by an error in geometrical offsets or computational offsets. The difference between the measured and modelled forces can be seen to lie in conjunction with the difference in position profiles. With these results, it was seen to be acceptable to move onto the next section of the project where the platform could be developed and tested with the chosen QSS model for low speed situations.

4.10 Summary

In summation, the following was done in this chapter. A test rig was designed and made to allow a cheetah tail to both remain static and swing in a pitch motion in a wind tunnel. A Euler-Lagrangian dynamic model was developed to analyse the logged data from the tests to develop an aerodynamic model that can be used in further simulation steps. The static tests allowed for a quasi-steady-state model to be developed using 30 nodes per tail and tail orientation. The dynamic swings tested the addition of added mass to the QSS and found that at higher speeds, the QSS-AM model was not proven accurate due to the transition period and the turbulent flow of the air. It was also seen that the lower the relative wind velocity, the more accurate the QSS model was. This means that the QSS model can be used for relative wind speeds of 10 m/s and below.

Chapter 5: Mechatronic Design of AeroDima

This chapter depicts the steps taken for the mechatronic design of the robotic platform, *AeroDima*. This chapter will look at the mechanical design behind the gearbox and shaft structure, and the design of the robotic aerodynamic tail, followed by the design of the rest of the car. Following that design, the embedded systems design will be looked at, including the microcontroller, sensors, and software architecture.

5.1 Gearbox and Shaft Design

While this research, as seen in the later sections, only looks at a single degree of actuated motion, for future studies, this tail and gearbox structure will be designed for two degrees of actuated motion, both the pitch and the roll motion. These two degrees of motion are taken based on the motion of animal's tails and their range of motions where the tail is situated at the back of the body. An illustration of various motions can be seen in Figure 5.1 where the colour of the arrow and sides of the box represent the roll, pitch, and yaw axis respectively. The tail was designed to be actuated in the pitch and roll axis, while being able to rotate in an unactuated yaw axis.

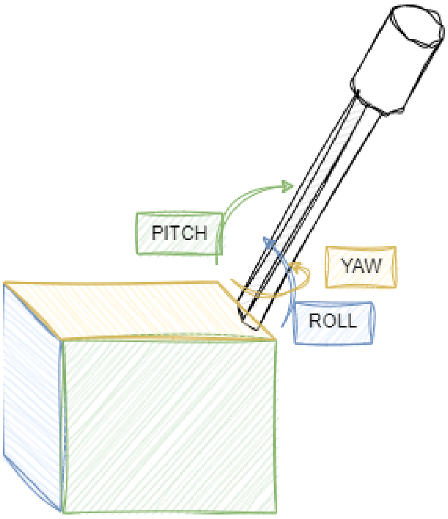
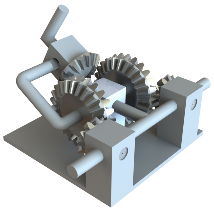


Figure 5.1: A representation of the pitch, roll, and yaw axis. Each colour of the arrows and the sides of the box represent the planes of motion.

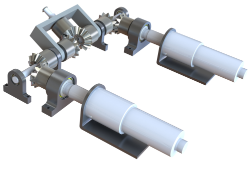
The actuated yaw axis is not a motion that this research paper is interested in. For this reason, an actuated yaw axis will not be designed for. However, this still leaves an actuated roll and pitch motion. For a system to have two actuated motions, it will need two actuators and some mechanical configuration that orientates these motions into the correct planes. For this system, two motors were chosen to be the actuators. Once this decision was made, the gearbox configuration could be designed. While the gearbox design from *Dima* II could have been used, new motors had been designed and then acquired by the researchers which had better specifications than the previous motors used. Due to the difference in form of the motors, a new gearbox needed to be designed. This gearbox went through a number of different design iterations as can be seen in Figure 5.2. The first design was a simple gearbox that was inspired by a differential gearbox and adapted from [46] and can be seen in Figure 5.2a. This design was more concept based and needed to be adapted due to the size, mass, and inertial constraints that came with the design. A major constraint was the size of the gearbox. The gearbox needed to be small enough to fit in the body of the platform while being light enough to not add too much weight to the platform. The third constraint was the inertial properties of the gearbox. The gearbox needed to have a low moment of inertia to not add a lot of rotational inertia to the platform. Taking these into consideration and adding the motors required, the next concept design was made. This design can be seen in Figure 5.2b. The motors added to this design were two Maxon motors. These motors were used in the previous platform, *Dima*, and in the wind tunnel experiments mentioned in Chapter 4.

This design was then re-evaluated, however it was seen that the size of the design was still too large to fit in the body of the platform from the basic dimensions of the platform chosen at this stage. The exact size of the platform will only be designed in the later section, Section 5.3. The large contributor to the size limitations was both the width of the gearbox due to the length of the shafts and gears as well as the length of the gearbox due to the length of the motors. For this reason, new motors were chosen which housed an internal gearbox and were much more narrow. These motors were the AK70-10 T-Motor motors. These motors were chosen due to having the motors available and that they fit the sizing and torque requirements. The torque requirements were seen from the wind tunnel usage. The T-Motors could produce a max torque of 24.8 Nm which was enough to swing the tail at wind speeds of 30 m/s. With the new motor choice, a new gearbox design was made which simplified the design significantly by reducing the complexity of the external gearbox. This in turn reduced the mass, inertia, size, and cost of the design. The new designed gearbox can be seen in Figure 5.2c. The tail shaft is not included in this figure and an elbow joint of some kind would have had to be implemented to the end of the seen to shaft to operate the tail as required.

Though this design could have worked, it could still be further optimised by reducing the weight, cost, inertia, and total size. This was done in a new design that removed any external gearing completely by rotating one of the motors. This rotation of the motor resulted in the 2DOF gearbox being able to be placed in the centre of the platform, only taking up half the space. The new design reduced the size considerably as well as removed all excess weight that come from external bearings and gears. Not only that, the new design also reduced the cost of the gearbox as it required less manufacturing. The new design can be seen in Figure 5.2d.



(a) First concept design of the gearbox, inspired by [46].



(b) Second concept design using two Maxon Motors.

(c) Third concept design using two AK70-10 T-Motors.

(d) Fourth concept design using two AK70-10 T-Motors with no external gearing.

Figure 5.2: The various iterations of the gearbox design. Subfigures (a) - (d) shows the progression of the design from the first concept design to the final design.

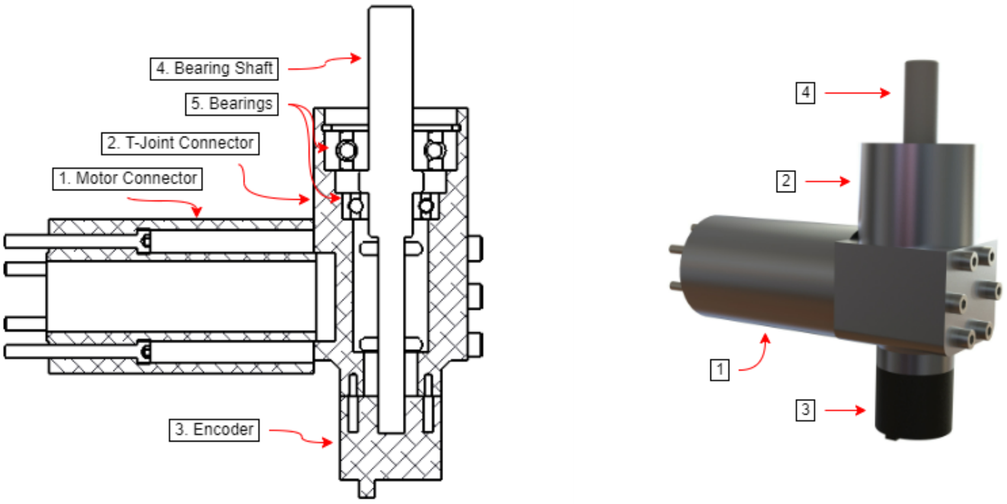
While there was conceptual additions from the second to the fourth conceptual designs, according certain certain design metrics, there was a quantitative improvement in the designs. Four metrics were looked at, the mass of the gearbox design, the total size of the design, the cost of the design, and the complexity of the design as seen by the number of parts included in the design. The mass was measured in kg and the size was measured in width x length of the extremities of the design. The cost was evaluated from the researchers grasp of manufacturing costs and the processes required to manufacture or purchase each part. Quotes were not gotten for this section and so a numerical cost could not be given. In lieu of this, cost was scaled from 1 to 10 based on materials needed and machining complexity. Lastly, the design complexity was chosen to be the number of parts needed for the design to be complete, up till the shaft holder part. The evaluation of these designs can be seen in Table 5.1.

Table 5.1: Conceptual Design Evaluation Metrics.

Design	**Mass** (kg)	**Size** (mm x mm)	**Cost** (/10)	**Complexity** (No. Parts)
2	2.793	216x267	9	55
3	1.46	241.5x125	6	23
4	1.247	155x89	4	6

Based on these evaluations, it was seen that the fourth design weighed the least, was the least size restrictive, and had the least complexity. It also had the lowest assumed manufacture and purchase cost. For these reasons, it was chosen to continue the gearbox design with this latest conceptual design. The latest design of the gearbox was then the first design to include a proper mechanism to attach the shaft of the tail to. This part was connected straight to the output of the middle motor and had a slot cut into it to allow the shaft to sit in and be fitted to it. While this design of the shaft holder was a viable design, it ignored one of the major design requirements for the tail, that of it having a third, unactuated, degree of freedom. This design kept the tail stationary and would result in the tail being unable to move along its center of rotation allowing the tail to swing in only one direction if the tail were a non-symmetric design. The design of the tail is detailed in Section 5.2, but the tail was chosen to be a non-symmetric design. For this reason, the next part designed was the shaft holder which should allow for a third degree of freedom in the tail.

This coupling part needed to be able to connect to the output shaft of the motor, it needed to allow for a third degree of freedom, and it needed to be able to measure this third degree of freedom. In the design of the part, the manufacturing processes and costing was also taken into consideration and so the part was split into multiple bodies. The first needed to be one that could connect to the motor's shaft. This part was designed to be a simple cylindrical part with mounting holes. It was designed to be hollow to reduce the mass and inertia of the part. The next body to be designed was the T-joint connector. This part needed to connect to the motor connector, while housing multiple bearings to allow for the third degree of freedom. This was done with a dual bearing setup with space allowed for thermal expansion. The internal parts were held in with an internal circlip. Attached to the bearing was a shaft that would be freely able to turn. This alone would not be able to connect to the tail shaft, however with the use of a simple coupler, the shaft was extended all the way to the sail section of the tail. The sectioned and isometric view of the T-joint can be seen in Figure 5.3. Attached to the bottom the T-joint connector is an encoder that connects to the end of the bearing shaft to measure the angle of the tail.



(a) Sectioned view of the shaft holder. **(b)** Isometric view of shaft holder.

Figure 5.3: The shaft holder that is attached to the T-Motor and the tail shaft. The part is made up of five main parts, the motor connector, the T-joint connector, the encoder, the bearing shaft, and the bearings.

With this design done, it was combined with the fourth conceptual design, the T-Motor gearbox design, to produce the final design of the gearbox and shaft holder. A few things were adjusted from the conceptual design of the gearbox to the final design to strengthen the design mechanically, so not to worry about bending stresses. The way the middle motor was mounted to the outer motor was also adjusted to bring the motor's shaft closer to the center of rotation, removing the weak point of a single bend while also reducing total inertia for pitch swings.

Table 5.2: Conceptual Design Evaluation Metrics.

Design	**Mass** (kg)	**Size** (mm x mm)	**Cost**	**Complexity**
Final	1.225	159x89	4	5

The final evaluation metrics of the design can be seen in the Table 5.2 above, while the final render of the gearbox with the shaft holder can be seen in Figure 5.4.

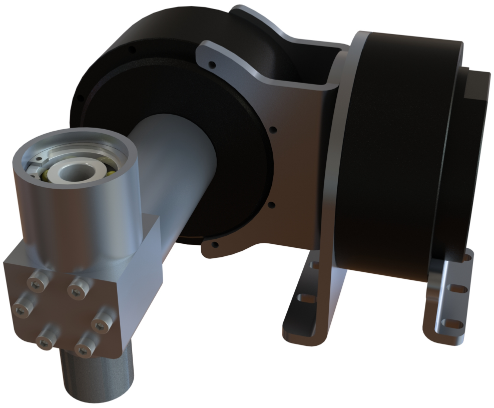


Figure 5.4: Final Design of the gearbox using two T-Motors with the shaft holder.

5.2 Design of Aerodynamic Robotic Tail

Bio-inspired robotic tails are not common and while some research has been done into them, see Section 2.2.2, an optimal shape for an aerodynamic tail has not been studied. The two aerodynamic tails that have been used, on the Ghost Robotics Minitaur [29] and the SailRoACH [36], have used commonplace shapes. The SailRoACH has used a flat plate as a sail, while the Minitaur used a half cylinder shaped tail. In this section, the design of the tail is discussed, and the results of shape analysis using SolidWorks Flow Simulation are presented.

Eight shapes were studied chosen to be compared. These shapes were inspired by the shapes used on previous aerodynamic robotic tails, as well as taking inspiration from other sources of shapes that have large drag coefficients. The first six shapes studied come in pairs, one being the shape by itself while its partner being the shape with a backing board. The reason why the shape with a backing board was chosen was due to the design condition that the tail has an extra degree of freedom, one that allows it to rotate along the axis of the tail. This backing board would then act as a rudder which would ensure that the tail would constantly be facing into the largest wind vector, thus providing as much drag as possible. The three shapes included a flat tail design (FT), a half cylinder design (HC), and a half cylinder with end caps design (HCEC). These shapes were also studied with a backing board which resulted in the six shapes. The final two shapes studied were inspired by parachute design due to a study that looked into the aerodynamics of various parachute designs [47]. Two shapes where taken and adapted in this study due to their ability to be designed as a solid structure. These designs were the disc gap band (DGB) Mars pathfinder (MPF) and the DGB Taper. What differentiates these designs from the first six designs are their gaps between the backplates and the front air shaping structure. These shapes can be seen in Figure 5.5.

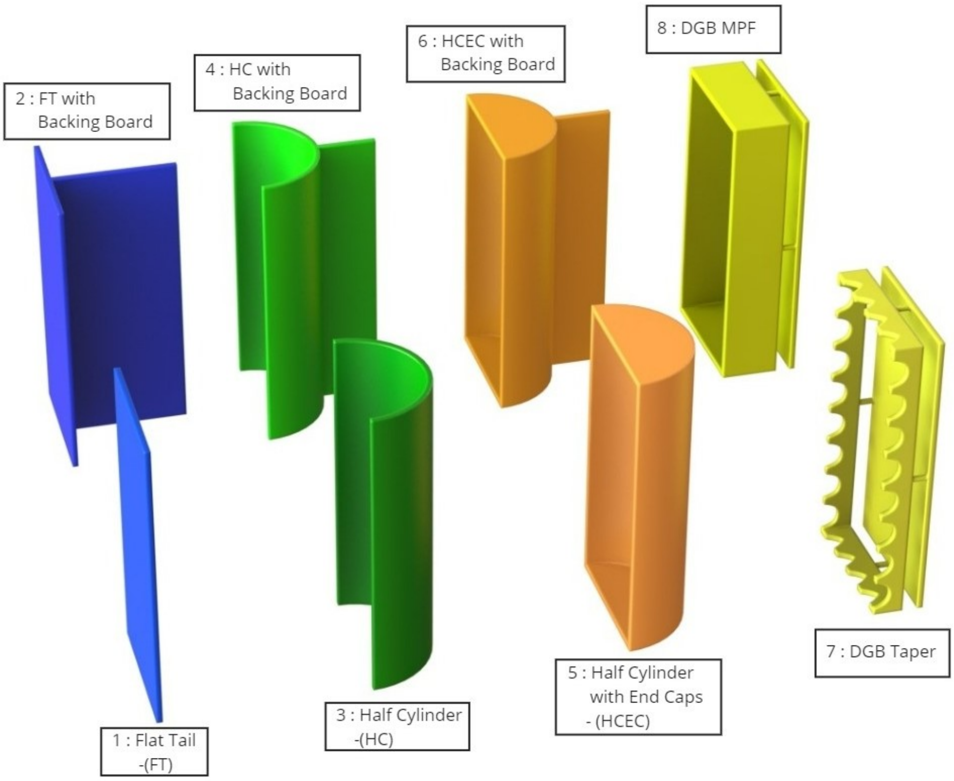
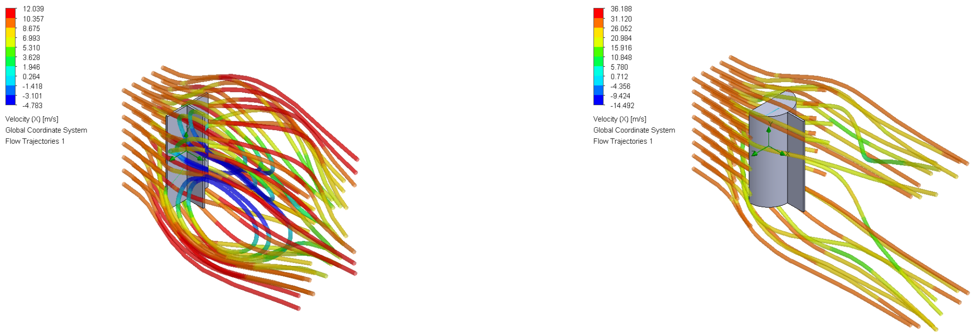


Figure 5.5: The 8 different tail designs that were tested in SolidWorks Flow Simulation.

Once these designs were chosen, a flow simulation was done on them using SolidWorks Flow Simulation. SolidWorks uses a computational fluid dynamics (CFD) solver to simulate the flow of air around the shapes. The specific scenarios were chosen to replicate those of the static tests of the tails in the wind tunnel as described in Section 4.4 where three different wind speeds were tested with six different angles of attack. Using the Flow Simulation's goalplots, the coefficients of drag and lift at the various wind speeds and angles of attack were calculated. Examples of the resulting airflow around the shapes can be seen in Figure 5.6.



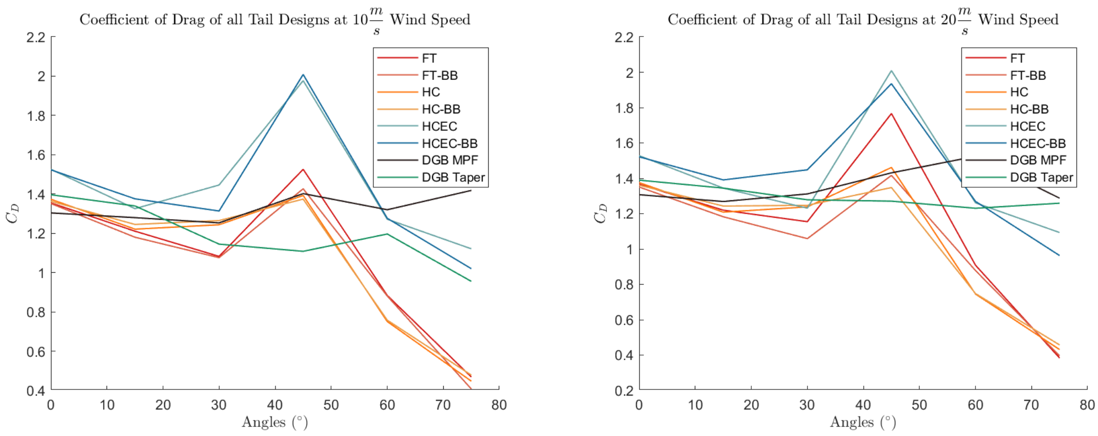
(a) Flow trajectory of the FT design at 10 m/s. **(b)** Flow trajectory of the HCEC design at 30 m/s.

Figure 5.6: The flow trajectories of the air around the tail designs at 10 m/s and 30 m/s.

After obtaining the coefficients by creating a global equation, which makes SolidWorks solve for a certain variable in the simulation, the various shapes were compared. This global equation, as called in SolidWorks, is a formula that is solved by taking the variables within the model. The global equation used is seen in Equation 5.1. In this equation, F_A is determined using SolidWorks built in force calculator in the same direction as the fluid flow, ρ is the air density, A is the front surface area of the tail given by 5e-3 m^2 due to the size of the tail, and v_{wind} is the velocity of the air.

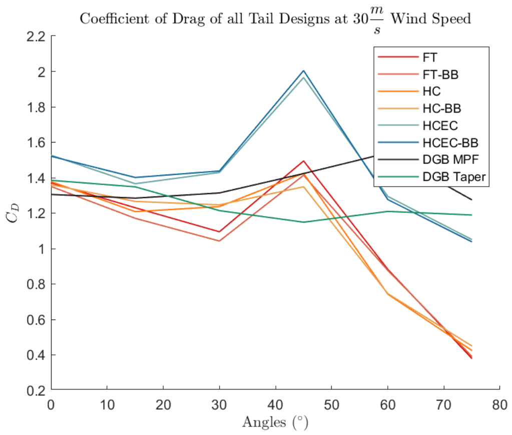
$$C_D = \frac{2 * F_A}{\rho * A * {v_{wind}}^2} \tag{5.1}$$

It was seen that the HCEC designs had the highest drag coefficients across the various angles of attack. This can be seen in Figure 5.7 and the result aligned with the various shape's coefficients as seen in Table 2.1. Due to these results, it was then designed that the HCEC shape would be used as the final design.



(a) 10 m/s Wind speed

(b) 20 m/s Wind speed



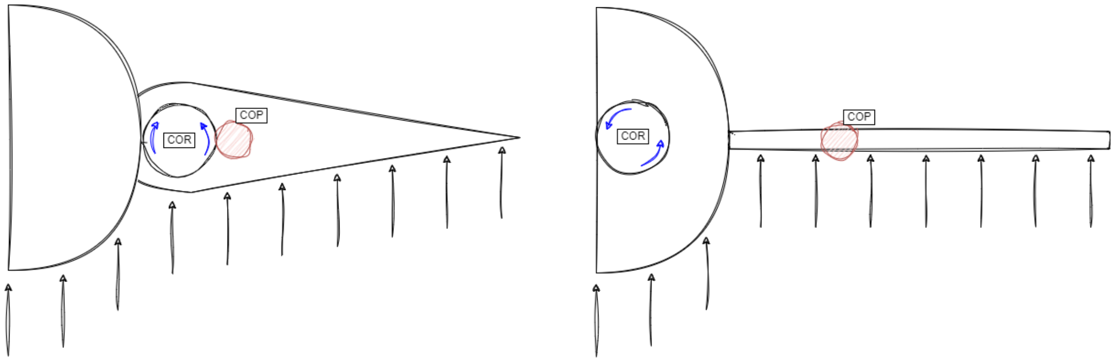
(c) 30 m/s Wind speed

Figure 5.7: The drag coefficients of all eight tail designs for varying wind speeds across various angles of attack. The results are shown for the 10 m/s, 20 m/s and 30 m/s wind speeds in their respective subfigures.

Although the HCEC design had been chosen, the decision over the backing board remained. The backing board was ultimately chosen to allow for the tail to freely rotate into the direction of the wind. With the tail shape chosen, the tail needed to be designed properly. Due to the concept of the tail being one whose sole purpose was aerodynamic drag, the materials chosen to make up the tail needed to be lightweight. For this reason, a mixture of aluminium and carbon fibre was chosen for the structural components. The tail's shaft would be a carbon fibre shaft, the half rings gripping the shaft would be aluminium which the supporting shaft between the two half rings would be carbon fibre. The tail's backing board would be 3D printed and made of PLA. The design was then done in SolidWorks and rendered as can be seen in Figure 5.8. The sail material would be made out of a light plastic film, similar to those found on the front of plastic binders, allowing for the rigidity of the sail while maintaining the low mass element of the design.

Figure 5.8: The final tail design that was tested in SolidWorks Flow Simulation.

This design was manufactured and tested. It was seen in testing that the tail rotated in non-desired directions around the third degree of freedom of the tail: the backing board didn't aid to angle it into the wind. After consideration, it was noted that the design's center of pressure (COP) was located too close to the center of rotation (COR). This meant that, depending on the strength of the wind, the tail could angle either way into the wind and to solve this issue, either the COP or the COR needed to change. While it was possible to change the COP by increasing the surface area of the backing board, this would add weight constrictions as well as make the tail too large and bulky. To avoid this issue, the COR was brought closer to the front of the tail. This increases the moment arm and allows for the tail to rotate correctly into the wind. This concept change in COR can be seen in Figure 5.9.



(a) An illustration of the center of pressure of the original tail design.

(b) An illustration of the center of pressure of the final tail design.

Figure 5.9: A sketch of the difference in designs from the original to the final design of the tail. The arrows are a uniform distribution of wind pressure. The further the center of pressure from the center of rotation, the more the tail will angle itself into the wind.

With this change in mind, the tail's design needed to change. The plastic film that acted as the end caps on the first design was removed and merged with the half rings aluminium pieces. This, due to the larger weight, was changed from aluminium to 3D printed PLA. The backing board was also changed, this time to a multilayered stiff paper design to reduce weight and complexity. These design changed can be seen in Figure 5.10.

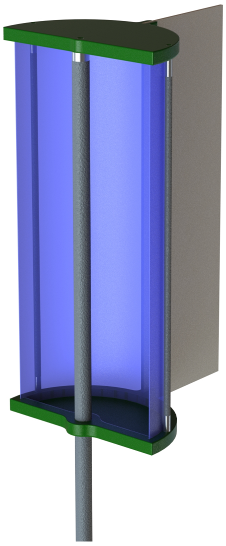
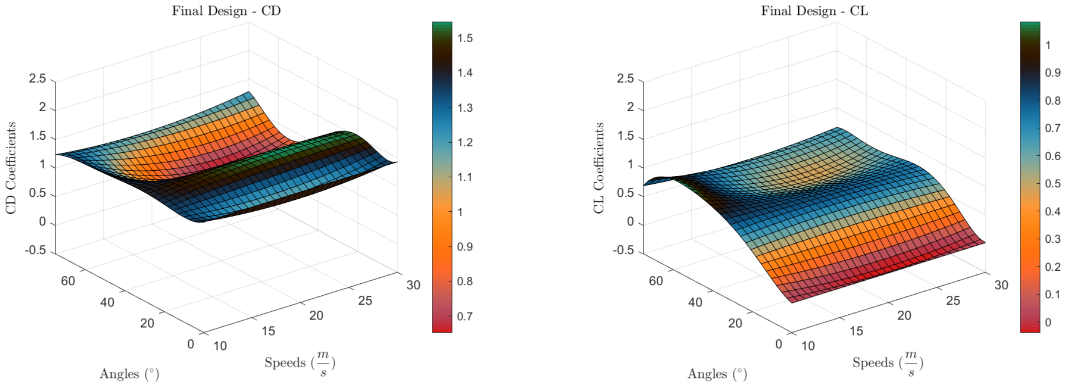


Figure 5.10: The final tail design that was tested in SolidWorks Flow Simulation.

Using this tail design, another flow simulation calculation was run. This was to get the coefficients model to use for the model simulation. These results will be compared to experimentally tested results that will be done in Section 6.3. A surface plot of the drag and lift coefficients interpolated from the 30 nodes of results of the final tail design can be seen in Figure 5.11.



(a) Surface Plot of the Coefficient of Drag. **(b)** Surface Plot of the Coefficient of Lift.

Figure 5.11: Surface Plots of the Coefficient of Drag and Lift of the final tail design with varying wind speeds and angles of attack.

5.3 AeroDima's Body Design

AeroDima's design was utilised the concept of *Dima* II, used for inertial tail studies done by Patel [35]. As the platform was based on the previous platform's, certain sections of the body design were reused. The parts that were reused were mainly from the original RC vehicular structure. *Dima* was built off Traxxas's consumer model, the Traxxas Stampede RC Monster Truck [48]. The chassis was used as the base for the body design. The stampede was altered, however, from the regular model. The suspension unit as well as the wheels were changed to upgraded parts. The chassis was also modified to fit a plate above it to house all the electronics. *Dima* also changed the electronic speed controller (ESC) and the two motors, the steering servo motor, and the drive motor. For the purpose of this research project, the base chassis was reused from the *Dima* platform, however the plate above the chassis as well as the motors were all changed. The parts that were reused from *Dima* can be seen in a render of the design in Figure 5.12.

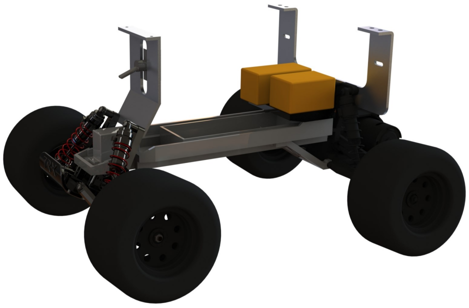


Figure 5.12: Parts of the chassis that was used from the previous platform, *Dima*.

With the base chassis sorted, the first step in this design process was to design the plate upon which everything would sit. This was designed to have a similar shape to *Dima's* plate however was modified to be more modular and lighter. The design finalised included mounting holes for the gearbox, cut out sections to reduce the mass, as well as slots for mounting other components. Due to the time frame of the design with the mechanical aspect needing to be done before the electrical design was done, the exact placements of the electronics was unknown. For this reason, slots were used in the design to allow for the electronics to be placed on the plate when the electronics were finalised. Along with the plate, a roll cage was also added to the design to protect the electronics and the rest of the chassis from ground impact that will be experienced during testing. The roll cage was also designed to be able to be removed. While the cage was welded together, it was fixed to the plate with lock nuts. The plate was also designed to be able to be removed from the chassis and was held in place with lock nuts as well to allow for easy assembling of the car and electronics for testing purposes. The design of the plate and roll cage can be seen in Figure 5.13. In this figure, the T-Motor gearbox and shaft holder was included to show the protective effects of the cage. This roll cage, in testing, proved to be sufficient in protecting all internal components of the platform.

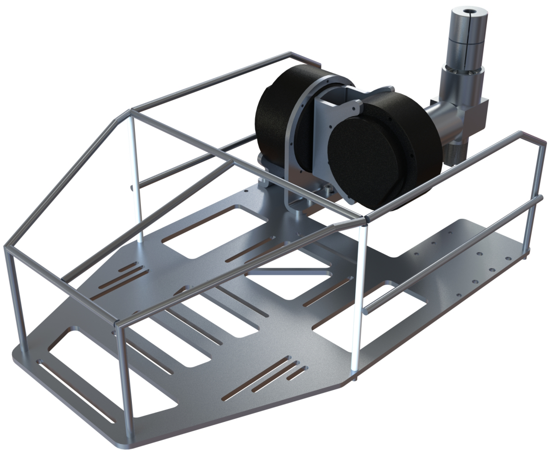


Figure 5.13: The upper plate that was added to the chassis to house the electronics and the gearbox as well as the roll cage.

With the tail length, the placement of the motors, and the physical boundary of the roll cage, the range of motion of the 2DOF tail could be calculated using the physical layout of the car and taking boundary points at certain angled intervals. This full range of motion graph would not be incorporated into the single swing, however, it will be useful for a full 2DOF motion control system. To move the tail to a specific position in the 3D environment would require this range of motion. The range of motion of the tail can be seen in Figure 5.14.

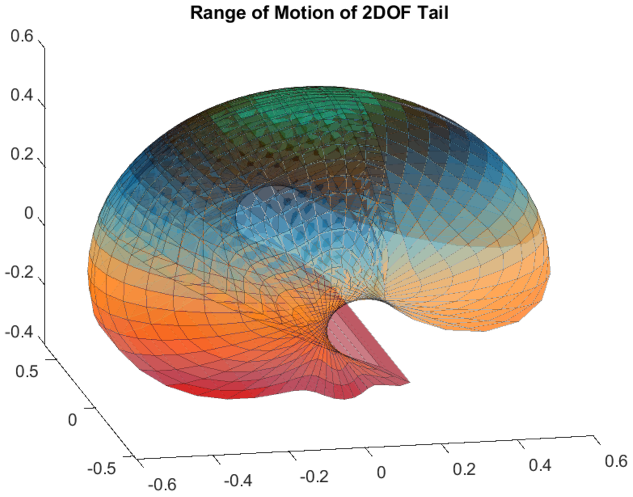


Figure 5.14: The range of motion of the tail.

The design of the body had a few design considerations which included the height of the center of mass, the weight of the car and how the electronics would be able to fit in. The center of mass's height needed to be high enough to produce a toppling effect when turning as the higher the center of mass, the greater the centrifugal moment. The weight of the also plays a large roll. The larger the weight, the more torque would be required to drive the car and without the correct motor, this would result in the car driving slower than desired. These values were not explicit but were included in the thought process of the design. The electronics also needed to be able to fit in the car. While the electronics are only discussed in Section 5.4, the full design of the car, including electronics can be seen in Figure 5.15. In the figure, there is a wooden board attached to the rear end of the car. This was used as an anti-ramp board (ARB). It was seen later in testing with a previous motor and less weight, the car had sufficient torque to flip the car and so this ARB was used to stabilise it during acceleration manoeuvres. The final car can be seen in Figure 5.16. In this image, it can be seen that adhesive taps are employed on the backing plate of the tail. Ensuring the position of the backing plate was a challenge while wanting to maintain the lightweight properties of the tail. Therefore, lightweight mounting options were employed to ensure that the structure of the tail maintained its desired form.

Figure 5.15: Final Render of *AeroDima*

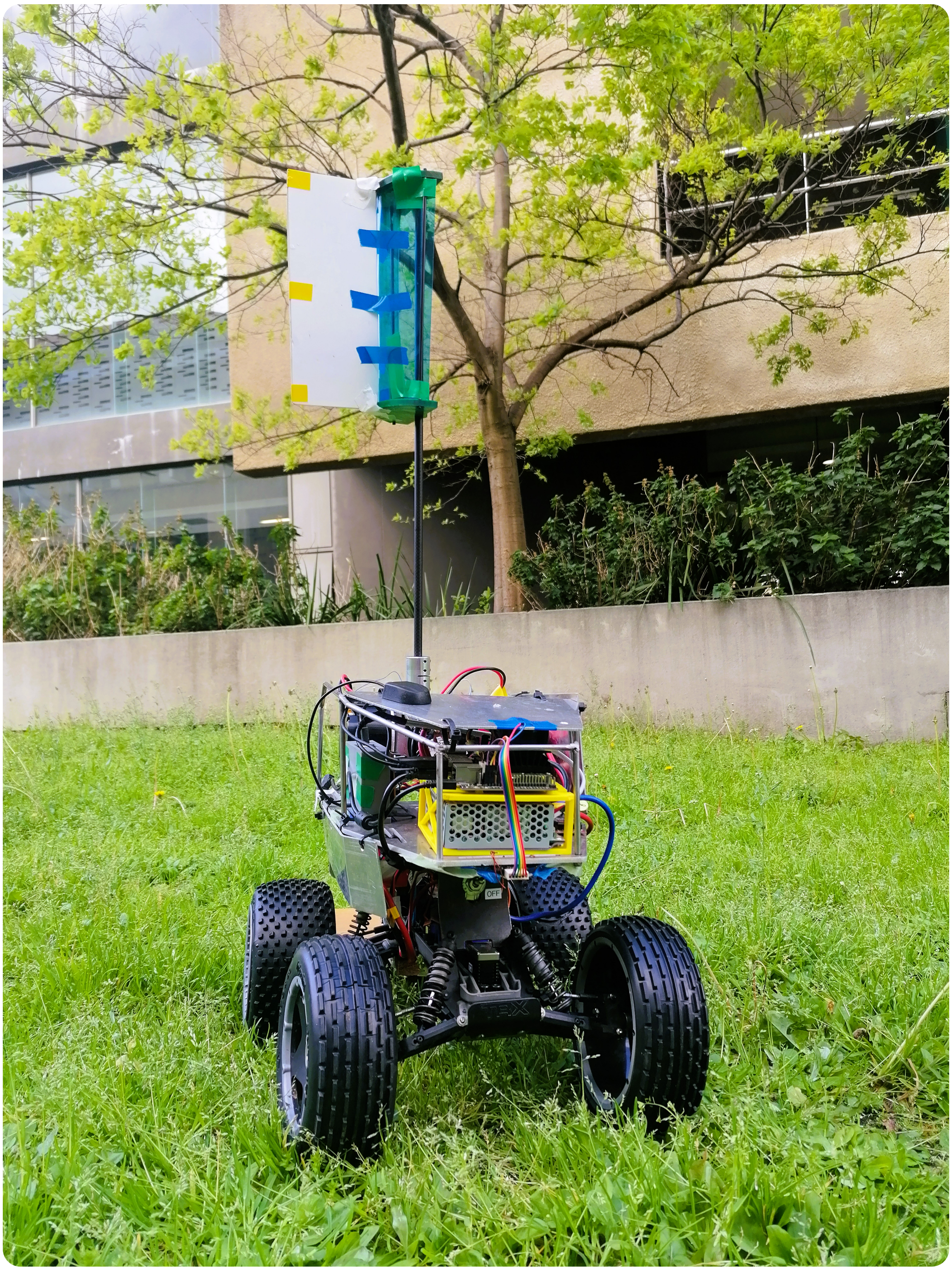


Figure 5.16: A portrait of *AeroDima*.

5.4 Embedded Design

The embedded design of the research includes the battery system design, the motor system design, the sensory system, the microcontroller system, and the communication system. This section details hardware and software design of the electronic components included in *AeroDima*.

5.4.1 Battery System

The battery setup needed for the system amounted to two sets of batteries. These being a 7S LiPo made up of a 3S and 4S in series to power the T-Motor AK70-10 that require an input voltage from 24 V to 48 V , and a 4S LiPo used to power the rest of the electronics. For over current protection, 30A ceramic slow blow fuses were added to the battery leads to protect the components from damage which could occur at high current draw. The three batteries used on the car are shown in Figure 5.17.

(a) 3S 4200 mAh LiPo **(b)** 4S 4200 mAh LiPo **(c)** 4S 5200 mAh LiPo

Figure 5.17: The three LiPo batteries used on the car. The two 4200 mAh batteries are connected in series to power the tail motors, while the 5200 mAh battery powers the rest of the electronics. Images taken from [49].

5.4.2 Vehicle Motor System

The car has two motors, a servo motor used for steering and a brushless DC motor used for throttle. The servo, a CLI-380 servo motor was powered by 6 V supplied by a universal battery eliminator circuit, the Hobbywing 10A UBEC. The servo used is shown in Figure 5.18a.

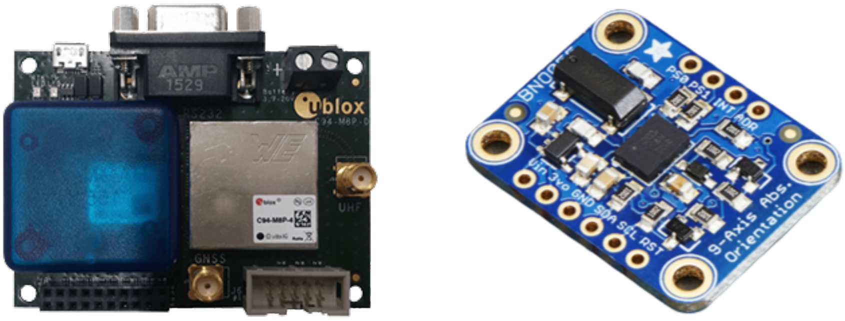
The brushless DC motor chosen was the sensored Hobbywind Xerun G3 2250kV. This was operated by an electronic speed controller (ESC), the Novak Crusher ESC. The brushless motor can be seen in Figure 5.18b.

(a) The steering servo. Image taken from [50]. **(b)** The brushless motor. Image taken from [51].

Figure 5.18: The two motors used on the car, namely a CLI-380 38Kg servo motor (a) and a Hobbywing Xerun G3 2250kV brushless motor (b).

5.4.3 Sensory System

To operate correctly, the system needs to be able to sense its environment. The sensors used were the following: a GPS and an IMU. The GPS was needed to determine the vehicle speed while the IMU was needed to determine the vehicle's orientation, angular velocity, and angular acceleration. These data points are important to both the control system, and the logging and comparison system. For these components, the C94-M8P GPS unit and the BNO055 9-DOF IMU was used. The GPS unit needed to be mounted to the car somehow and so a casing was designed and 3D printed for the unit to sit in which was mounted to the plate. The IMU unit was rigidly mounted to the plate using a mixture of adhesives and mounting wire. The two units can be seen in Figure 5.19.



(a) GPS Unit **(b)** IMU Unit

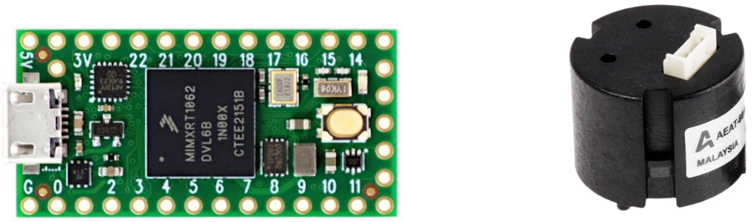
Figure 5.19: The C94-M8P GPS unit and the BNO055 9-DOF IMU unit. Images taken from [52] and [53] respectively.

5.4.4 Microcontroller System

The control of the system came from the microcontrollers. The microcontroller system was responsible for the following: controlling the vehicle and tail, and logging the data. The microcontroller system was made up of the following components: a microcontroller, a Jetson Nano, and a DC-DC converter. The microcontroller was used as an intermediate controller to communicate with the motors. The microcontroller used was the Teensy 4.0 which has built-in PWM, serial, and CAN communications. This allowed it to communicate with the tail motors which communicated through the CAN communication protocol, the servo motor and the brushless motor which communicated through a digital PWM signal, and the Jetson Nano which communicated through a serial communication protocol. The Jetson Nano was used to control the system, log the data, and communicate with an external computer. The Jetson Nano was used as it has a high computing power which was required for a study that was working in parallel with this research that required camera vision and image processing. The DC-DC converter was used to convert the 14.8 V from the 4S LiPo to the 5 V that the Jetson Nano required. The DC-DC converter used was the SD-25B-12 DC-DC converter and can be seen in Figure 5.20.

Figure 5.20: The SD-25B-12 DC-DC converter. Image taken from [54].

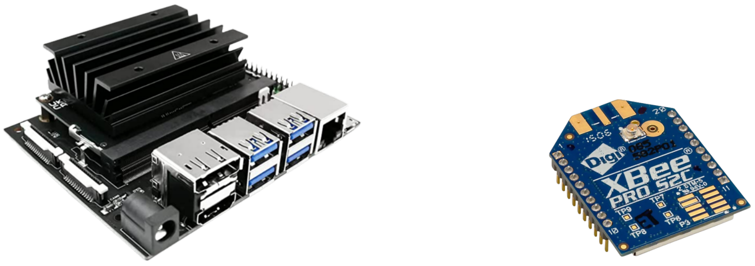
To communicate with the tail motors, a daisy chain communications protocol, controller area network (CAN), was needed. However, the Teensy does not include a CAN transceiver which converts the CAN signal to a CAN high or CAN low signal which the motors can read. A TD322DCAN transceiver was used for this purpose. The data structure sent will be explained in the Section 5.4.5. The sail's angle was also needed to be recorded and so the Teensy needed to read the angle through the encoder used. The encoder used, the AEAT-6010-A06 absolute rotary encoder, was connected to the digital input put of the Teensy. The Teensy was also used to control the steering servo motor and the brushless motor and both motors were controlled through a PWM signal. The Teensy microcontroller and the rotary encoder can be seen in the Figure 5.21.



(a) Teensy 4.0 **(b)** Rotary Encoder

Figure 5.21: The Teensy 4.0 microcontroller and the AEAT-6010-A06 absolute rotary encoder. Images taken from [55] and [56] respectively.

The Teensy and the Jetson Nano communicated through serial communication. The Nano, which is essentially a mini computer that runs a Linux operating system, was used to log the data from the sensors and the motors. This data would be stored onboard the Nano and retrieved at a later stage after testing was done. To control the car however, inputs from the user were required and this part required communication between an external computer and the Nano. This was achieved through the wireless communication of RF transceivers. The two transceivers used were the XBee-Pro S2C RF transceivers where these modules can support a long-range line-of-sight communication distance of up to 1200 m. The Jetson Nano and the RF transceiver can be seen in the Figure 5.22.



(a) Jetson Nano (b) RF Transceiver

Figure 5.22: The Jetson Nano and the XBee-Pro S2C RF transceiver. Images taken from [57] and [58] respectively.

A final list of all the components uses can be seen in Table 5.3.

Table 5.3: *AeroDima*'s electronics list.

Component	**Details**
Steering Servo Motor	CLI-380 38Kg
Throttle Brushless DC Motor	Hobbywing Xerun G3 2250kV
Tail's Brushless DC Motor	T-Motor AK70-10
UBEC	Hobbywing 10A UBEC
ESC	Novak Crusher
Tail Battery	7S LiPo 25C 4200 mAh
Electronics Battery	4S LiPo 25C 5200 mAh
GPS	C94-M8P GPS Unit
IMU	BNO055 9-DOF IMU
Microcontroller	Teensy 4.0
Processing Unit	Jetson Nano
DC-DC Converter	SD-25B-12 DC-DC Converter
CAN transceiver	TD322DCAN CAN Transceiver
Encoder	AEAT-6010-A06 Absolute Rotary Encoder
RF Transceiver	XBee-Pro S2C RF Transceiver

5.4.5 Communications Protocol

Three main communication channels were needed the embedded system. The first was the channel between the computer and the Jetson Nano, the second was the channel between the Nano and the Teensy, and the last channel was the channel between the Teensy and the motors. To start, the last channel will be discussed first. To power the servo and the brushless motors, a PWM signal is sent from the Teensy at a 5 V high. The brushless motor required a microsecond signal of between 1100 μs and 1900 μs with an idle signal of 1500 μs. Any signal larger than 1500 would result in the car being driven forward while and signal lower would result in a reverse motion. The steering servo motor required a signal of between 950 μs and 2050 μs with an idle signal of 1500 μs. A range between these signals would result in a range of steering motion from left to right. For these PWM signals, the Arduino *Servo* library was used. The Teensy was also required to read an angle from the rotary encoder and this was done through reading a digital input send by the encoder at a 3.3 V signal and shifting it to a 12 bit number. This number was then mapped to a 0° to 360° range. The input was recorded at a rate of 1000Hz. Lastly in this channel, the Teensy needed to communicate with the tail motors using the CAN protocol. This meant that each message sent to the motors needed to be in a specific format. For this format, the *FlexCAN T4* library was used. The library requires a message ID and a message buffer in which the ID determines which device the message is sent to. Due to CAN being daisy chain, all devices on the chain receive all the messages but for the message to be read and accepted, the message ID needs to match with the device's CAN ID. The message buffer is the data that is sent to the device. This data structure is specific to the device. The AK70-10 motors require a position, a velocity, a KP, a KD, a torque, and a state value. The various states that the motor can be that are used are: enter motor control, exit motor control, set origin position, and neutral mode. To operate the motor, an origin position needs to be set and this needs to be done out of the motor control mode. Once that has been completed for the motor to be actuated, the motor first needs to be set in motor control mode after which the position, velocity, KP, KD, and torque values need to be set. This results in a data structure that is sent to the motor as seen in Figure 5.23. How this message was coded before being mapped to its respective message can be seen as follows:

"ID ; STATE ; POS ; VEL ; KP ; KD ; TOR |"

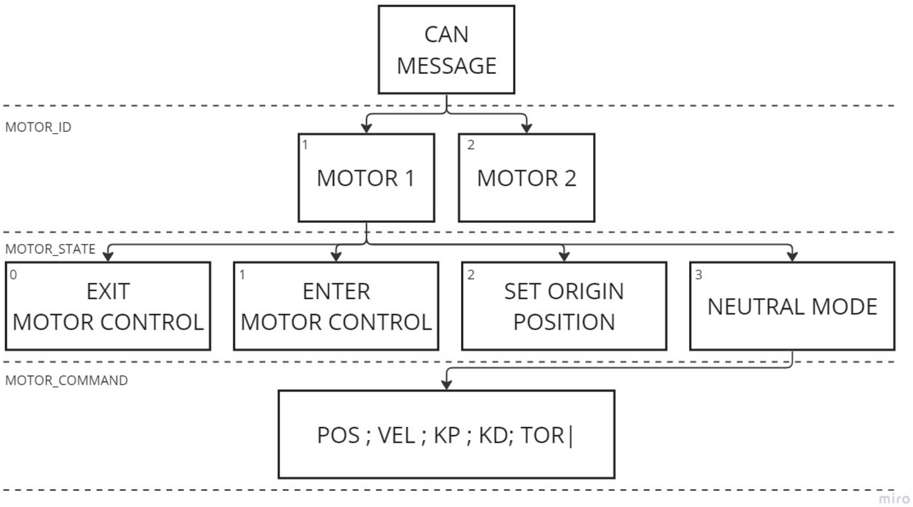


Figure 5.23: CAN message structure of the AK70-10 Motor.

The communication between the Jetson Nano and the Teensy used a similar structure, except it used a serial string. This included the commands to the two tail motors and the commands to the servo and brushless motor. The string is sent in the following format where the can message for each motor was as previously discussed:

"CAN MSG 1 ; CAN MSG 2 ; SERVO CMD ; THROTTLE CMD |"

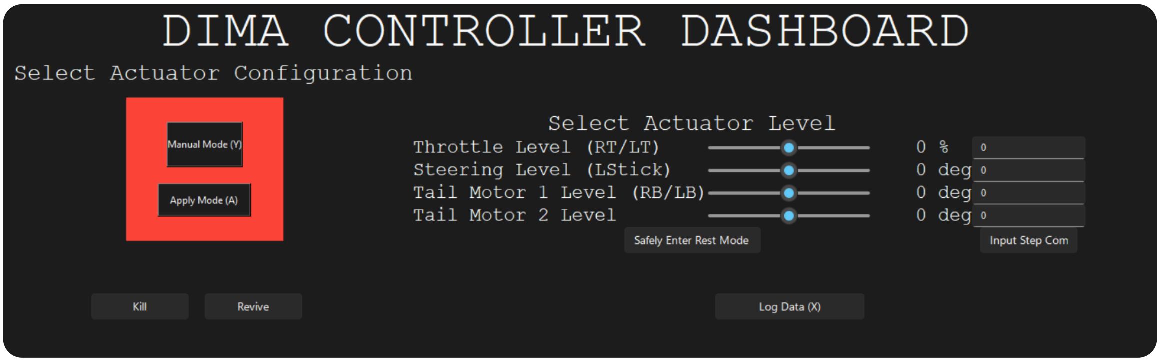
The '|' character was used as a delimiter to identify the end of the string. Once the Teensy received a command, it replied with a message which has the format of:

"ID 1 ; POS 1 ; VEL 1 ; CUR 1 ; ID 2 ; POS 2 ; VEL 2 ; CUR 2 ; ENC POS |"

The last remaining communications channel was the one between the Jetson Nano and the computer. This was done through two XBee-Pro S2C RF transceivers. One transceiver was connected to the Jetson Nano and the other was connected to the computer. To do this ergonomically, a graphical user interface (GUI) was developed. This GUI had the options of changing the states, logging the data, and sending certain commands as defined by either a slider, a step input, or a gaming remote. The GUI was developed using the *TKinter* library. The GUI was controlled by the user either through keyboard and mouse inputs or through the use of a gaming remote when a more manual driving was desired. The gaming remote was connected to the computer through a USB port. The gaming remote was then connected to the computer and used the *XInput* library to read the inputs from the gaming remote. Using the variables in the GUI, the command structures were sent to the Jetson Nano through the RF transceiver. This used the *XBeeDevice* library. This library required the data to be sent in a byte array with a defined structure. To receive the data on the Nano's side, the same structure was required to decode the message. The structure of the messaged was 'BBffff' which represented 'Bytes' or 'floats' and was defined as follows:

"CLASS | MODE | THROTTLE CMD | STEERING CMD | PITCH TAIL CMD | ROLL TAIL CMD"

In this structure, the class is used to identify the type of message that is being sent, such as whether it needs to send a message, whether it needs to start logging, or whether it needs to do an emergency stop. The mode defines what mode the car needs to drive in, from a manual drive mode, to a control drive mode, to a resting mode. The four command floats that make up the end of the message are the commands to the servo, the brushless motor, the pitch tail motor, and the roll tail motor to be used for the manual or the automatic drive mode. These values are set to 0 for all other modes. The GUI developed can be seen in Figure 5.24.



Throttle Level (RT/LT) 0 % 0
Steering Level (LStick) 0 deg 0
Tail Motor 1 Level (RB/LB) 0 deg 0
Tail Motor 2 Level 0 deg 0

Figure 5.24: GUI to send commands to the Jetson Nano.

Lastly, the starting of the program and obtaining the data through the Nano was done through a secure shell (SSH) method. For this to work, the Nano first needed to be connected to the computer through a serial connection which allows the Nano to connect to a local network. Due to the Nano being attached to the car, it needed to be then connected to a wireless network. This was done through the creation of a personal network hotspot with the use of a phone's mobile data. Once the computer and the Nano were connected to the same network, the computer could SSH into the Nano and, through the now connected terminal, run the programs. To obtain the data after experiments were run, the logged files could be downloaded through the same connection to the computer.

A holistic view of the embedded design can be seen in Figure 5.25. This includes red lines for power, blue and green lines for communication.

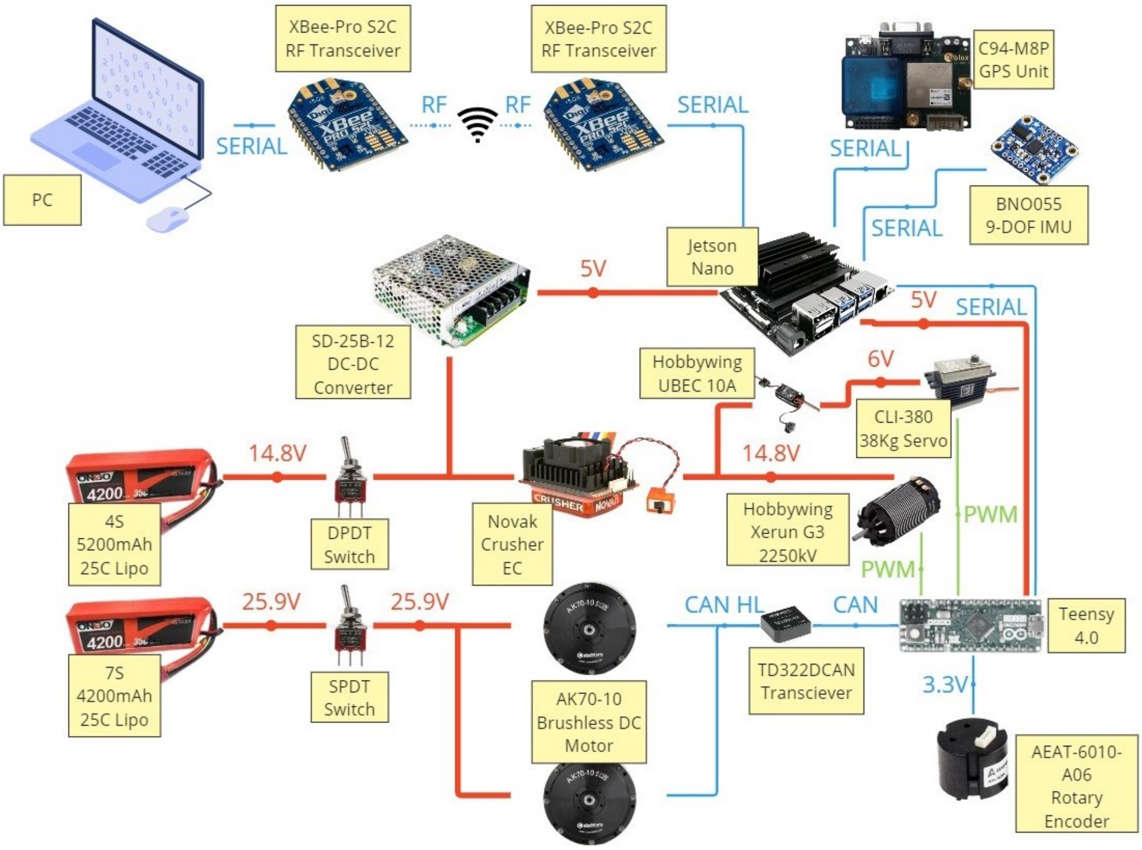


Figure 5.25: Holistic view of the Embedded Design of the car.

5.5 Summary

In summation, this chapter looked at the mechatronic design of *AeroDima*. This design incorporated the design of the chassis and body of the car which was based on the previous platform, *Dima*, but still largely changed. The mechatronic design also looked at the design of a gearbox and shaft holder to allow a 2DOF actuated tail with a third unactuated degree of freedom which allowed the sail aspect of the tail to constantly face the largest relative wind. The sail shape was also chosen by comparing 8 different designs that were analysed using SolidWorks' Flow Simulation module. The last part of the mechatronic design showed the electronic components used and the communications protocols that enabled everything to work together. These were controlled by a Jetson Nano and a Teensy 4.0 microcontroller.

Chapter 6: Simulation Modelling

This chapter will look the modelling of the simulation of the platform. In this section, all multibody models used within the simulations were obtained through converting the SolidWorks models from the mechatronic design into an .STL file that Simscape can use to determine shapes and frames.

6.1 Simscape Multibody Model Development

The simulations were performed using MATLAB's Simscape Multibody simulation environment. To fully simulate the system, all components of the platform needed to be included. These components include each of the wheel systems, the chassis, the drive motor and gearbox, as well as the tail's aerodynamic model. In this section, each of these subcomponents will be elaborated on, while the tail's model will be described in the Section 6.3. Included in this section are experiments that were run to verify certain parameters like the spring constants of the suspension units and the straight line speed model.

6.1.1 Wheel System

AeroDima is a four wheeled car, with each wheel having its own suspension unit. On the platform the front wheels were connected to the chassis while attached to a steering mechanism, while the rear wheels are connected to chassis with a universal joint to transfer the drive power. As the front and back subsystems were different, these subsystems needed to be modelled independently.

Each front wheel was condensed to consist of five main simulated parts. Some of the physical links were excluded from the model due to the way Simscape models joints and bodies. However, due to their lightweight characteristics while just functioning as physical links or rods, these parts were excluded and replaced with simple linear or rotary joints. This also aided to reduced computing complexity as the more closed loop connections that there were, the longer the model took to compile. The parts that were modelled fully without simplification were the simple spring-damper suspension, the bottom connector, the top connector, the steering parallel joint, and the wheel. This resulted in the front wheels being modelled as seen in Figure 6.1.

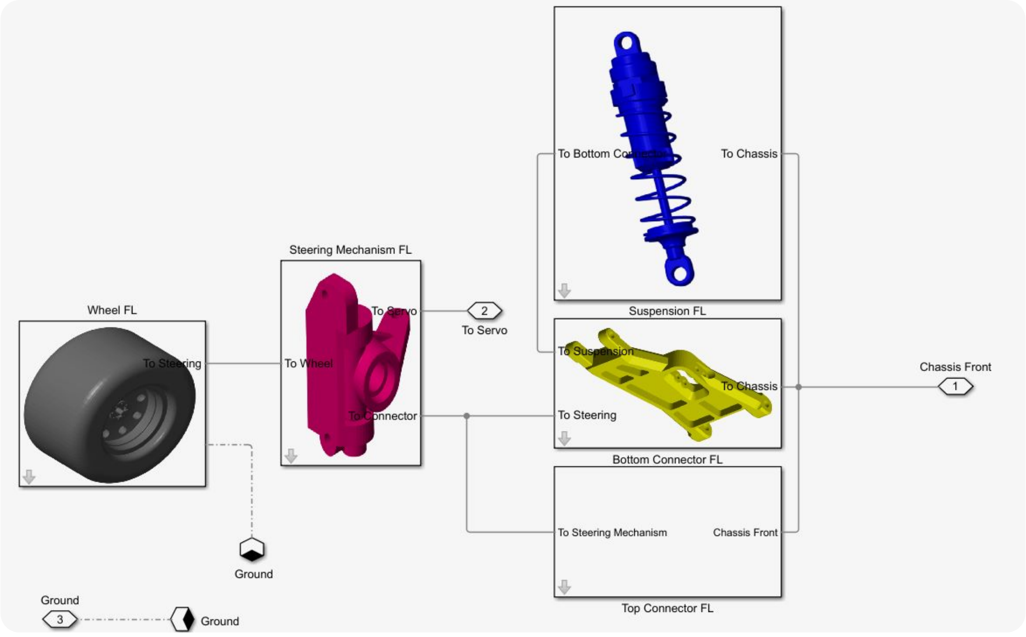
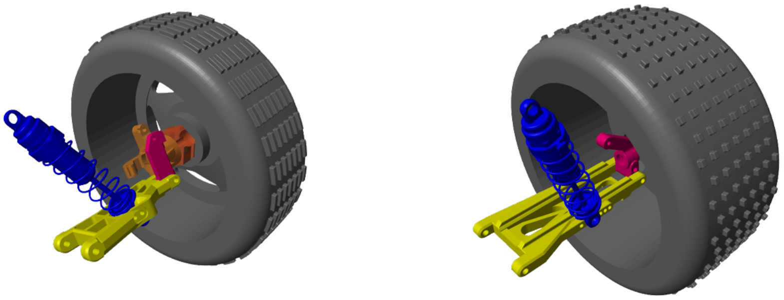


Figure 6.1: The subcomponents making up the systems of each of the front wheels in the Simscape Multibody simulation.

The bottom and top connectors were just static bodies and were connected in such a way to provide a parallel frame which ensured that the steering mechanism, and in turn the wheel, remained in a vertical position regardless of the suspension angle. The steering parallel joint was a revolute joint that connected the steering mechanism to the servo mounted on the chassis which was used to steer the front wheels and was connected to the servo motor in a separate subsystem. The tyre itself was made of two main parts: a cosmetic part and a functional part. The cosmetic part was the frame that can be seen in the simulations with a computationally complex geometrical shape with no mass or inertia. The functional part was a simple cylinder with the same properties as the wheel except it included the mass and inertial properties. This approximation of the tyre was made to reduce the complexity of the model due to the complex structure of the tyre. While possibly reducing the accuracy of the model, this approximation was chosen to allow the simulation to perform acceptably when weighing up computation complexity and compilation time. The tyre then interacted with the reference ground plane through a spatial contact force preventing the tyre, and by extension the rest of the vehicle, from phasing through the ground.

The rear wheels were modelled similarly to the front wheels and these simulated models can be seen in Figure 6.2. They, however, weren't connected to the steering servo, but rather to the differential gearbox through a universal joint shaft. The rear wheels were also connected to the chassis through a simple spring-damper suspension unit.



(a) Front Wheel Model.   **(b)** Rear Wheel Model.

Figure 6.2: The Simscape Multibody models of the front and rear wheels respectively.

6.1.2 Spring Damping Constants

For modelling accuracy, all constants, from the spring constants to the inertial and mass properties, needed to be determined. For certain properties, like the dimensions, mass, and inertia, the values could be obtained through the SolidWorks model that was discussed in Chapter 5. In the case of the spring constants, experimental data was used to determine the constants. All four tyres as well as the four suspension units required their spring-damper constants to be modelled and were done so as simple spring-damper systems. The spring constant for each component was developed using experimental procedures while the damping constants were set to be proportional to the spring constants. To determine the spring constants, each of the different tyres and suspension units were attached to a boom that could record the vertical displacement of the unit. These components were then put on top of a force plate to measure the vertical forces. Varying forces were then applied by hand to the boom and the displacement was recorded. Due to spring constants having a direct relationship between force and displacement with the equation $F = kx$, the spring constants could be determined by plotting the force against the displacement and finding the slope of the line of best fit. The results of one such test done can be seen in Figure 6.3.

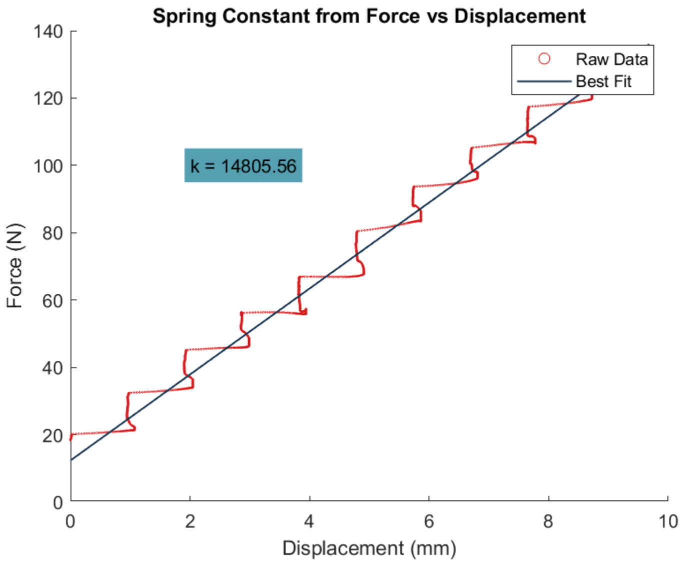


Figure 6.3: An example of the force-displacement curve from an experiment done to determine the spring constant of the back tyres.

After doing multiple tests for the front wheel, the back wheel, the front suspension, and the back suspension, the spring constants for each of these units can be seen in Table 6.1. The damping coefficients used were estimated to be just the spring constants multiplied by a factor of $1e-1$ to ensure that the damping coefficients were proportional to the spring constants. Due to the unknowns of the material and other factors, the damping coefficients were not obtainable. For this reason, the damping constant were assumed to be a factor of the spring constant for simplicity. It was seen later in the modelling that these specific variables did not play a major role in the simulation's outcome. Therefore it was deemed acceptable for the estimated constants to be used.

Table 6.1: Spring and damping constants for the front and back tyres and suspension units.

Unit	Spring Constant	Damping Constant
Front tyres	9869.34	986.93
Back tyres	14805.56	1480.56
Front Suspension	3463.30	346.33
Rear Suspension	3024.78	302.48

6.1.3 Chassis and Gearbox System

The chassis was modelled as a rigid body with the mass and inertia properties determined from the SolidWorks model. Attached to the chassis was an anti-ramp board (ARB) to prevent the car from flipping during accelerating movements. This was modelled to interact with a certain portion of the ground while the car was accelerating, but would allow the car to quickly build to the desired speed without preventing the motion of the turning by accidentally knocking the ground during the manoeuvre. The chassis was the centre frame of the simulation, with 6-axis DOF which allowed the simulation to move in any orientation. Being central to the design, everything was connected to the chassis, including the wheels, the tail, and the drive motor. The chassis model, as can be seen in Figure 6.4, included the ARB, the electronics, batteries and any other part that remained stationary relative to the car. This allowed for simplification of the model and reduced the need for multiple bodies and joints in replace of a single body with a single COM and inertia.

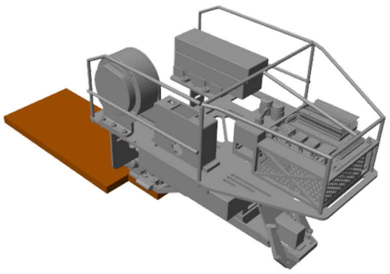
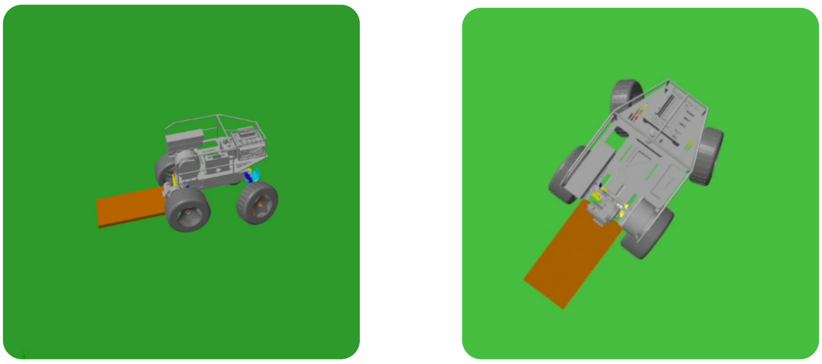


Figure 6.4: The chassis model in the Simscape Multibody simulation.

The differential gearbox, while possible to model with individual gears, was simplified and modelled with a built-in MATLAB block. The block was found in the library within the *Powertrain Blockset*

/ Drivetrain / Final Drive Unit module. This block required the various torque inputs from each wheel and the motor, as well as the inertia values of all the components. Using this block, the gearbox was modelled as a simple differential gearbox with the gear ratio of 1:1. The block also allowed for the simulation of the gearbox to be done in a single step, rather than having to simulate each gear individually. This resulted in the simplification of the simulated model as the relations within the drivetrain were combined rather than having multiple relations. After the drivetrain was modelled, the full car was modelled. With everything, excluding the tail, developed, some simulations were run to verify that things were working as expected. While these were not yet to be compared to experimental data, a simple simulation was run where the car was to accelerate to a speed of 4 m/s while having the steering oscillate between -30 and 30° with a sinusoidal motion. The Figure 6.5 shows an isometric and top-down snapshot of the simulation at a time of 1 second. It was seen in this simulation that everything worked together according to the estimated motions of the simulation.



(a) Isometric view. **(b)** Top-down view.

Figure 6.5: A snapshot of the simulation at a time of 1 second where the tailless model performed a sin wave steering motion while accelerating to a speed of 4 m/s.

6.2 Straight Line Speed Model

The vehicle's speed model needed to be determined. The simplest way to do this was to measure the straight line speed of the actual vehicle and then use those results as a goal for the simulated model. The simulation model would then accelerate to a certain desired speed in as short a period as possible which would allow the simulation to run for shorter periods as it would allow the car to reach its desired speed in 2 seconds compared to 15 seconds of the real model. This was largely why the anti-ramp board was included, to prevent high tipping torques during the accelerating model. The straight line speed model of the car was also needed due to the fact that the car's was controlled through an open loop control system. To control the speed of the *AeroDima*, a throttle command was given which was mapped to a PWM signal which was then used to control the drive motor. The speed of the car was then dependent on the throttle command sent and because of this, a speed model was needed to be able to model the throttle command to the output speed.

The model was obtained through performing a series of experiments where the throttle command was varied, and the speed was measured. This was done for a certain range of throttle commands. Due to the fact that the car, when flipping without the tail, did not need to go at full speed to flip, a throttle range between 30% and 60% was tested. The reason for starting at 30% was due to the load torque of the motor being too high for the motor to overcome at lower throttle commands. The steps used in the straight line experiments can be seen below and the setup of *AeroDima* on the field can be seen in Figure 6.6. In between every couple of tests, the batteries were recharged to ensure a sufficient load was able to be outputted.

1 Set up *AeroDima*, without the tail, on the field by powering the platform, connecting to it, and starting the programs

2 Start the logging of the sensors.

3 Set a throttle command of 30% and wait for the car to get to steady-state.

4 Apply a 0% throttle command and stop logging.

5 Repeat steps 2-4 two more times.

6 Repeat steps 2-5 while incrementing the throttle command by 10% every three times.

7 This will result in a total of 12 tests, with three tests per throttle command.

(a) Isometric view. **(b)** Side view.

Figure 6.6: The setup of *AeroDima* on the field for the straight line speed model experiments. The tail was not included for this experiment

After the experiments were concluded, the results were then analysed. Due to the purpose of this experiment, only the GPS data was needed. which was then used to determine the speed of the car. The speed profiles were then plotted to determine the maximum speeds the car reached and can be seen in Figure 6.7.

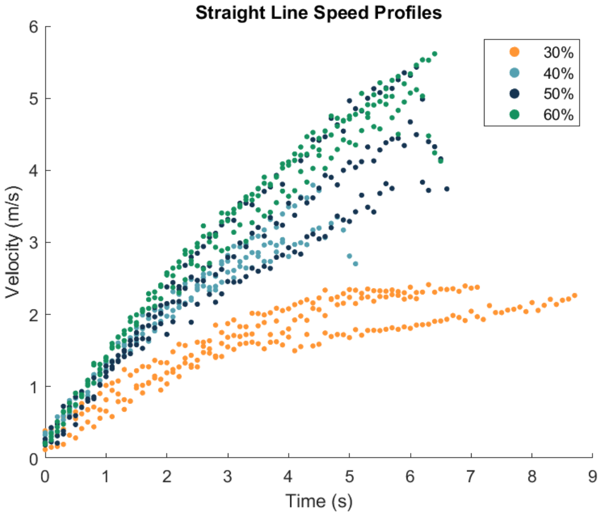


Figure 6.7: The speed profiles of the straight line speed model experiments till a steady-state was achieved.

Using these profiles, the maximum speeds were obtained and plotted against their throttle commands. Using a line of best fit, the linear speed model was calculated and was plotted, as can be seen in Figure 6.8. This model was used to determine the throttle command needed to achieve a certain speed.

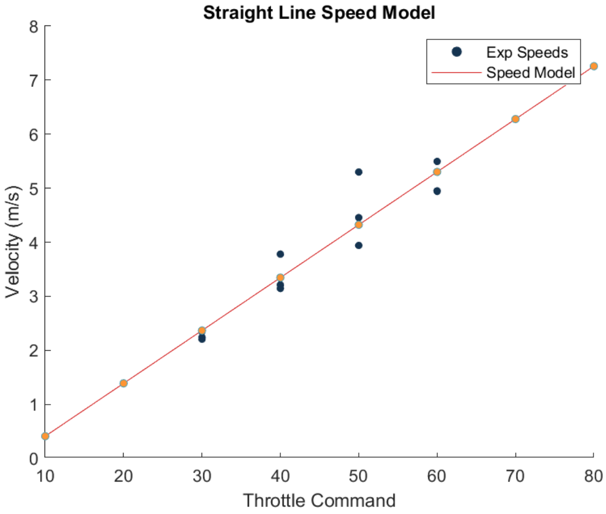


Figure 6.8: The speed model of *AeroDima* from the straight line speed experiments.

6.3 Tail Model Development

The tail model's development was different to the previous model development. The difference mainly involved the aerodynamic modelling of the tail's shape as, due to the importance of the tail in the overall system, an accurate model was required. This was done through the development of the tail in Simscape while comparing it to an experiment done in the wind tunnel. While the tail was designed and built, as described in Section 5.2, and theoretical aerodynamic coefficients were found, experimental data was deemed necessary to validate or even update this model. This was done in two parts: the first being a wind tunnel experimentation and data collection, and the second being the simulated model development and analysis.

6.3.1 Wind Tunnel Experiment Methodology

The experiment was done in a similar vein as the experiment done in Chapter 4, however, with two differences. One being that the developed robotic tail was obviously used in place of the cheetah tail and the second development was the rig that had to be adjusted to fit the tail and motors. This adjustment included changing the mounting structure, but it also included changing the actuating motor from the Maxon motor to the AK70-10 motor which allowed for a simple integration with the communications protocol developed in Section 5.4.5.

Figure 6.9: The wind tunnel rig with the tail attached. The tail is in a position such that sail end is in the wind.

The method of experimentation involved the tail being set up to be swung into the wind in the pitch axis. The initial angle of the tail was set that it was fully in the wind which occurred at angles of about 50°. The tail, starting from a vertical position out of the wind to account for any biases, was swung until it was stationed in the wind. Once it was in position, it was swung to a commanded value of 0°, a vertical down. This was done throughout six different scenarios where the wind speed was altered to be 5, 7.5, and 10 m/s and two different speeds were commanded per wind speed. Due to the internal controller of the motors, a position command was sent and by varying the controller gains, the speed at which the tail swung to get to that position differed. These two different controller gains were a Kp of 4 and 10. With five tests being done for each of these scenarios, a total of 30 tests were done. The method of logging the data is the same as the previous experiment, with the data being logged at 1000 Hz through the use of the Speedgoat system. The force data was still measured through the use of a 6-axis load cell, the Axia80-M20 load cell. The data was then processed using MATLAB's built-in filtering functions. Using a 6 order Butterworth filter with a cutoff frequency of 20 Hz, the data was prepared to be used for further analysis.

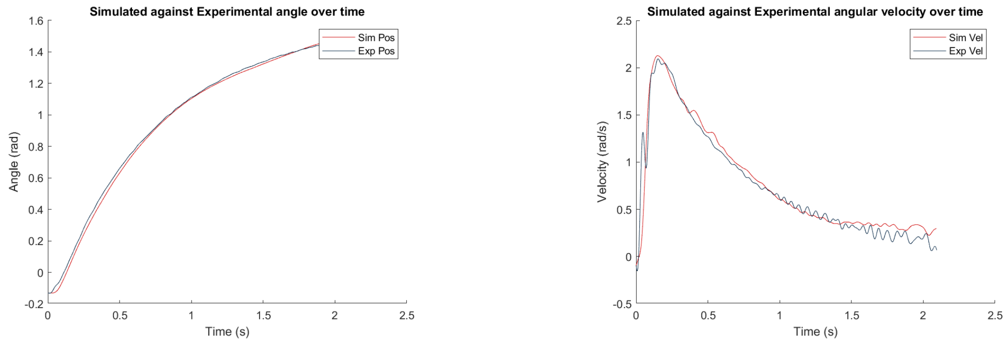
6.3.2 Simscape Tail Model Development

The cheetah tail analysis used Euler-Lagrangian dynamics to simulate and analyse the model, however, for this model, a Simscape model was developed instead which allowed for the model to be easily integrated with the rest of the model. The multibody components of this model were more complicated than the chassis' body: the AK70-10 motor was modelled to include the individual gearing within its built-in gearbox, the tail and shaft was modelled as a single part, while the T-joint connector was a third part. Each of these parts had joints which allowed them to move in the correct orientation. The motor's joint also had an input torque which was used to control the motor and the input torque would be obtained through the current logged in the experiment. The last, if not most crucial, component was the aerodynamic force applied to the tail. This was simplified to be a single point to reduce computing complexity. The force was realised through the use of an *External Force and Torque* block. This block allows for a force vector to be applied to a specific point with a specific orientation. For this model, an attached frame reference was used, so to ensure the correct orientation of the forces throughout the simulation.

Before the aerodynamic analysis, the friction experienced by the motor first needed to be isolated and then modelled and this was done using a simple coulomb-viscous model. To determine the friction values, the tail was attached to the AK70-10 and the rig was operated in a way that meant that the aerodynamic forces were small enough to be presumed negligible. This was done by operating in a lab with no motion of the ambient air and the tail was also made to move at a speed that was low enough that resulted in low aerodynamic forces. While aerodynamic forces are present in all objects situated within fluids, in this scenario the forces were low enough to be assumed to be zero. The tail was then situated at a vertical down and for five tests, it was made to swing to a commanded position of 90°. The position, velocity, and current values for each swing were then logged and saved. In the Simscape model, the same motion was commanded, using the current input. The Simscape model used the built-in ode15s solver which was automatically chosen by MATLAB to be the most efficient solver for the situation. For each test, the friction coefficients of the simulation were then manually changed to produce the same results as the experiment. The frictional force was modelled to be a simple coulomb-viscous model as spoken about in Section 4.7. This resulted in an equation as seen in Equation 6.1, where C_f is the coulomb friction coefficient and V_f is the viscous friction coefficient. The friction was not modelled with a tanh function for continuity as it was seen that Simscape Multibody could handle the discontinuity well enough.

$$\tau_{friction} = sign(\omega_{tail})(C_f + V_f|\omega_{tail}|) \tag{6.1}$$

While an optimisation function could have been used, due to the duration it took to perform each simulation, this was not a time feasible option. For this reason, the friction coefficients were manually changed until the position and velocity profiles of the simulated and real experiments aligned. The results of the profiles of one of these tests can be seen in Figure 6.10. The friction coefficients were then combined and used a weighted average method, according to the resnorm of the position profiles, to determine the final used coefficients as mentioned in Equation 4.31. These coefficients that were used can be seen in Table 6.2.



(a) Position Profiles.

(b) Velocity Profiles.

Figure 6.10: The position and velocity profiles of the experiment and simulation of the friction test.

Table 6.2: The coulomb and viscous friction coefficients found and used in the Simscape tail model.

Variables	**Swing 1**	**Swing 2**	**Swing 3**	**Swing 4**	**Swing 5**	**Final Values**
Cf	0.165	0.160	0.160	0.160	0.160	**0.161**
Vf	0.060	0.055	0.055	0.055	0.060	**0.057**
Resnorm	5.05E4	5.51E4	4.90E4	6.15E4	6.14E4	*N/A*

6.3.3 Simscape Tail Model Analysis

The aerodynamic coefficients were next be determined and this was done by using the Simscape model and comparing the results of the simulation to the results from the experiment done in 6.3.1. The simulation was run with the same wind speeds and used the measured current as an input, and the aerodynamic forces used the same model as Equation 4.18. With these three force inputs; motor torque, friction, and aerodynamic force, the model was ready to be run. To get an idea of what the model looked like, Figure 6.11 shows the model with the tail in a horizontal position. While the tail was not positioned like this in the simulation, it was positioned this way for illustrative purposes.

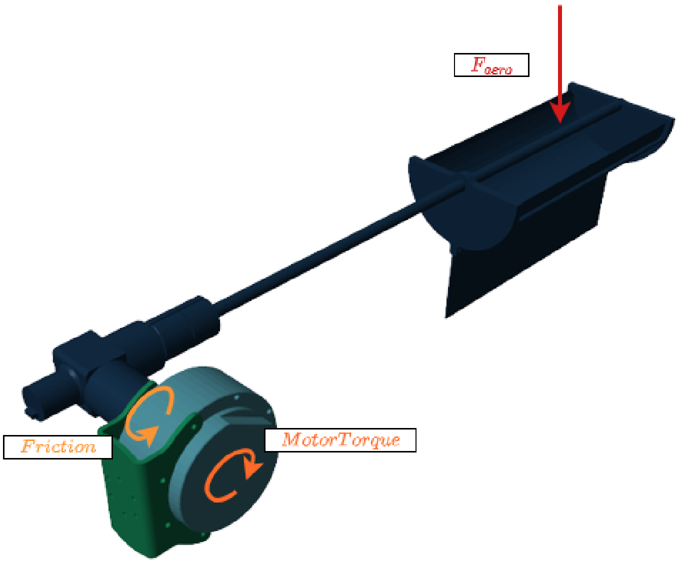
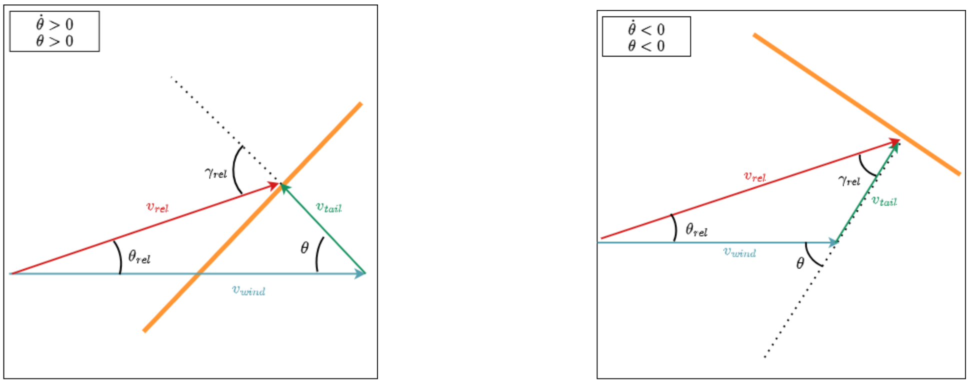


Figure 6.11: The Simscape Multibody model of the tail with an illustration of the three force and torque inputs.

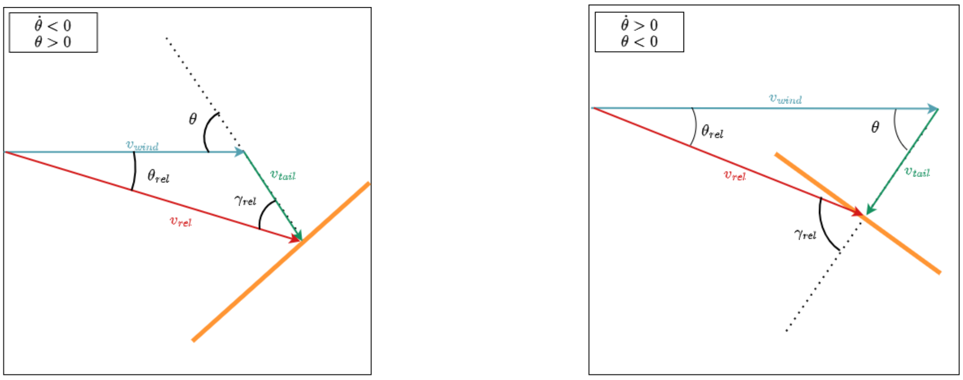
These external forces can be modelled through mathematical equations. The motor torque is simply the current multiplied by the torque constant of the motor as seen in Equation 6.2, where the torque constant, K_t, is 0.123, and where the current is taken from the measured current from the experiment.

$$T_{motor} = K_t I_{motor} \tag{6.2}$$

The aerodynamic force vector positioning was different from the Euler-Lagrangian model. While the force has the same principles as Equation 4.18, due to the nature of the Simscape external forces, the reference angle needed to be calculated separately. To analyse the reference angle, four scenarios needed to be considered. The differences in the scenarios were when the tail was either at a positive or a negative angle, relative to vertical down, as well as when the tail's velocity was either positive or negative and using these two differences combined gave four scenarios. These four scenarios are represented in Figure 6.12 where the orange bar represents the tail, the dotted black line being perpendicular to the tail. Included in the figure are the various angles and velocities that are needed to calculate the reference angle.



(a) Scenario 1. **(b)** Scenario 2.



(c) Scenario 3. **(d)** Scenario 4.

Figure 6.12: The four scenarios that need to be considered when calculating the reference angle with their relative scenario identifiers of tail angle and velocity.

Analysing these four scenarios would result in a model which could be used in all the scenarios. Firstly, the tail velocity, v_{tail} needed to be calculated by multiplying the radial velocity of the tail (ω_{tail}) by the distance of the point to the center of rotation (L_{cor}). This equation can be seen below in Equation 6.3.

$$v_{tail} = L_{cor}\omega_{tail} \tag{6.3}$$

Following that, the relative wind velocity, v_{rel}, could be calculated and split into its individual x, y parts and can be seen in Equations 6.4 and 6.5.

$$v_{rel,x} = v_{wind} + v_{tail}cos(\theta) \tag{6.4}$$

$$v_{rel,y} = v_{tail}sin(\theta) \tag{6.5}$$

Once the relative wind velocity was determined, the angle that the wind made relative to the horizontal could be found. This angle, θ_{rel}, can be seen in Equation 6.6. Using this relative angle and the different aspects of the four scenarios, the angle of attack, γ_{rel}, could be determined. This can be seen in Equation 6.7.

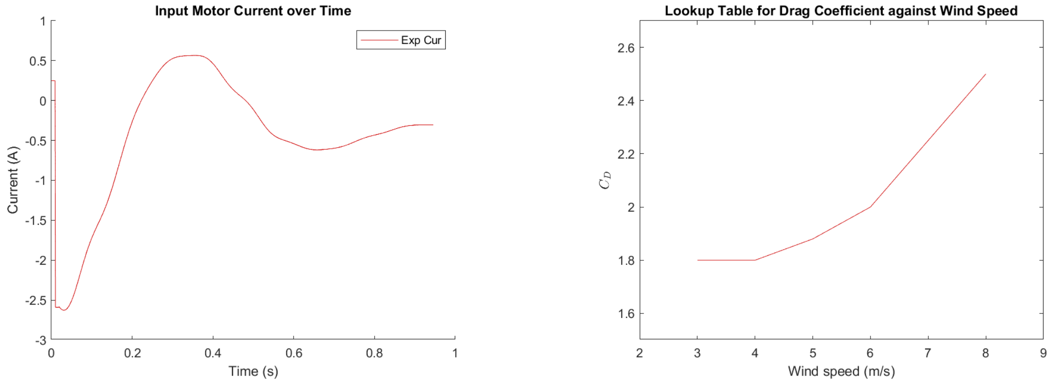
$$\theta_{rel} = atan(v_{rel,y}/v_{rel,x}) \tag{6.6}$$

$$\gamma_{rel} = |\theta_{tail}| - sign(\omega_{tail})|\theta_{rel}| \tag{6.7}$$

This angle of attack is then used in determining the final aerodynamic force, F_D, as seen in Equation 6.8. The force is then applied perpendicularly to the tail. The lift force can also be calculated in this manner. However, for this simulation the lift force was not considered due to it being coincidental with the center of rotation of the tail which means that the lift force would not have any effect on the tail's motion.

$$F_D = \frac{1}{2}\rho C_D(v_{rel})Acos(\gamma_{rel})v_{rel}^2 \tag{6.8}$$

The drag coefficient in this final equation is dependent on the relative velocity. This was obtained, in Simscape, through a lookup table. Due to the nature of the experiment, the swings occur at various different relative wind velocities. As varying velocities result in varying drag coefficients, a lookup table was needed that checked the relative wind velocity and applied a certain drag coefficient as an output. This lookup table, throughout the various simulations, was adjusted by hand to produce the best results. and while these results may not have been optimised per se, a full optimisation run was not viable to use due to the computing complexity and time constraints that arises when using any optimisation function within a Simscape model. Each swing operated within a range of relative velocities and so the drag coefficients for each relative wind velocity were manually changed until the position and velocity profiles closely lines up for each of the swings performed. An example of a drag coefficient lookup table can be seen in Figure 6.13. This figure shows the motor current input and the drag coefficients used for a wind tunnel test at 5 m/s with a motor Kp gain of 4.

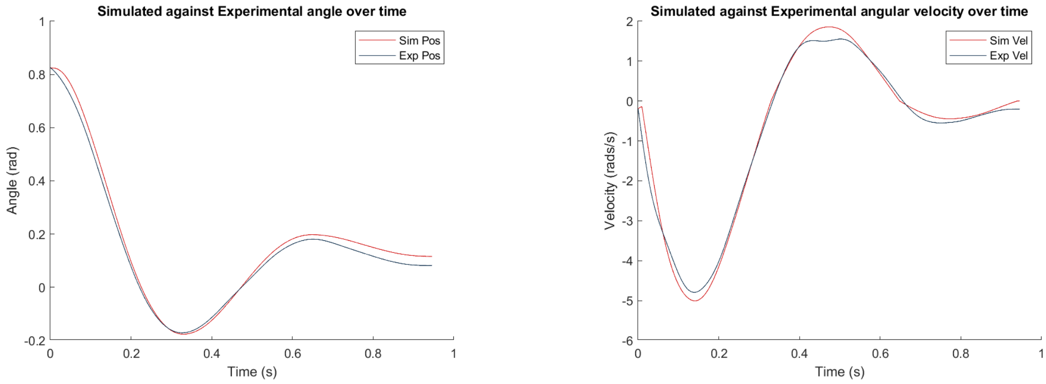


(a) Motor Current **(b)** Drag Coefficients

Figure 6.13: (a) is the motor current from the experiment that is fed into the simulation. (b) is the drag coefficients used, obtained from a lookup table with the relative wind velocity as the reference values. This experiment is an example from a wind tunnel test at 5 m/s with a Kp gain of 4.

Using this current input and drag coefficients profile, Figure 6.14 shows the simulated and experimental position and velocity profiles for one experiment at 5 m/s wind speed with a Kp gain of 4. Each of these simulations resulted in an error term between the experimental data and the simulated data. The sum of each of these error terms (θ_{exp}, θ_{sim}) was used as the resnorm for each test as can be seen in Equation 6.9. The resnorm was used to determine the accuracy of the model for when choosing the final model to use for the simulation.

$$resnorm = \sum_{i=1}^{n}(\theta_{exp,i} - \theta_{sim,i}) \tag{6.9}$$



(a) Position Profiles.

(b) Velocity Profiles.

Figure 6.14: The position and velocity profiles of the experiment and simulation of a wind tunnel test at 5 m/s with a Kp gain of 4.

Each swing was performed five times for six different scenarios. Using these manually adjusted drag coefficients, a scatter plot for the full experiment was plotted to determine the model used for the final full simulation. Using these data points, three potential models were identified. The first potential model was a weighted average model, weighted by the inverse of the resnorm for each test. This provided a decently accurate model but fluctuated quite significantly as the relative wind speed increased. The second potential model was a linear line of best fit. This simplified the model, however was not as accurate as the weighted average model and was just about as complicated to implement in the simulation. The final model option was a much simpler model that was one of a single drag coefficient. This was not as accurate as the other two models at higher speeds but this model showed more accuracy around lower relative wind speeds and became less accurate towards higher relative wind speeds. These three models, as well as the data points, can be seen in Figure 6.15. The full set of these results can be found in Appendix 11.2 but an example of one of the tests done can be seen in Table 6.3.

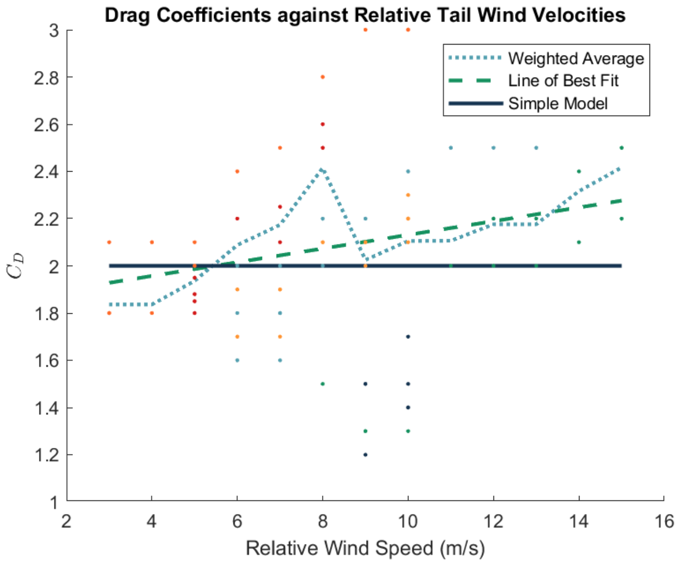


Figure 6.15: A scatter plot of the drag coefficients obtained from the various simulation and the experiment. The plotted lines are three different models: weighted average, linear regression, and straight line.

Table 6.3: Results from the simulated tail model for a wind speed of 5 m/s with a Kp gain of 4.

Relative Velocity (m/s)	Test Conditions: Wind Speed = $5m/s$, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
3	1.80	1.80	1.80	1.80	1.80
4	1.80	1.80	1.80	1.80	1.80
5	1.88	1.80	1.95	1.95	1.85
6	2.00	2.00	2.20	2.20	2.00
7	2.25	2.25	2.25	2.25	2.10
8	2.50	2.60	2.60	2.60	2.60
Resnorm	11.15	17.34	9.04	4.01	13.66

Looking at the analysis of the dynamic cheetah tail model in Section 4.9, it was seen that the model was also more accurate at lower wind speeds due to the Reynolds Number. For the combination of this reason, and the simplicity of the computational aspect, the single coefficient model was chosen. The above figure shows the chosen drag coefficient to be 2. Comparing this model to the values obtained through the SolidWorks flow simulation, as described in Section 5.2, this value was, on average, higher. Those tests were also done for wind speeds of 10, 20, and 30 m/s however the SolidWorks CFD analysis did not take into account the motion of the tail and so those results represent a static analysis. These results represent a dynamic analysis of the tail. In the CFD model, the drag coefficients could achieve a value of over 2 depending on certain factors which shows that the values obtained through the Simscape model could align with the CFD results under certain conditions. This occurred at an angle of attack of 45°, however due to the nature of the Simscape model, only relative wind velocity was used as a dependent variable and not the angle of attack when determining the drag coefficient. This was done to determine a simple model that was acceptable enough for it to be used across all states and not over complicate the model. It was therefore determined that the drag coefficient of 2 was a good enough value to use for the simulation.

6.3.4 Tail Control System

Lastly, the control system needed to be designed and while many control designs could be used, the scope of this research did not include an optimal control design. Therefore, a basic, non optimised system was used. Firstly, it was necessary to determine what was required of the control system. The tail needed to be held in a steady position while the car was driving. When the car turned, this would trigger the tail which would start swinging. Once the tail hit a certain angle, the tail would decelerate. The tail also had a limited range of motion due to the constraints of the tail hitting the ground at the end of the swing. To provide some safety in case of overshoot, the limits were set as $\theta \in [90; -90]$ where 0° was vertically upright. Because of that, the tail could only travel a certain distance before requiring to slow down. Another thing to keep in mind for the control system was the tail's torques needed to counter the centrifugal torque experienced by the car and not add to it - causing the car to flip further. This was because the aim of the tail was to increase the manoeuvrability and stability of the car, and not aid it in flipping at high speeds. Even so, this torque controlled aspect was not incorporated into the control design due to the simple nature of the control system. For future work, this should be a consideration to take into account.

The control system ended up resulting in a state based design structure with two states: a position, and a speed state. Once the car's control scheme started, the tail would initially move to its horizontal position after which the tail would then be held in a steady position for an indefinite length of time. Once the car started moving, the tail would wait until a trigger occurred to start its swing. At this point, the tail would accelerate to a certain speed and once the tail reached a commanded angle, the tail would slow down and move to the end of its motion, the opposite horizontal position. To visualise this, a speed region and profile was created for illustrative purposes and can be seen in Figure 6.16. In this figure, the green shaded zone is the region where the tail's inertia is stabilising the car, while the red shaded zone is where the inertia is destabilising the car. A depiction of the speed profile is shown by the teal line. This line shows the speed of the tail through the swing motion where it starts at rest and accelerates quickly to its maximum speed. Once it has reached the commanded angle limit, the tail will slow down and slowly come to the end of its swing. The position limits are shown as well.

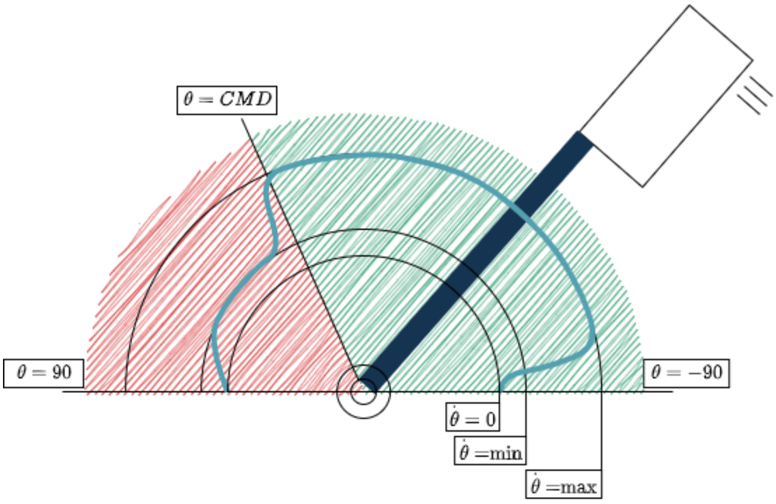


Figure 6.16: A state flow diagram of the tail's speed profile. The green shaded region is where the tail is stabilising the car, while the red shaded region is where the tail is destabilising the car. The teal line is the speed profile of the tail.

There were multiple triggers than could have been utilised to start the swing. One trigger could have been an open loop trigger of a step command which would mean that when the servo was sent a command to turn the car, the tail would be triggered after a certain delay period. This was not sufficient for the desired purposes of the control loop due to the inherent limitations of not accounting for any other aspect of the car other than the step steering. The next trigger that could have been used was a linear acceleration trigger, although this was put in contention with a yaw rate trigger. Both of these triggers would mean that the tail would only start swinging after the car had reached a certain tipping point. The tipping point, in a simple physics model, can be determined by equations of motion, but for this car model, the equations of motion of a simple model proved invalid to use due to the calculated numbers being too high for what the model experienced. This was seen in the Simscape model when doing testing and so the equations of motion of the system when treating it like a box would not work for this purpose. Ultimately the yaw rate was used as a trigger point and the trigger value for this was obtained through the Simscape model. The wheel contact points were plotted against the yaw rate of the car, and it was seen that the inside wheels, in a turn, maintained contact with the ground until the yaw rate reached a value of 1 rad/s. At this point, the inside wheels lifted off the ground and so it was this point that was chosen to start the tail swing.

The simple controller that ended up being chosen and designed was a state based controller. The output of this controller sent commands to the roll motor depending on the trigger, the tail states, as well as the current state of the controller. As the controller was not dependent on other sensors apart from the roll motor's encoder, the operating control frequency could be implemented at a frequency of 100 Hz. An overview of the controller can be seen in Figure 6.17. This controller required feedback from the yaw rate, the tail state, and the controller state and then outputted a motor command that the motor understood, as discussed in Section 5.4.5.

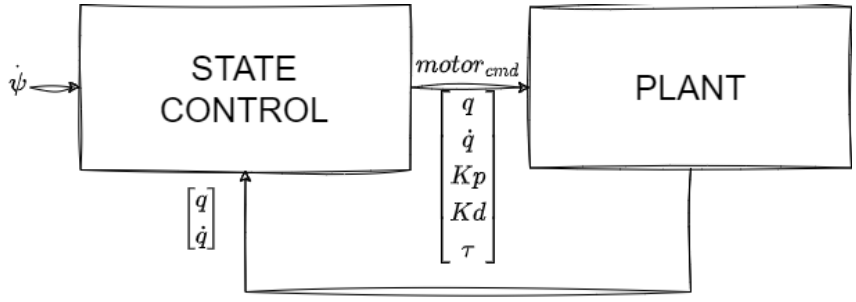


Figure 6.17: An overview of the state based controller design.

After the design of the controller was chosen, the implementation of the design in the model was done using MATLAB's *Stateflow* toolbox. The two states modelled were the position and speed state and the *Stateflow* chart can be seen in Figure 6.18. The position state allowed the controller to either maintain a position or slowly get to the desired position. This was achieved by having Kp and Kd gains of $[17; 5]$ for the maintaining state, and Kp and Kd gains of $[5, 2]$ for the slow motion. After the tail was triggered by a yaw rate of 1 rad/s, the speed controller started the motion of the swing. The chart logic is explained as follows. The tail would stay at a horizontal 90° by having Kp and Kd gains of $[17; 5]$. The tail would remain at that state until the tail was triggered by a yaw rate of 1 rad/s. Following which the tail would start its swing, getting to a speed of 8 rad/s by having a Kp and Kd gain of $[0; 3]$. The commanded tail speed could be changed depending on the simulation run. It would remain in this state until the tail reached a certain commanded position. This was chosen to be 130° as it allowed the tail enough time to slow stop and stop at the opposite horizontal. After the tail reached the 130°, it would then revert to a position controller with the reduced gains to allow for the tail to get to the horizontal at a slower speed. To see the specifics of the motor command from the controller for each state, see the Table 6.4.

Table 6.4: Motor Commands for the various states of the tail motor controller.

Motor Command	**Position State 1**	**Delay State**	**Velocity State**	**Position State 2**
$\begin{bmatrix} q \\ \dot{q} \\ Kp \\ Kd \\ \tau \end{bmatrix}$	$\begin{bmatrix} 90 \\ 0 \\ 17 \\ 5 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 90 \\ 0 \\ 17 \\ 5 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 8 \\ 0 \\ 3 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -90 \\ 0 \\ 5 \\ 2 \\ 0 \end{bmatrix}$

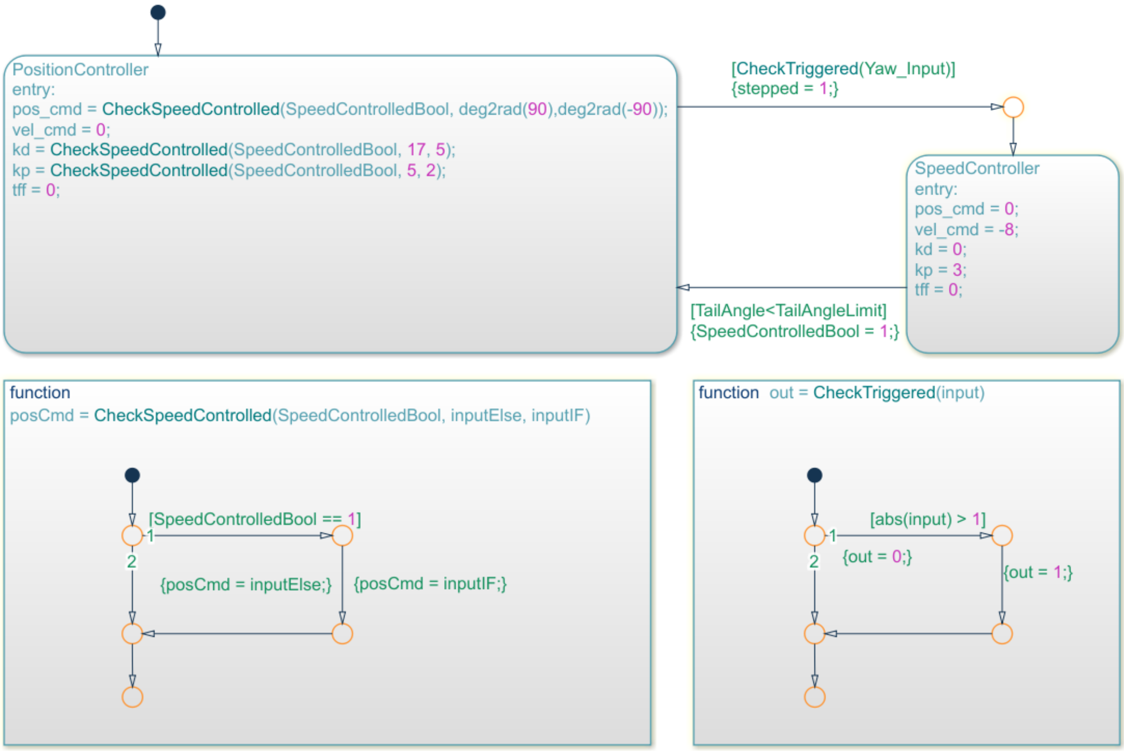


Figure 6.18: The speed controller used in the Simscape model, designed with the Simscape Stateflow tool.

6.4 Summary

In summation, this chapter looked at the Simscape Multibody modelling of *AeroDima*. This modelling was performed that various control structures could be identified and tested before trying them on the physical *AeroDima*. This modelling was done by importing the .STL files from the SolidWorks model and creating the skeleton of the car. From there, the constants needed were determined, either through the SolidWorks model or through experimental testing. The tail model was then looked at. This looked at the multibody aspect of the model and the aerodynamic force aspect. The aerodynamic model was determined by wind tunnel tests of the robotic tail and comparing the data obtained from the experiments with a simulated model of the tail - the final model was chosen to be a single number drag coefficient model. The tail control system was then designed to be a state based controller. The two states were a position and speed state which allowed the tail to maintain its position and swing at a certain speed after a yaw rate trigger was activated. With all this modelled, the full Simscape simulation of *AeroDima* was modelled and could be used for testing of the performance of the tail on the car. These testing sequences are discussed in the next chapter.

Chapter 7: Experimental Testing of AeroDima

In the previous chapters the design and development of *AeroDima* has been discussed. This chapter will therefore discuss the experimental testing of the car. This chapter will be split into two sections; the first will perform and discuss the simulation experiments, while the second will perform and discuss the physical experiments. These experiments were performed to test the manoeuvrability of the car and were performed with and without the tail to see the effect of the tail. The simulations were used as a base mark and as a validity testing method while the physical experiments were used to test the validity of the simulation and to see how the car performs in real life. The results and analysis of these experiments will also be discussed in this chapter.

7.1 Simulation Experiments

The simulation was developed to be able to test the performance of *AeroDima* without potential damaged done to the platform. To measure the performance of the car, two configurations would be tested, namely being the car without the tail, and the car with the tail. This was to see the difference that the tail makes to the performance of the car and so before these configurations were tested on *AeroDima*, they were simulated using the Simscape model of the platform developed in Section 6.

Before the simulations could be run, a reference frame from which the data of both the simulation and real car could rely on was set. This reference frame was set to be the same as the reference frame used by the IMU, which differed from the Simulation. All simulation data recorded was then rearranged and put into the correct frame of reference. This frame of reference can be seen in Figure 7.1. All rotation based measurements were recorded in the same 3 axes but converted to a rotation base using the right-hand rule where, when holding up one's thumb, the direction of the remainder of the fingers indicate the direction of the rotation frame.

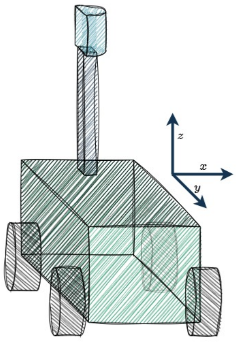


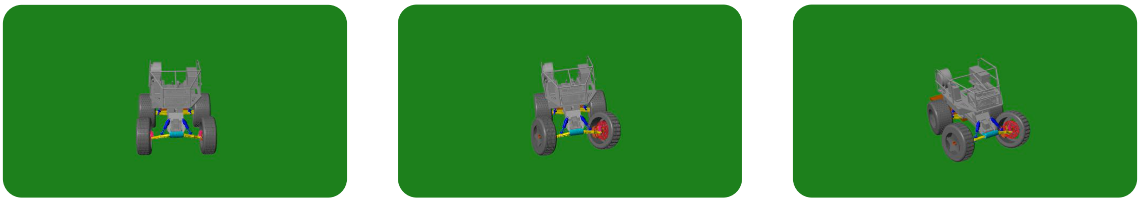
Figure 7.1: The frame of reference in relation to the car.

7.1.1 Simulated Tailless Manoeuvre Tests

The first simulated configuration tested was the simulated tailless manoeuvre. In this section, the model was run without the tail while set to drive to a certain speed and then perform a turning motion. The turning motion was a 30° wheel turn that was performed for a period of 0.5 s. When looking at the physical car's steering motion, it was timed that the servo motor took 0.14 s to turn the wheels from 0° to 30°. This meant that after the 0.14 s delay, the servo motor had completed its step motion. At the end of the 0.5 s period, the servo would then turn back to a forward position. This also took 0.14 s which resulted in the total steering motion from the initial step to the wheels pointing forward again taking 0.64 s. This was the steering motion that was used in the simulation. The car was set to get to the correct speed within a 2 s interval, and after 2 s, the steering motion would be applied. For these tests, the commanded speed was ramped up in 0.28 m/s increments from 0.56 m/s until the car fell over to see the speed at which the car flipped. The metrics obtained at the speed just before flipping would then be used to be compared to the tailed version and the physical versions. The steps taken in this sequence can be seen below.

1 Set up the simulation model without the tail.
2 Set the commanded speed to 0.56 m/s.
3 Wait for 2 s to allow the car to get to the correct speed.
4 Apply a 30° turn to the servo motor.
5 Wait for 0.5 s before straightening the wheels.
6 Log the data at 1000 Hz.
7 Repeat steps 2 to 6 until the car flips after increasing the commanded speed by 0.28 m/s.

With the process laid out, then simulations were run. It was seen that the car could handle turns up to a speed of 3.61 m/s before it flipped over. Frames of the simulation's successful turn occurring at a speed of 2.78 m/s can be seen in Figure 7.2. These frames are the before, during, and after snapshots of a successful turning manoeuvre.

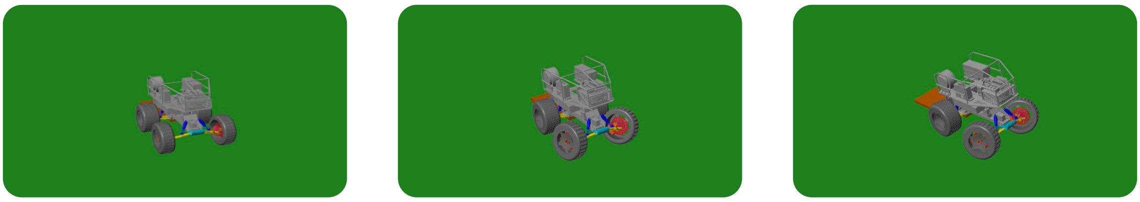


(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.

(d) Frame 4. **(e)** Frame 5.

Figure 7.2: 5 frames showing the before, during, and after a successful simulation of a 30° turn at a speed of 2.78 m/s.

The snapshots of an unsuccessful turn where the car flipped over, occurring at 3.61 m/s, can be seen in Figure 7.3.

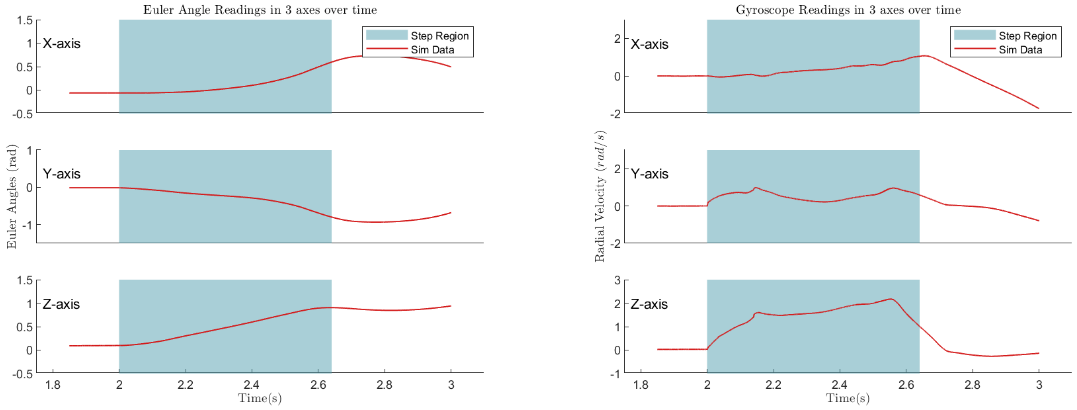


(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.

(d) Frame 4. **(e)** Frame 5.

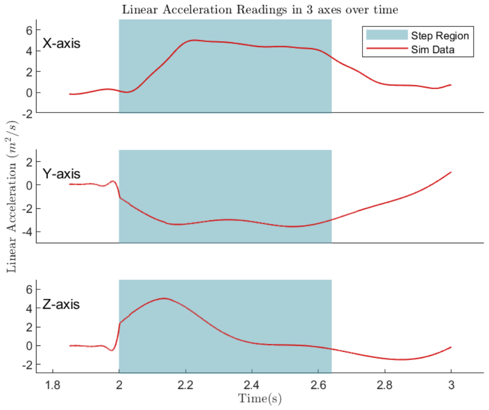
Figure 7.3: 5 frames showing the before, during, and after an unsuccessful simulation of a 30° turn at a speed of 3.61 m/s.

Each simulation took a period of 4 s. During this period, the data was logged and the profiles for each simulation could be looked at and used to determine the metrics of the manoeuvrability of the car. A sample of the data logged can be seen in Figure 7.4. These profiles look at the Euler angles, the gyroscopic readings, and the linear acceleration readings of the car. The profiles were shortened to a time frame that just looked at the turning motion of the simulation with a small period both before and after the turning motion. The turning motion duration can be seen in the shaded region of the plots.



(a) Euler Angles.

(b) Gyroscope Readings.



(c) Linear Acceleration.

Figure 7.4: The data logged from the tailless manoeuvre experiment at a speed of 3.33 m/s The IMU data (subfigures (a), (b), and (c)) was logged at 1000 Hz. These plots included the Euler angles, the gyroscope readings, and the linear acceleration in the X, Y, and Z axes.

After these profiles were obtained, the seven metrics looked at could be calculated. These metrics were the maximum yaw rate, the average yaw rate, the maximum linear acceleration, the average linear acceleration, the maximum Kohut, the average Kohut, and the integral of the roll rate. The reason for these choices was discussed in Section 3.3. Each metric, barring the integral of the roll rate, was separated into the average and the maximum. They were separated because the average metric could be used to determine the overall performance of the car while the maximum metric could be used to determine the peak performance of the car during turning manoeuvres. They were also separate as the literature introducing the metrics was rather ambiguous and did not specify whether an instantaneous maximum or an average metric was used. To calculate these metrics, the period during which the turning motion occurred was looked at where, for the simulation, this happened between 2 s and 2.64 s. The maximum metrics looked at the maximum absolute value of each data set during this period, while the average metrics looked at the average value of each data set during this period. This allowed performance over the duration of the turn and instantaneous performance to be measured. The equations used to calculate these metrics can be seen below in Equations 7.1 to 7.7. For the average metrics, the sum of the values between the sample time $t \in [2 : 2.64]$ and were divided by the number of samples taken during the turning period, which in this case was 640. The velocity used for the Kohut metrics were taken at the initial speed of the turning motion, at $v(2)$, which was done as the simulation's forward velocity was not constant during the turning motion. While it hardly varied, it was decided to use the initial velocity to ensure that the metrics were calculated correctly.

In the Equations 7.1 to 7.7, a and its variants refer to the linear acceleration, $\dot{\theta}$ and its variants refer to the yaw rate, v_{car} refers to the forward velocity of the vehicle, K and its variants refer to the Kohut manoeuvrability metric, while the ϕ and its variants refer to the roll rate of the vehicle. The first two equations below are means of calculating the maximum and average linear acceleration of the vehicle.

$$a_{max} = max(abs([a(2), ..., a(2.64)])) \tag{7.1}$$

$$a_{ave} = \frac{\sum_{i=2}^{2.64} a(i)}{640} \tag{7.2}$$

The next two equations are means of calculating the maximum and average yaw rate of the vehicle.

$$\dot{\theta}_{max} = max(abs([\dot{\theta}(2), ..., \dot{\theta}(2.64)])) \tag{7.3}$$

$$\dot{\theta}_{ave} = \frac{\sum_{i=2}^{2.64} \dot{\theta}(i)}{640} \tag{7.4}$$

The next two equations are means of calculating the maximum and average Kohut manoeuvrability metric of the vehicle.

$$K_{max} = max(abs([v_{car}\dot{\theta}(2), ..., v_{car}\dot{\theta}(2.64)])) \tag{7.5}$$

$$K_{ave} = \frac{\sum_{i=2}^{2.64} v_{car}\dot{\theta}(i)}{640} \tag{7.6}$$

The next equation is a way of calculating the roll of the vehicle by integrating the roll rate of the vehicle over 4 s.

$$\phi_{roll} = max(\int_0^4 \dot{\phi}(t)dt) \tag{7.7}$$

The metrics calculated, taken from the fastest tailless run at 3.33 m/s can be seen in Table 7.1. The metrics were calculated for each simulation and then plotted against the speed at which the simulation was run. This will be shown and discussed in Section 7.1.3.

Table 7.1: The metrics obtained from the tailless manoeuvre experiment at a speed of 3.33 m/s.

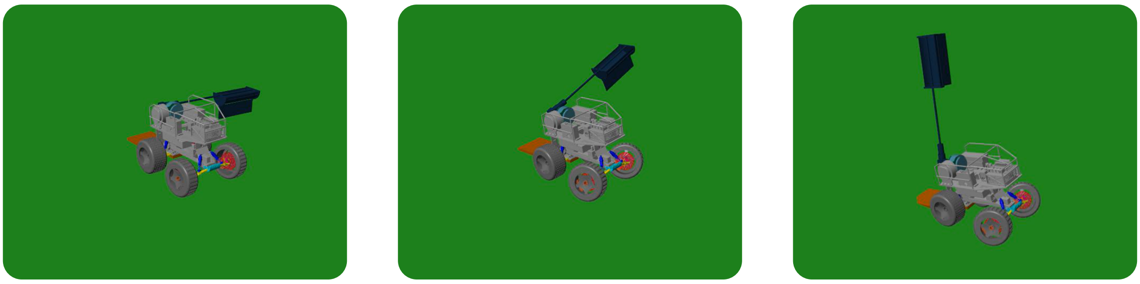
Speed (m/s)	**Max Yaw Rate** (rad/s)	**Average Yaw Rate** (rad/s)	**Max Lin Acc** (m/s^2)	**Average Lin Acc** (m/s^2)	**Max Kohut** ($\frac{\dot{\theta}m}{s^2}$)	**Average Kohut** ($\frac{\dot{\theta}m}{s^2}$)	**Integral of Roll Rate** (rad)
3.30	2.17	1.46	5.01	3.70	413.31	276.06	932.43

7.1.2 Simulated Tailed Manoeuvre Tests

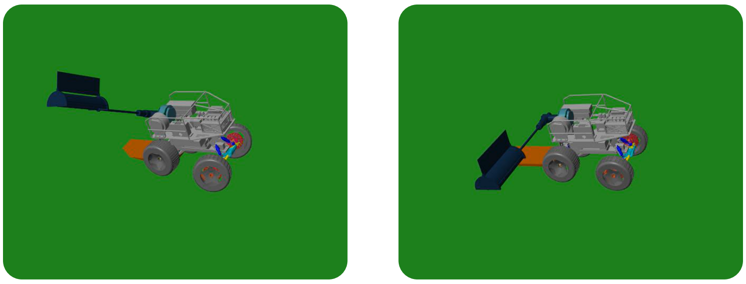
With the tailless simulations completed, the tailed simulations were next to be performed. These simulations were performed differently to the tailless simulations as the tail was then added to the simulation and the tail speed was set to be a variable to adjustable in the varying simulations. Due to the control of the tail setting the tail's speed to a constant speed, varying tail speeds were used to see what the difference in tail speeds resulted in. Other than that, the simulations were performed in the same way and the steps taken in this sequence can be seen below.

1	Set up the simulation model with the tail.
2	Set the commanded speed to 3.33 m/s.
3	Set the tail speed to 2 rad/s.
4	Wait for 2 s to allow the car to get to the correct speed.
5	Apply a 30° turn to the servo motor.
6	Wait for 0.5 s before straightening the wheels.
7	Log the data at 1000 Hz.
8	Repeat steps 3 to 6 while incrementing the tail speed by 2 rad/s until the speed was 14 rad/s.
9	Repeat steps 2 to 8 until the car flips after increasing the commanded speed by 0.28 m/s.

It was seen that the simulation flipped at a speed of 4.72 m/s, and so the simulation ran from 3.33 m/s to 4.44 m/s with the tail changing between 2 rad/s and 14 rad/s. Snapshots of the turning motion of a tailed simulation can be seen in the Figure 7.5



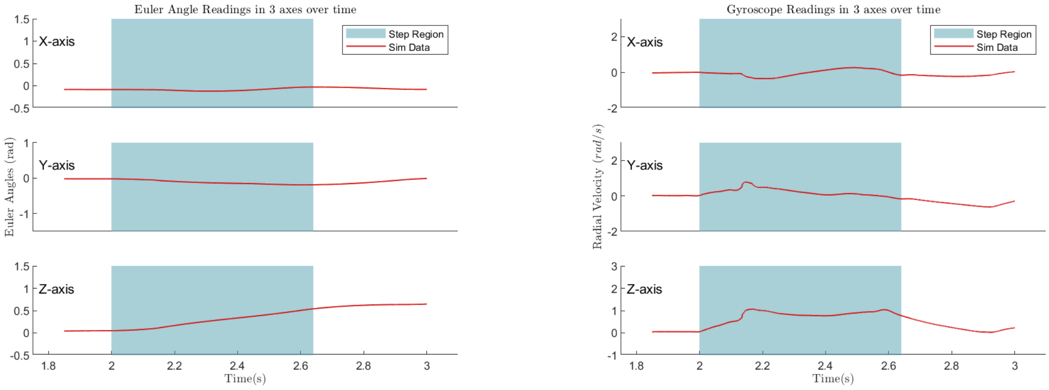
(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.



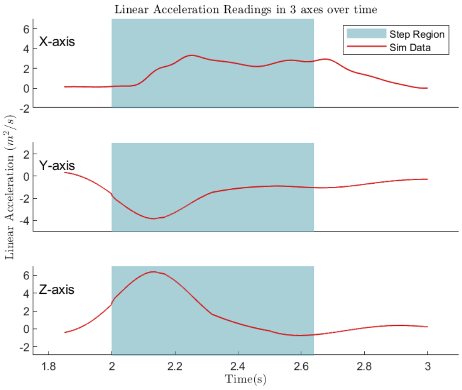
(d) Frame 4. **(e)** Frame 5.

Figure 7.5: 5 frames showing the before, during, and after a successful tailed simulation of a 30° turn at a speed of 4.36 m/s.

The data sets logged split into the turning motion can be seen for the 3.33 m/s simulation in Figure 7.6 at a tail speed of 8 rad/s. An example of a 4.44 m/s run with a tail speed of 8 rad/s can be seen in Figure 7.7.

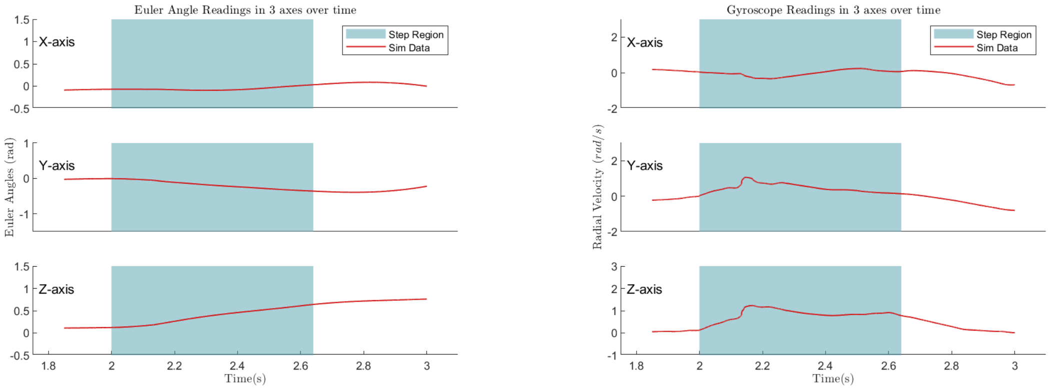


(a) Euler Angles. **(b)** Gyroscope Readings.



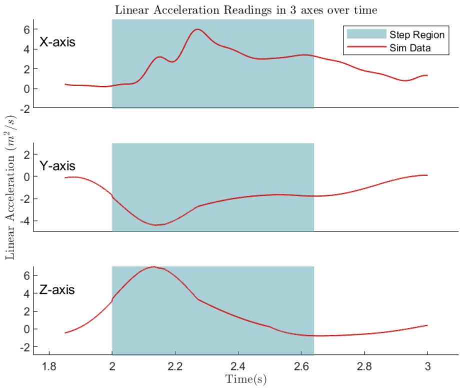
(c) Linear Acceleration.

Figure 7.6: The data logged from the tailed manoeuvre experiment at a speed of 3.33 m/s with a tail speed of 8 rad/s. The IMU data (subfigures (a), (b), and (c)) was logged at 1000 Hz. These plots included the Euler angles, the gyroscope readings, and the linear acceleration in the X, Y, and Z axes.



(a) Euler Angles.

(b) Gyroscope Readings.



(c) Linear Acceleration.

Figure 7.7: The data logged from the tailed manoeuvre experiment at a speed of 4.4 m/s with a tail speed of 8 rad/s. The IMU data (subfigures (a), (b), and (c)) was logged at 1000 Hz. These plots included the Euler angles, the gyroscope readings, and the linear acceleration in the X, Y, and Z axes.

The metrics calculated, taken from the tailed run at 4.4 m/s with a tail speed of 8 rad/s, can be seen in Table 7.2. The metrics were calculated for each vehicle velocity and tail speed and were plotted against the speed at which the simulation was run. This will be shown and discussed in Section 7.1.3.

Table 7.2: The metrics obtained from the tailed manoeuvre experiment at a speed of 4.4 m/s.

Speed (m/s)	**Max Yaw Rate** (rad/s)	**Average Yaw Rate** (rad/s)	**Max Lin Acc** (m/s^2)	**Average Lin Acc** (m/s^2)	**Max Kohut** ($\frac{\dot{\theta} m}{s^2}$)	**Average Kohut** ($\frac{\dot{\theta} m}{s^2}$)	**Integral of Roll Rate** (rad)
4.36	1.24	0.85	6.04	3.40	304.66	213.67	435.92

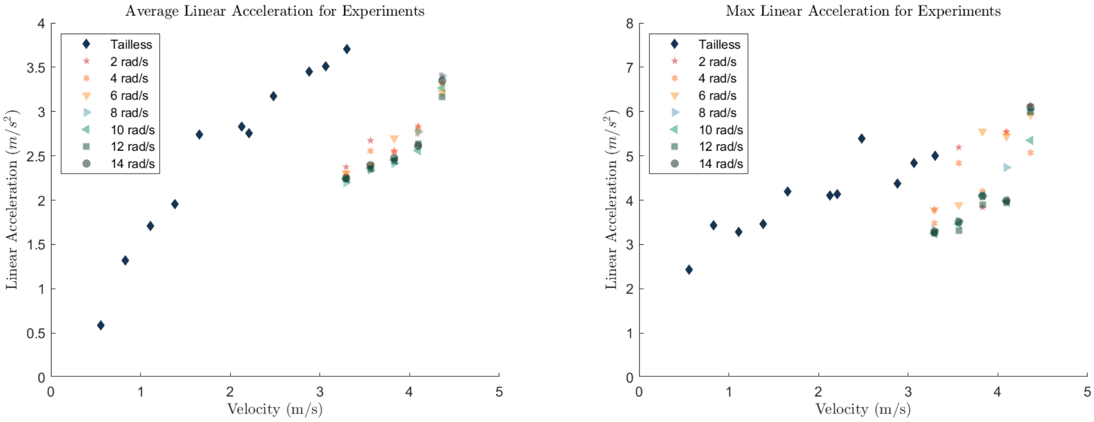
7.1.3 Simulated Results and Analysis

The results were plotted and analysed after the simulations were run. The same seven metrics were used to compare all the runs. All the results were plotted together to give a comparison between the tailless and the tailed versions. The metrics were plotted against the speed at which the simulation was run and, due to the speed which the simulations achieved, the tailless results ended up encompassing the left side of the plots.

Several trends were found from the results. The first thing noted was that the tail had a large effect on the performance of the car. Additionally, the speed of the tail had an effect. The variance of the speed had more of an effect on the linear acceleration and the integral of the roll rate than on the Kohut parameter and the yaw rate. The results, being so spread out in the linear acceleration and the integral of the roll rate, show that the control of the tail can have a significant effect on the outcome of the experiments. While these simulations were run at a fixed commanded tail speed, it can be seen that there will be a control structure that changes the swing speed as needed to produce the optimal results. This different control structure is something that can be looked at in future work as, for this research, the physical experiments were performed with a single speed to test the validity of the design and the basis from which future research can be performed.

It was seen that the maximum metric was higher than the average metric for the three metrics that were compared. This makes sense as the average falls incorporates the initial and end values which were lower than the maximum value. The average metric shows the result of the full turning manoeuvre, while the maximum metric shows the result of the peak of the manoeuvrability. The plotted results and the noted trends for the tailed and tailless simulations are as follows.

For the linear acceleration in Figure 7.8, the average linear acceleration seems to have a linear trend with the tailless model. When adding the tail to the model, the average linear acceleration seems to divert from that trend. It follows its own linear trend at a lower average gradient which shows that the tail was detrimental to the average linear acceleration over the turning manoeuvre. For the maximum linear acceleration, the tailed version seems to roughly follow the same linear model as the tailless version. This shows that the tail has no significant effect on this metric for the maximum value across each tail speed, and not for all tail speeds.



(a) Average Linear Acceleration.

(b) Max Linear Acceleration.

Figure 7.8: The linear acceleration results of both the tailed and the tailless simulations.

When it comes to the yaw rates in Figure 7.10, the tailed model has a lower average and maximum yaw rate than the tailless model. While the tailless yaw rates seem to follow a constant value with variances, the tailed yaw rate seems to follow a lower constant value. This means that the addition of the tail makes the car turn at slower turning speeds and is detrimental to the total turning capabilities of the vehicle. This does not affect the forward velocity but rather the turning speed of the car. It makes sense, however, that the addition of the tail's aerodynamic force means that, while adding a stabilising torque, it produces a turning torque that is opposite to the turning motion. This effect is made obvious when positioning the tail at its vertical position mid-swing. The aerodynamic force creates a torque around the center of rotation that is opposite to the turning motion. An illustration of this can be seen in Figure 7.9. This was also hinted at in other research that looked at using a sail to aid turning [36] where the stabilising torque needs to be maximised while the drag force needs to be minimised. To ensure that the turning motion is not hindered by the tail, it is recommended to look into providing better turning aid to the car. Something like making the car turn for longer periods, or adding more weight to the front axle to ensure traction between the turning wheels and the ground could counter help this issue. Another potential solution would be to position the tail on another part of the car. This could then result in a beneficial turning aid. This, however, is not something seen in biology and so was not considered for this research. That being said, these potential fixes are things that can be looked at in future work.

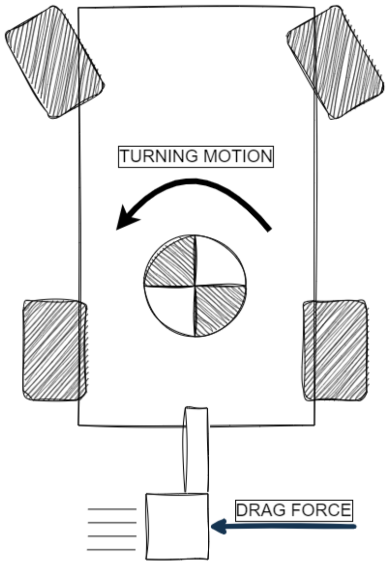
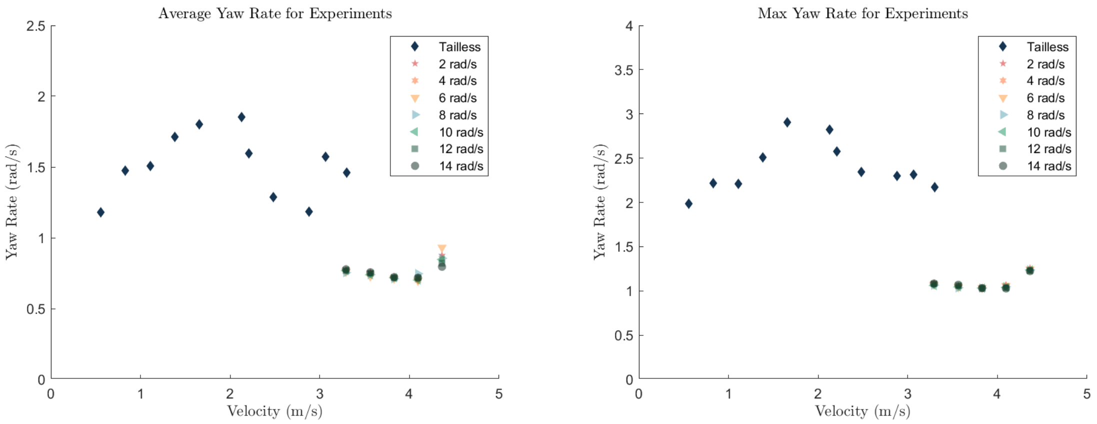


Figure 7.9: An illustration of the turning motion of the car with the aerodynamic force acting as opposite torque on the turning motion of the car.

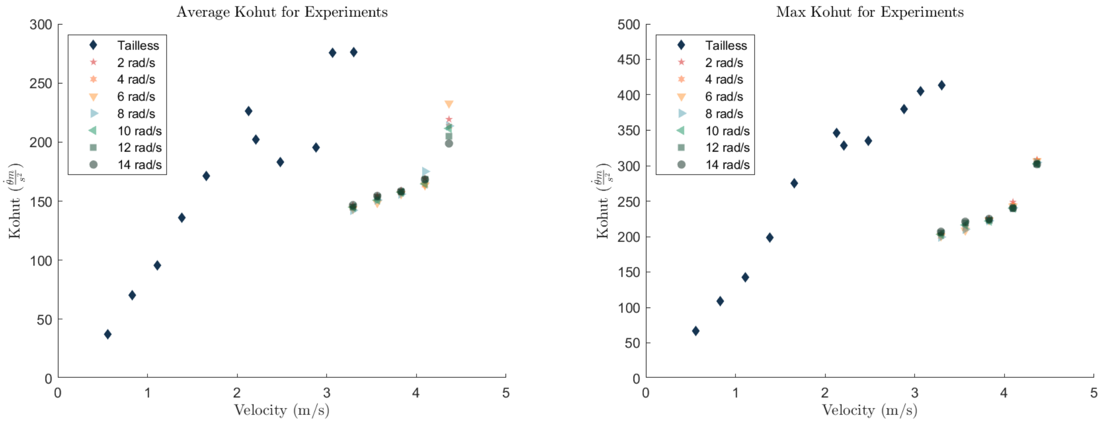


(a) Average Yaw Rate.

(b) Max Yaw Rate.

Figure 7.10: The yaw rate results of both the tailed and the tailless simulations.

As the Kohut metric, shown in Figure 7.11, is directly proportional to the yaw rate, it is not surprising that the tailed simulation has lower average and maximum values than the tailless version. These simulation results show that the tail lacks the ability to increase the manoeuvrability of the car by definition of both the linear acceleration and the Kohut metric.



(a) Average Kohut Metric.

(b) Max Kohut Metric.

Figure 7.11: The Kohut metric results of both the tailed and the tailless simulations.

However, there are two other ways to look at that the tail shows a beneficial effect in. These two ways are the stability of the vehicle, and the speed at which it can turn. The stability, shown by the maximum integral of the roll rate in Figure 7.12, shows how stable the car is in the turning motion. The addition of the tail decreases the value, which means that it increases the stability as the more the car flips, the higher the integral value will be. This means that the tail increases the stability of the car. The other way to look at the tail's effect on the car is the forward velocity at which the car can turn. The tailless model flips at a speed of 3.61 m/s while the tailed model flips at a speed of 4.72 m/s which means that the tail increases the speed at which the car can turn by 23%. This increase in speed is a beneficial effect of the tail as it means that the car can turn at higher speeds. While at higher speeds the car may not turn as much as at lower speeds, the simulations run show that high-speed manoeuvres can be performed with increasing stability due to the addition of the tail.

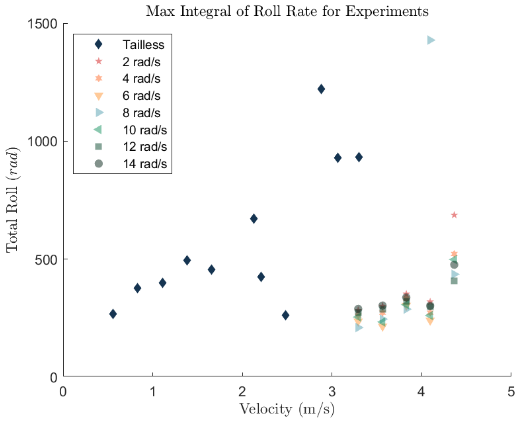


Figure 7.12: The integral of the roll rate results of both the tailed and the tailless simulations.

Taking each of the tailed simulations and finding the best results of the various metrics shows the need for an optimised control system. The maximum values of the maximum linear acceleration, the maximum Kohut values and the minimum values of the integral of the roll rate were found and were plotted with their associated tail and vehicle speeds in Figure 7.13.

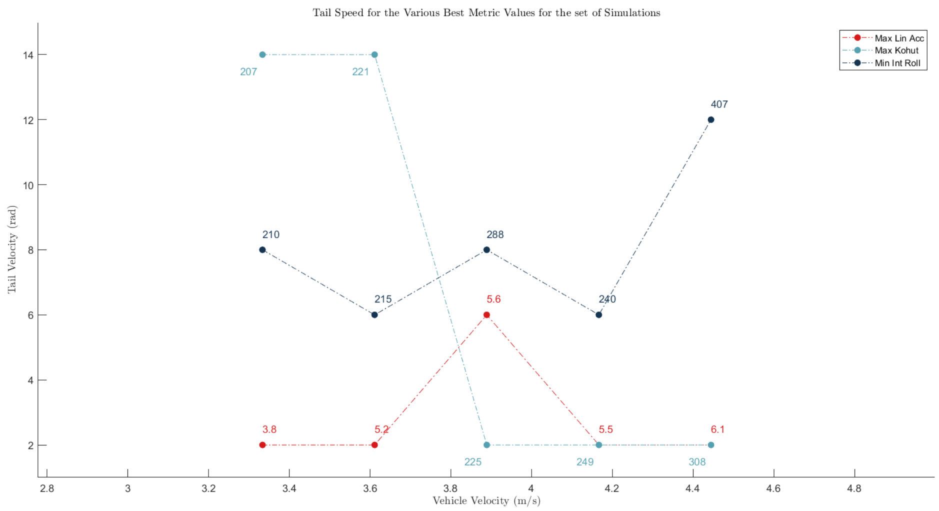


Figure 7.13: The best metrics of the tailed simulations.

Here it can be seen that the best metrics were found at different tail speeds, depending on the desired metric. This shows that the tail control can be optimised to produce the best results for the desired metric and is something that can be looked at in future work. However, for the purposes of this research, it was decided to set the physical tail's speed to be a constant speed chosen to be 8 rad/s. The chosen speed was due to three reasons. The first being that it was the average speed from the tested simulations which was a good starting point as the different maximum metrics had their best values both above and below this 8 rad/s. The second reason was due to the research done with the previous iterations of *Dima*, where the inertial tail was shown to flick at speeds of up to 9 rad/s [35]. The last reason was from the physical safety of the robot. It was seen that 8 rad/s achieved, on average, the most stable results from the lowest integral of the roll rate which meant that at the chosen speed, the car would not flip as easily.

7.2 AeroDima Experiments

The final step in the process was to take what was done in simulation and transfer that the *AeroDima*, the physical robot, meaning that the tailless and tailed experiments were to be performed on the physical robot. The exact steps taken were changed slightly due to the physical limitations of the car, but the concepts to show the effect of the tail were the same. This section, like the previous section, will look at the tailless and tailed experiments separately. The tailless experiments will be looked at first, followed by the tailed experiments. After that, the two sets of experiments will be compared to see the effect of the tail on the physical robot.

7.2.1 Physical Tailless Manoeuvre Tests

The tailless manoeuvre were performed to get a base metric of the car's performance without the tail. In this experiment, the car was set to drive in a straight line and then turn at a constant angle for a certain period. The car was to repeat this while increasing the speed at which it drove until the car flipped over. The steps taken can be seen below.

1	Set up *AeroDima*, without the tail, on the field by powering the platform on, connecting to it, and starting the programs.
2	Set the program to AUTO at a throttle command of 30%.
3	Repeat step 2 while increasing the throttle command until *AeroDima* flips.

The AUTO program is mentioned in the steps above and is the tailless control sequence, run by the Jetson, that is repeatable. This sequence also starts the logging of the GPS and IMU. The AUTO program's sequence is described below in simple steps.

1	Set throttle command to desired command.
2	Start the logging of the sensors.
3	Drive *AeroDima* for 15 seconds to ensure steady-state.
4	Apply a 30° turning angle to the servo motor.
5	After 0.5s, straighten the wheel by applying a 0° angle.
6	Wait for 1s before applying a 0% throttle command.
7	After 20s from the start, stop the logging sequence.

The throttle input being a PWM percent was discussed in Section 6.2 where the straight line speed was modelled to the PWM percent. With the steps laid out, the experiment was performed. As the speed was increased, it was seen that the tailless *AeroDima* could handle turns at speeds of up to 3.37 m/s. This was the maximum speed that *AeroDima* could handle without flipping. The car remained steady until the turning motion and, after steering left, the car would tip to its two right tyres. Once the steering was reverted to zero, the car would then go back to being on all four tyres.

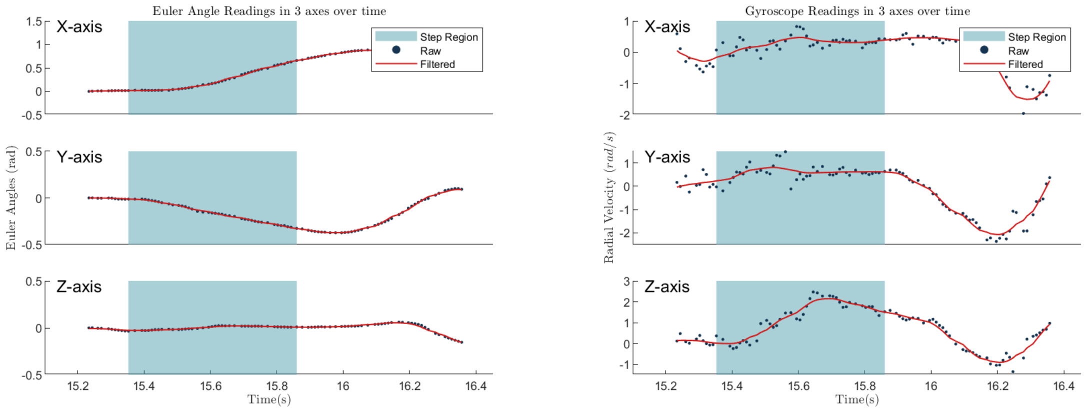
This turning motion can be seen in Figure 7.14. This motion mimicked the simulation's tailless motion and reproduced the expected motion of the vehicle.

(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.

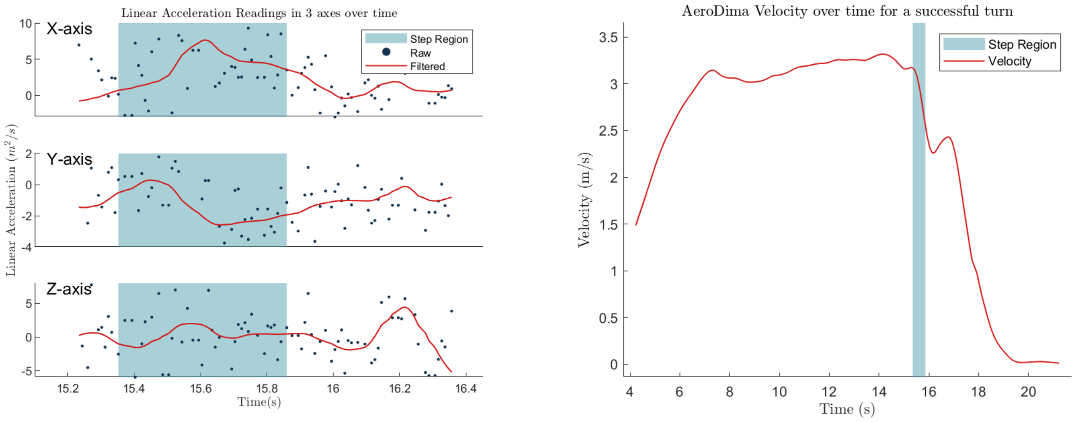
(d) Frame 4. **(e)** Frame 5.

Figure 7.14: 5 frames of the before, during, and after a successful 30° turn at a speed of 3.37 m/s of the tailless car.

The data that was logged for this motion included the Euler angles, the radial velocity, the linear acceleration and the linear velocity. These values were all logged at various frequencies from different sensors. The IMU, which measured the Euler angles, radial velocity, and linear acceleration, logged at 100 Hz while the GPS, which measured the linear velocity, was logged at 10 Hz. The data was then plotted to see the profiles of each sensor while allowing the measuring of the manoeuvrability metrics. An example of the plotted data can be seen in Figure 7.15. This example was obtained from the fastest stable turn, occurring at 3.37 m/s. The shaded region in each graph indicates the period at which the servo was commanded to turn. The raw data was then filtered using two different filters: the IMU data was filtered using a 2nd order Butterworth filer with a cutoff frequency of 5 Hz, while the GPS data was filtered using a 2nd order Butterworth filter with a cutoff frequency of 1 Hz. The filtered data was then used to obtain the metrics for further analysis.



(a) Euler Angles. **(b)** Gyroscope Readings.

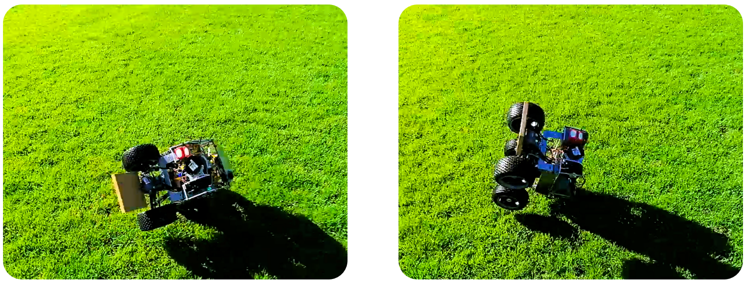


(c) Linear Acceleration. **(d)** Linear Velocity.

Figure 7.15: The data logged from the tailless manoeuvre experiment at a speed of 3.37 m/s. The IMU data (subfigures (a), (b), and (c)) was logged at 100 Hz while the GPS data (subfigure (d)) was logged at 10 Hz.

In the experiment, the speed was increased until the car flipped and it was seen that at 3.61 m/s, the car flipped over. This motion can be seen in Figure 7.16. The manoeuvre started with the same motion as the successful turn as, in the turning motion, the car went onto its two right wheels. However, with the centrifugal force being more than it could handle, it ended up flipping over the front right end of the car. This also matched the motion of the simulation's flipping motion.

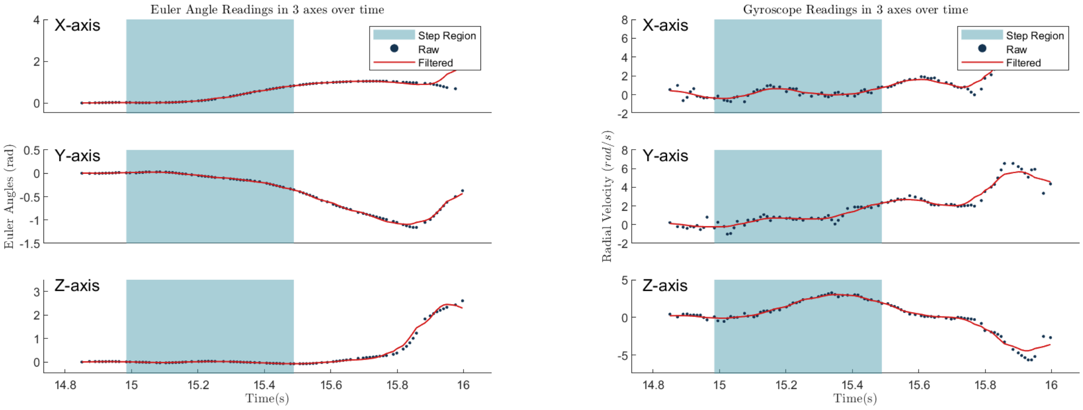
(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.



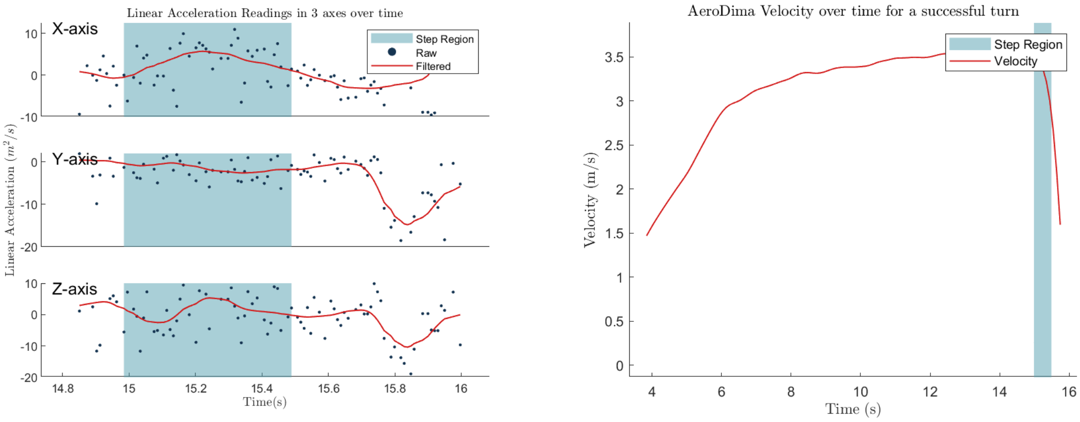
(d) Frame 4. **(e)** Frame 5.

Figure 7.16: 5 frames showing the before, during, and after an unsuccessful 30° turn resulting in a flip at a speed of 3.61 m/s of the tailless car.

Once more, the logged data was filtered and plotted to show the profiles of the manoeuvre. This data can be seen in Figure 7.17. The shaded region in each graph indicates the period at which the servo was commanded to turn.



(a) Euler Angles. **(b)** Gyroscope Readings.



(c) Linear Acceleration.

(d) Linear Velocity.

Figure 7.17: The data logged from the tailless manoeuvre experiment that flipped at a speed of 3.61 m/s. The IMU data (subfigures (a), (b), and (c)) was logged at 100 Hz while the GPS data (subfigure (d)) was logged at 10 Hz.

This data set correlates to the experiment where the car flipped over. Between the two data sets, it is clear which is the unsuccessful turn and can be seen by the significantly larger values in all the IMU readings after the stepping motion. This shows a more aggressive manoeuvre at the end of the motion which occurred when the car flipped. For example, the Euler angle in the Y-axis reached values between 1 and -3 for the successful turn while the same readings got to between 1 and -15 for the flip.

The metrics were calculated for each vehicle velocity and were plotted against the speed at which the simulation was run. These will be shown when comparing the tailed and the tailless runs. However, the metrics for the fastest successful tailless turn can be seen in Table 7.3.

Table 7.3: The metrics obtained from the tailless manoeuvre experiment at a speed of 3.33 m/s.

Speed (m/s)	**Max Yaw Rate** (rad/s)	**Average Yaw Rate** (rad/s)	**Max Lin Acc** (m/s^2)	**Average Lin Acc** (m/s^2)	**Max Kohut** ($\frac{\dot{\theta}m}{s^2}$)	**Average Kohut** ($\frac{\dot{\theta}m}{s^2}$)	**Integral of Roll Rate** (rad)
3.37	1.91	1.24	4.79	3.27	366.82	237.94	25.43

Other than the integral of the roll rate which is dependent on the time step between each data sample, all of these metrics of the experiment are fairly similar to the simulation's metrics. This will be further expanded upon after all experiments are reported on.

7.2.2 Physical Tailed Manoeuvre Tests

The climatic experiment was last to be performed. The steps taken were similar to the tailless experiment, but with the tail attached. The steps can be seen below.

1	Set up *AeroDima*, with the tail, on the field by powering the platform on, connecting to it, and starting the programs.
2	Set the program to AUTO at a throttle command of 35%.
3	Repeat step 2 while increasing the throttle command by 5% until *AeroDima* flips or until a final chosen percent.

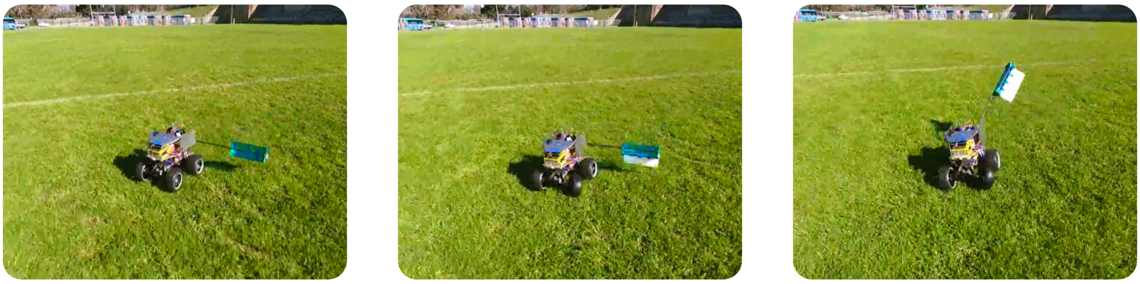
The tests were performed according to these steps. The tail control, as spoken about in Section 6.3.4, was used to flick the tail and was triggered when the yaw rate reading was over 1 rad/s. As spoken about, the control system flicked the tail at a steady speed. This speed was chosen to be 8 rad/s for various reasons spoken about in Section 7.1.3, but largely due to this being the speed that produced the lowest roll rate across the various speeds in the simulation.

While future work may focus on the control of the tail and incorporate more tail speeds, for the scope of this research a single chosen speed was enough to prove the function and form of *AeroDima*. Due to this limited control structure, the turning sequence of the platform was also limited. As the tail was only actuated in the roll motion, the tail would end up on the opposite side of the robot at the end of the flick. This meant that the robot could only turn in opposite directions sequentially, while not being able to turn in the same direction twice in a row. With a better control structure, this would be mitigated by allowing the tail to move back to the side it was needed on before a turn was attempted, albeit in a slower motion than the flick itself. This also meant that the turning sequence had to be preemptively decided upon, otherwise the tail would hinder the motion more than aid it. With a different, 2DOF motion, this limitation would be reduced and would allow a better instinctual turning structure. However, since the purpose of this research was more to optimise a single turn rather than multiple turns, it was deemed to not have to worry about this limitation at this period in time.

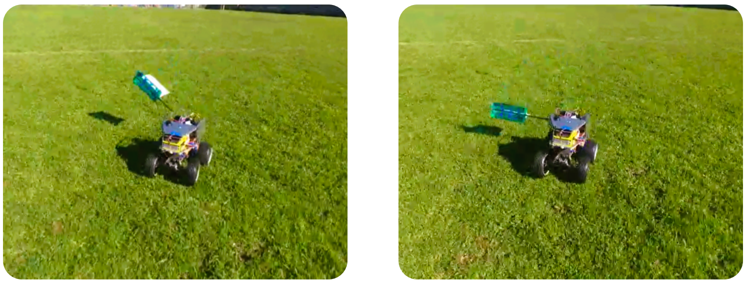
The velocity of the car was tested successfully, each time increasing the forward velocity up to a throttle command of 50%. Once a successful turn was completed, the speed was increased to the next level. However, at 55% something interesting happened, the tail didn't flick and car performed its turning manoeuvre without flipping. It was unsure of why the tail didn't flip, however there are two potential reasons for it. The first being that the IMU got disconnected or stopped recording and so the tail couldn't be triggered during the turning motion. This cause is likely as the logged IMU data was only recorded for the first couple of seconds and stopped logging during the acceleration period for the 55% test. The second potential reason for it was that, even though the IMU stopped logging, the yaw rate never exceeded the threshold which would cause the tail to turn. The result of the unintended experiment proves what was mentioned in the simulation section, that a finer control of the tail would aid the stability and turning capabilities of the vehicle. At higher speeds, it could be seen that something of a more passive tail may be beneficial while at lower speeds, an active tail is more beneficial for turning manoeuvrability. This relationship is something that can be looked at in future work. This meant that there were only four successful, recorded turns operated until this phenomenon occurred. An example of a successful turn at a speed of 3.61 m/s can be seen in Figure 7.18. A video of the experiment can be found in the link in the footnote *.

---

*An accompanying video: https://youtube/NxzenyeAGKg



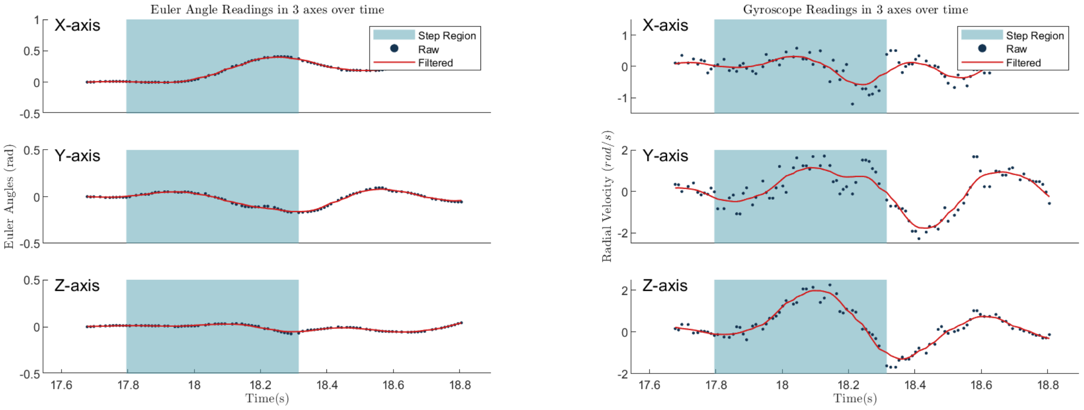
(a) Frame 1. **(b)** Frame 2. **(c)** Frame 3.



(d) Frame 4. **(e)** Frame 5.

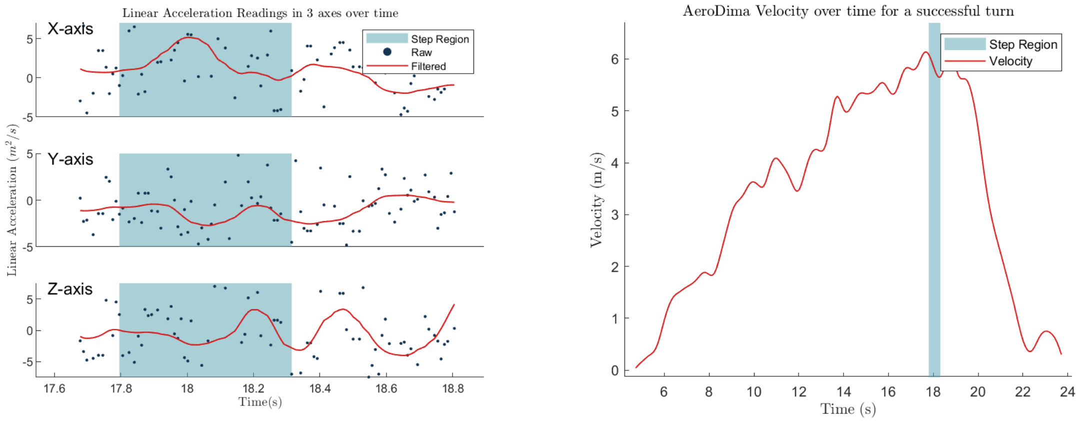
Figure 7.18: Before, during, and after a successful 30° turn at a speed of 6.10 m/s.

The data logged from each test was sorted into its profiles, and the metrics evaluated. The profiles of the Euler angles, the gyroscope readings, the linear acceleration, and the linear velocity for the fastest tailed experiment measured can be seen in Figure 7.19. The shaded region in each graph indicates the period at which the servo was commanded to turn. The data was then filtered using the same filters as the tailless data and this filtered data was then used to obtain the metrics for further analysis.



(a) Euler Angles.

(b) Gyroscope Readings.



(c) Linear Acceleration.

(d) Linear Velocity.

Figure 7.19: The data logged from the tailed manoeuvre experiment at a speed of 6.10 m/s. The IMU data (subfigures (a), (b), and (c)) was logged at 100 Hz while the GPS data (subfigure (d)) was logged at 10 Hz.

The fastest tailed test was used to get the maximum performance metrics, while the other runs were also analysed to get the metrics for their tests. The results of the fastest run's metrics can be seen in Table 7.4 below.

Table 7.4: The metrics obtained from the tailless manoeuvre experiment at a speed of 3.33 m/s.

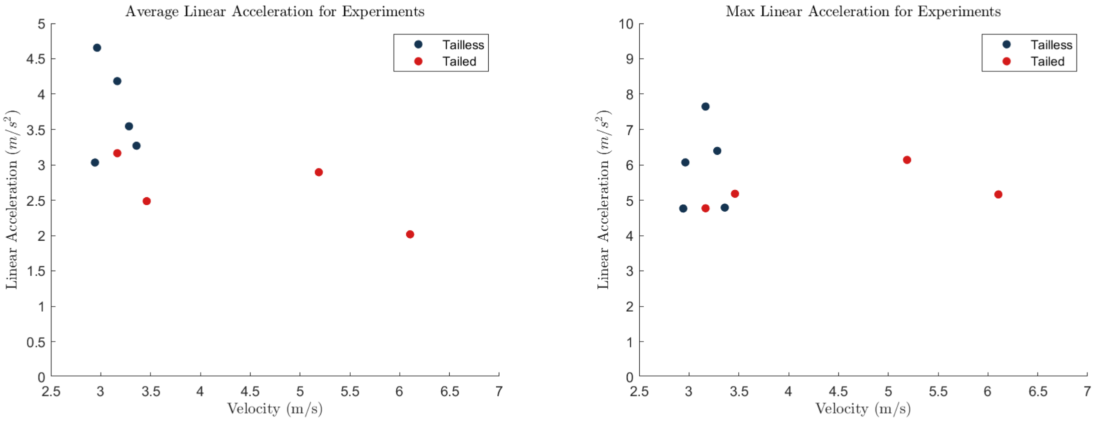
Speed (m/s)	**Max Yaw Rate** (rad/s)	**Average Yaw Rate** (rad/s)	**Max Lin Acc** (m/s^2)	**Average Lin Acc** (m/s^2)	**Max Kohut** $(\frac{\dot{\theta}m}{s^2})$	**Average Kohut** $(\frac{\dot{\theta}m}{s^2})$	**Integral of Roll Rate** (rad)
6.10	1.98	0.70	5.16	2.01	692.57	245.34	15.98

7.2.3 Experimental Results and Analysis

The tailed and tailless metrics were then compared after all the desired experiments were run. The metrics looked at the average linear acceleration of the car over the turning period, the maximum linear acceleration, the average yaw rate, the maximum yaw rate, the average Kohut, the maximum Kohut, and the integral of the roll rate. These results will be plotted below.

The real experiment had fewer data sets than the simulation due to the nature of running the experiments. The simulation had multiple tests at speeds that went from 0.56 m/s to 4.44 m/s with the tailless and tailed models, but the experiment with *AeroDima* had fewer runs as the tailless runs did not incorporate as many lower speed runs as the simulation. This was because the focus was only getting a comparison near the limits of the tailless version, compared to getting an overall trend of the vehicle's metrics across all possible velocities. With the tailed version, each test was performed at an increasing speed. These speeds, while having larger speed intervals than the simulation, ended up going much higher than the simulation. The maximum speed that the tailed version ran at was 6.10 m/s compared to the maximum speed of the simulation of 3.30 m/s.

Looking at the specific metrics, the linear acceleration of the vehicle in Figure 7.20 seemed to remain constant along the tailed runs regardless of the forward velocity of the vehicle. The tailless linear acceleration experienced some scatter, much largely similar across the runs. The fact that the trends for both the average and the maximum linear accelerations for the runs look similar to each other show that the profiles of the linear acceleration remain similar except being scaled in the y-axis. These results show that the added aerodynamic tail does not seem to have a beneficial effect on the linear acceleration of the vehicle and this is similar to the simulation results. It does not seem to be as detrimental to the average linear acceleration as the simulation showed, but it does not seem to be beneficial either.

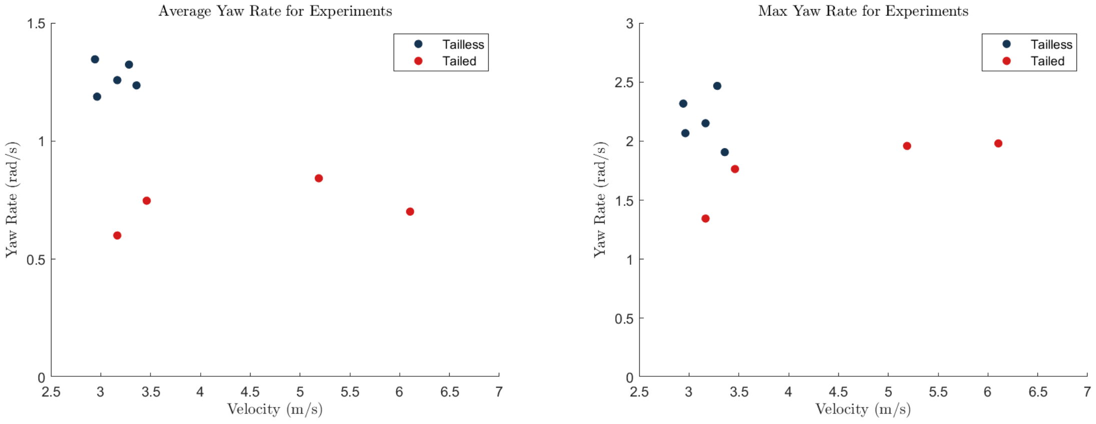


(a) Average Linear Acceleration.

(b) Max Linear Acceleration.

Figure 7.20: The linear acceleration results of both the tailed and the tailless experiments.

When looking at the yaw rates in Figure 7.21, the tail seems to reduce the yaw rate compared to the runs without the tail. The tail, or at least the implemented control system of the tail, seems to hinder the yaw rate of the vehicle. This means that the vehicle can turn more at lower speeds, and while it can still turn at high speeds, these turns are less effective. This aligns with what was seen in the simulations as well.

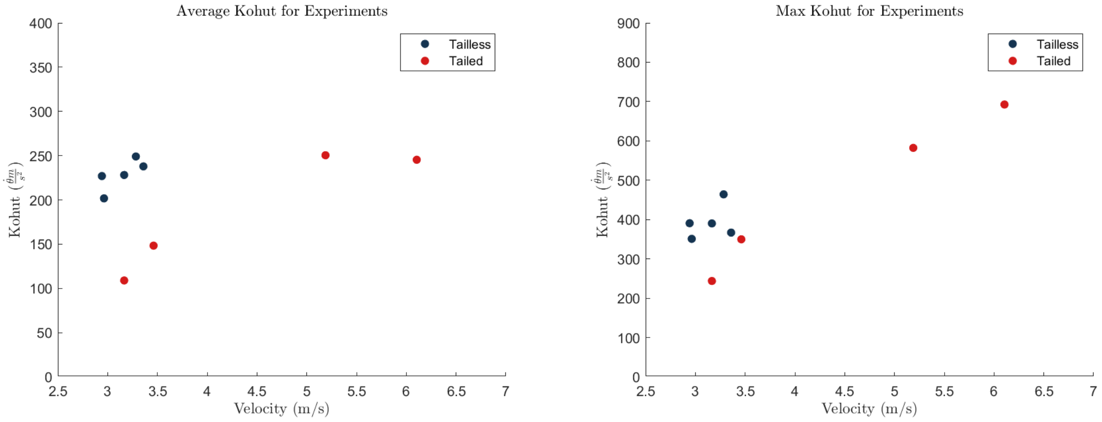


(a) Average Yaw Rate.

(b) Max Yaw Rate.

Figure 7.21: The yaw rate results of both the tailed and the tailless experiments.

For the average Kohut metric in Figure 7.22, the same trends can be seen from the simulation to the real experiments. The tail does not seem to increase the average Kohut metric, rather it is seen to decrease the average Kohut metric at lower speeds while keeping constant at higher speeds. What is surprising to note is the maximum Kohut metric. For this metric it seems that the tail is rather beneficial. As the average Kohut did not show this, it means that the peak yaw rate of the tailed *AeroDima* is significantly higher than the tailless version, however it has a shorter period for its yaw rate. This shows that the aerodynamic tail can provide a beneficial instantaneous manoeuvrability capability, but over longer periods, this beneficial capability falls away.



(a) Average Kohut Metric.

(b) Max Kohut Metric.

Figure 7.22: The Kohut metric results of both the tailed and the tailless experiments.

When looking at the integral of the roll rate in Figure 7.23, the tailed version's magnitude is significantly less than the tailless version's. This shows that the stability of the vehicle remained constant throughout the various tests. While the tailless runs showed the vehicle turning on its two outside wheels for a portion of the time, this was limited with the addition of the tail. This was seen in the simulation as well, but the real tests showed a more significant decrease in the integral of the roll rate which shows that the tail is significantly beneficial to the stability of the vehicle. The difference between the simulation and real absolute values can be attributed to the fact that they had different interval time steps. For this reason, the specific value of the integral of the roll rate is not as important as the trend of the metric between the tailless and the tailed versions.

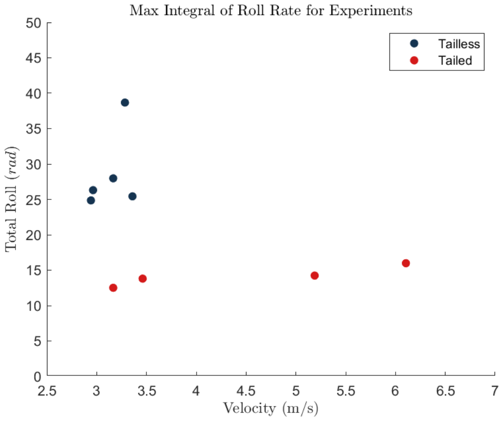


Figure 7.23: The integral of the roll rate results of both the tailed and the tailless experiments.

Depending on which metric one looks at, the aerodynamic tail design and created can be seen to be beneficial to the vehicle at high-speed manoeuvres. While the total turning capability decreases, the speed at which it can perform a turn increases significantly. From a speed of 3.37 m/s to a speed of 6.10 m/s, *AeroDima* was able to increase its forward velocity at which it could perform a turning manoeuvre by 81.01%. Not only this, but the stability of the vehicle increased significantly. From the integral of the roll rate, it can be seen that the tailless version had a value of 25.43 rad while the tailed version had a value of 15.98 rad. This decrease in magnitude results in a stability increase of 37.16%. The maximum Kohut value measured also increased by 88.80%. This shows that the designed dynamic aerodynamic tail provides better aid in instantaneous manoeuvres rather than drawn out movements. Once again, this shows that with the incorporation of a more advanced control system, the tail could be used to aid in the manoeuvrability of the vehicle to an even higher degree.

7.3 Summary

In summation, this chapter both simulated and physically tested the performance of *AeroDima*. The simulations showed that the aerodynamic robotic tail did not aid in the linear acceleration, yaw rate, and Kohut metrics, but it was beneficial in stabilising the vehicle and allowing the vehicle to increase the speed at which a turning manoeuvre took place. This increase in forward velocity was an increase of 23% up to a speed of 4.36 m/s. The physical experiments showed that the tail performed better than the simulations postulated. The tail was able to increase the forward velocity at which a turning manoeuvre took place by 81.01% up to a speed of 6.10 m/s. The stability of the vehicle also increased by 37.16% by looking at the values of the integral of the roll rate. The maximum Kohut value measured also increased by 88.80% which shows that the tail increased the manoeuvrability of *AeroDima*, however, this increase was an instantaneous manoeuvrability rather than a constant manoeuvrability. Therefore, the tail is beneficial to the vehicle, but it is not just the tail alone that affects the manoeuvrability of the vehicle. The control system of the tail is also a factor that affects the manoeuvrability of the vehicle and is something that future work can look into.

Chapter 8: Conclusion

This chapter concludes the research by summarising the work done and the results obtained. The significance of the research towards the field are discussed. Recommendations for future research on the topic are also provided.

8.1 Conclusion of Research

A robotic platform was designed and built to test the effects that an aerodynamic tail could have on the platform. The concept for an aerodynamic tail was inspired by the cheetah and its use of its tail during high-speed turning manoeuvres. Previous studies assumed all inertial effects while this study aimed at incorporating aerodynamic drag as the main force acting on the tail.

This was achieved through the study of the cheetah's tail. With the use of a wind tunnel, the aerodynamic coefficients of the tail were obtained for various wind speeds and angles of attack, producing an aerodynamic model that can be used to simulate the tail's behaviour under static conditions. This model was then used in comparing the tail's dynamic motion to the static model to determine the model's proficiency under dynamic conditions. It was seen that this quasi-steady-state model was sufficient for lower Reynolds Numbers for the cheetah tail, but as the Reynolds Number increased, the model's accuracy decreased.

The robotic platform, named *AeroDima*, was designed and built to test the effects of an aerodynamic tail on a robotic platform. The platform, a modified remote control car, was designed to reach speeds of over 20 m/s and built to turn at those speeds. A safety roll cage was added to protect the electronics housed in car in the event of a crash or roll-over. *AeroDima* was designed with multiple microcontrollers and motors that allowed for the control of the platform's motion.

The robotic aerodynamic tail was also designed. The shape of the tail was chosen between eight shapes that were analysed in SolidWorks' Flow Simulation toolbox. Comparing these eight shapes, the shape with the highest simulated drag coefficient was chosen. This shape was a half cylinder with end caps design. The tail was designed to be attached to the rear of the platform and be actuated in two degrees of freedom. The testing of the tail in this study however only looked at a single degree of freedom, the roll axis.

The tail was tested in a wind tunnel to determine its aerodynamic model like with the cheetah tail. A model was developed to simulate the tail's behaviour under the required conditions of the robotic tail. This model was a single drag coefficient model with a drag coefficient of 2. The model was then used in the simulation of the tail's behaviour on the robotic platform.

A simulation of *AeroDima* was developed using MATLAB's Simscape toolbox. This simulation served to provide a place where the control of the car could be tested before applying it to the physical robot.

The variables used in the simulation were either taken from the SolidWorks model of *AeroDima* or from various experiments performed on certain components of the platform.

A state based control system was designed to control the tail on the robot. This control system was a position and speed state system with a yaw rate trigger to activate the motion of the tail. The control system was tested in the simulation of *AeroDima* and then applied to the physical robot.

AeroDima was then tested, both in simulation first and then in reality. These tests included both without the presence of the tail, and with the presence of the tail. The tailless tests provided a basis from which the tail's effects could be compared to. The simulation showed the same effects that the real system portrayed except the simulation showed diminished effects. The real tests showed a more significant effect of the tail on the platform.

The addition of the tail on the platform showed an increase of certain manoeuvrability metrics while showing a decrease in others. The lateral acceleration of the vehicle decreased with the addition the tail as well as the yaw rate experience. The tail reduced the ability to turn at high speeds. However, the tail increased the stability of the car, the maximum Kohut metric, and the top speed that the car could perform a high-speed manoeuvre. The roll stability of the car increased by 37.16% with the addition of the tail. The maximum Kohut metric, the maximum instantaneous yaw rate multiplied by the forward velocity, increased by 88.8%. The top speed that the car could perform a high-speed manoeuvre increased by 81.01% achieving a top speed of 6.10 m/s.

8.2 Significance of Research

During this research project, the following contributions were made to the field of robotics and bio-inspired robotics:

- Three cheetah tails were studied in a wind tunnel to produce an aerodynamic model of the tail. This model was a quasi-steady-state model that was shown to be more accurate at low Reynolds Numbers. An added mass effect was also studied and shown to be aid the model but only at certain relative wind speeds.
- *AeroDima*, a robotic platform, was designed and built to test the effects of tails on a robotic platform. This mechatronic design included the mechanical and embedded design of the robot.
- A drag coefficient analysis was performed on eight different tail shapes to determine the best tail shape for an aerodynamic tail. This shape was a half cylinder with end caps both either ends.
- A simulation of *AeroDima* was built using MATLAB's Simscape toolbox. This simulation can be used to test the control system of the robot to safely measure the performance of the control system before applying it to the physical robot. This simulation included the aerodynamic model of the tail as measured through the aid of a wind tunnel experiment. The model included a single drag coefficient model with a drag coefficient of 2.
- Eight manoeuvrability metrics were measured to determine the effects of the added aerodynamic tail on the platform. The aerodynamic showed a beneficial effect on three of these metrics, while

showing a negative effect on five of them. The tail decreased the yaw rate of the vehicle, the lateral acceleration of the vehicle, and the average Kohut metric. The tail increased the stability of the vehicle, the maximum Kohut metric, the top speed that the vehicle could perform a high-speed manoeuvre.

- The addition of an actuated aerodynamic tail on a moving robotic vehicle allows for faster high-speed manoeuvres to be performed. The swing of the tail increases the stability of the platform on which the tail is on. The main benefit of the tail can be seen in the instantaneous increase of manoeuvrability as seen by the 88.8% increase in the maximum instantaneous Kohut metric. Manoeuvres of lengthy durations were not seen to be aided by the tail while the tail was actuated in the steady-state control system employed.

8.3 Recommendations for Future Research

The research steps in this study were done in a way that was controlled by the scope. Further research can be done using this research as a stepping stone to further the field of bio-inspired robotics. An obvious further step is to develop a more accurate aerodynamic model for both the cheetah's tail and the robotic tail that encapsulates the drag forces through the various Reynolds Numbers from static speeds to dynamic relative speeds up to 30 m/s. This range covers the laminar, transition, and turbulent flow regimes. The more accurate the model, the more accurate the simulations run to test the performance of the vehicle.

In that vein, a further development of the simulation can be done ensuring all physical properties of the model were correct rather than relying on the 3D model of the vehicle. This research provided a well assumed base from which testing could start which produced satisfactory results.

As seen in the results, the state based control showed beneficial results in three of the metrics. However, the results also showed that the tail's motion could be controlled in a more advanced control algorithm to provide better results. It was seen like at higher speeds, the tail could act as a more passive tail instead of an active tail. This will need further research to determine the best control system for the tail to provide the best results for manoeuvrability at high-speed manoeuvres. It is recommended to look into a torque controller for the tail. This will be best benefited by a more accurate aerodynamic model of the tail and so it is recommended to do the previous recommendations first before looking into using a torque controller.

AeroDima was built with an actuated 2DOF tail system but was tested using a 1DOF actuated system. This provides an obvious path of future work to implement the second degree of freedom into the tail's motion, requiring a more developed control system to control the tail's motion. It was seen in [7] that a conical motion of an inertial tail was attempted to aid in the manoeuvrability of a robotic platform. This was proven unsuccessful due to the large torques required for the tail to overcome gravity. The lightweight aerodynamic tail should not have this issue and so it is recommended to look into various motions that would aid in manoeuvrability, a conical motion being one of those.

8.4 Summary

Cheetah tails were studied to develop a quasi-steady-state aerodynamic model using the Euler-Lagrangian dynamics method. Following that, a robotic platform was designed and developed to test the validity of a light-weight aerodynamic tail on a moving platform. The testing and results of the platform and tail were analysed. The results of the aforementioned testing was detailed, conclusions were drawn, significant points expanded upon, and future recommendations were proposed.

Bibliography

[1] A. McClung, “Techniques for dynamic maneuvering of hexapedal legged robots”, Ph.D. dissertation, Stanford University, 2006.

[2] N. J. Kohut, A. O. Pullin, D. W. Haldane, D. Zarrouk and R. S. Fearing, “Precise dynamic turning of a 10 cm legged robot on a low friction surface using a tail”, in *Proceedings - IEEE International Conference on Robotics and Automation*, 2013, pp. 3299–3306. DOI: `10.1109/ICRA.2013.6631037`.

[3] N. C. Sharp, “Timed running speed of a cheetah (Acinonyx jubatus)”, *Journal of Zoology*, vol. 241, no. 3, pp. 493–494, Mar. 1997. DOI: `10.1111/j.1469-7998.1997.tb04840.x`.

[4] J. W. Wilson, M. G. Mills, R. P. Wilson *et al.*, “Cheetahs, Acinonyx jubatus, balance turn capacity with pace when chasing prey”, *Biology Letters*, vol. 9, no. 5, Sep. 2013. DOI: `10.1098/rsbl.2013.0620`.

[5] F. R. Walther, “Flight behaviour and avoidance of predators in thomson’s gazelle (gazella thomsoni guenther 1884) on jstor”, *Behaviour Vol. 34, No.3*, pp. 1884–221, 1969.

[6] BBC, *Cheetah chases wildebeest | the hunt - bbc one*, (Accessed on 01/12/2023), Nov. 2015. [Online]. Available: `https://www.youtube.com/watch?v=v7p6VZiRInQ&ab_channel=BBC`.

[7] A. Patel, “Understanding the motions of the cheetah tail using robotics”, Ph.D. dissertation, University of Cape Town, 2014. [Online]. Available: `https://pdfs.semanticscholar.org/58a5/8364b72009c05d6ade11425b4a4e2c91008c.pdf`.

[8] A. Patel, E. Boje, C. Fisher, L. Louis and E. Lane, “Quasi-steady state aerodynamics of the cheetah tail”, *Biology Open*, vol. 5, no. 8, pp. 1072–1076, 2016. DOI: `10.1242/bio.018457`.

[9] Z. Cao and D. K. Tafti, “Investigation of drag, lift and torque for fluid flow past a low aspect ratio (1:4) cylinder”, *Computers and Fluids*, vol. 177, pp. 123–135, 2018. DOI: `10.1016/j.compfluid.2018.10.002`.

[10] H. A. Son, S. Lee and J. Lee, “Numerical analysis of drag force acting on 2D cylinder immersed in accelerated flow”, *Water (Switzerland)*, vol. 12, no. 6, Jun. 2020. DOI: `10.3390/w12061790`.

[11] C.E.Brennen, “A review of added mass and fluid inertial forces”, Naval Civil Engineering Laboratory, California, Tech. Rep. January, 1982.

[12] B. Mutlu Summer and J. Fredsoe, “In-line force in oscillatory flow”, in *Hydrodynamics Around Cylindrical Structures*, Advanced S, World Scientific, 1997, ch. 4.1.3 The Morrison Equation, pp. 297–333. DOI: `10.1142/9789812795748_0007`.

[13] E. J. Grift, N. B. Vijayaragavan, M. J. Tummers and J. Westerweel, “Drag force on an accelerating submerged plate”, *Journal of Fluid Mechanics*, vol. 866, pp. 369–398, 2019. DOI: `10.1017/jfm.2019.102`.

[14] J. Boussinesq, *Application des potentiels à l'étude de l'équilibre et du mouvement des solides élastiques: principalement au calcul des deformations et des pressions que produisent, dans ces solides, des efforts quelconques exercés sur und petite partie de leur surface ou de leur intérieur; memoire suivi de notes étendues sur divers points de physique mathématique et d'analyse.* Gauthier-Villars, 1885.

[15] A. Basset, "On the steady motion of an annular mass of rotating liquid", *American Journal of Mathematics*, pp. 172–181, 1889.

[16] S. T. Paizis and W. H. Schwarz, "Entrainment rates in turbulent shear flows", *Journal of Fluid Mechanics*, vol. 68, pp. 297–308, 2 Mar. 1975, ISSN: 1469-7645. DOI: `10.1017/S002211207500081X`.

[17] A. Jusufi, D. T. Kawano, T. Libby and R. J. Full, "Righting and turning in mid-air using appendage inertia: Reptile tails, analytical models and bio-inspired robots", *Bioinspiration and Biomimetics*, vol. 5, no. 4, p. 045 001, Dec. 2010. DOI: `10.1088/1748-3182/5/4/045001`.

[18] A. Jusufi, Y. Zeng, R. J. Full and R. Dudley, "Aerial Righting Reflexes in Flightless Animals", *Integrative and Comparative Biology*, vol. 51, no. 6, pp. 937–943, Dec. 2011. DOI: `10.1093/icb/icr114`.

[19] E. Chang-Siu, T. Libby, M. Tomizuka and R. J. Full, "A lizard-inspired active tail enables rapid maneuvers and dynamic stabilization in a terrestrial robot", pp. 1887–1894, Dec. 2011. DOI: `10.1109/iros.2011.6094658`.

[20] T. Libby, T. Y. Moore, E. Chang-Siu *et al.*, "Tail-assisted pitch control in lizards, robots and dinosaurs", *Nature*, vol. 481, no. 7380, pp. 181–186, 2012. DOI: `10.1038/nature10710`.

[21] G. H. Liu, H. Y. Lin, H. Y. Lin, S. T. Chen and P. C. Lin, "A bio-inspired hopping kangaroo robot with an active tail", *Journal of Bionic Engineering*, vol. 11, no. 4, pp. 541–555, Oct. 2014. DOI: `10.1016/S1672-6529(14)60066-4`.

[22] D. G. Lee, H. G. Kim and T. W. Seo, "Experimental Study on Drag-induced Balancing via a Static Tail for Water-running Robots", *Journal of Bionic Engineering*, vol. 13, no. 4, pp. 537–543, Oct. 2016. DOI: `10.1016/S1672-6529(16)60326-8`.

[23] J. W. Young, G. A. Russo, C. D. Fellmann, M. A. Thatikunta and B. A. Chadwell, "Tail function during arboreal quadrupedalism in squirrel monkeys (*Saimiri boliviensis*) and tamarins (*Saguinus oedipus*)", *Journal of Experimental Zoology Part A: Ecological Genetics and Physiology*, vol. 323, no. 8, pp. 556–566, Jul. 2015. DOI: `10.1002/jez.1948`.

[24] P. E. Hudson, "The structural and functional specialisation of locomotion in the cheetah (acinonyx jubatus)", Ph.D. dissertation, Royal Veterinary Colledge, London, 2011, pp. 112–120.

[25] G. A. Bartholomew and H. H. Caswell, "Locomotion in Kangaroo Rats and Its Adaptive Significance", *Journal of Mammalogy*, vol. 32, no. 2, p. 155, 1951. DOI: `10.2307/1375371`.

[26] H. G. Kim, K. Jeong and T. W. Seo, "Analysis and Experiment on the Steering Control of a Water-running Robot Using Hydrodynamic Forces", *Journal of Bionic Engineering*, vol. 14, no. 1, pp. 34–46, Jan. 2017. DOI: `10.1016/S1672-6529(16)60376-1`.

[27] M. Rober, *Building the Perfect Squirrel Proof Bird Feeder - YouTube*, 2020. [Online]. Available: `%7Bhttps://www.youtube.com/watch?v=hFZFjoX2cGg%5C&ab_channel=MarkRober%7D` (visited on 06/05/2021).

[28] T. Fukushima, R. Siddall, F. Schwab *et al.*, "Inertial Tail Effects during Righting of Squirrels in unexpected Falls: From Behavior to Robotics", *Integrative and Comparative Biology*, Apr. 2021. DOI: `10.1093/icb/icab023`.

[29] J. Norby, J. Y. Li, C. Selby, A. Patel and A. M. Johnson, "Enabling Dynamic Behaviors With Aerodynamic Drag in Lightweight Tails", *IEEE Transactions on Robotics*, pp. 1–10, 2021. DOI: `10.1109/TRO.2020.3045644`.

[30] I. J. Yeaton, S. D. Ross, G. A. Baumgardner and J. J. Socha, "Undulation enables gliding in flying snakes", *Nature Physics*, vol. 16, no. 9, pp. 974–982, Sep. 2020. DOI: `10.1038/s41567-020-0935-4`.

[31] A. Jusufi, D. I. Goldman, S. Revzen and R. J. Full, "Active tails enhance arboreal acrobatics in geckos", *Proceedings of the National Academy of Sciences - PNAS*, pp. 4215–4219, 11 2008.

[32] J. Zhao, T. Zhao, N. Xi, M. W. Mutka and L. Xiao, "MSU Tailbot: Controlling Aerial Maneuver of a Miniature-Tailed Jumping Robot", *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 6, pp. 2903–2914, Dec. 2015. DOI: `10.1109/TMECH.2015.2411513`.

[33] R. Briggs, J. Lee, M. Haberland and S. Kim, "Tails in biomimetic design: Analysis, simulation, and experiment", in *IEEE International Conference on Intelligent Robots and Systems*, 2012, pp. 1473–1480. DOI: `10.1109/IROS.2012.6386240`.

[34] A. O. Pullin, N. J. Kohut, D. Zarrouk and R. S. Fearing, "Dynamic turning of 13 cm robot comparing tail and differential drive", in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 5086–5093. DOI: `10.1109/ICRA.2012.6225261`.

[35] A. Patel and M. Braae, "Rapid turning at high-speed: Inspirations from the cheetah's tail", *IEEE International Conference on Intelligent Robots and Systems*, pp. 5506–5511, 2013. DOI: `10.1109/IROS.2013.6697154`.

[36] N. J. Kohut, D. Zarrouk, K. C. Peterson and R. S. Fearing, "Aerodynamic steering of a 10 cm high-speed running robot", pp. 5593–5599, 2013. DOI: `10.1109/IROS.2013.6697167`.

[37] S. Hoerner, "Fluid-dynamic drag: Practical information on aerodynamic drag and hydrodynamic resistance", *Journal of Physics: Conference Series*, vol. 75, pp. 1–455, 1965, ISSN: 1742-6596.

[38] P. Webb, "Maneuverability - General issues", *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, pp. 547–555, Jul. 2004. DOI: `10.1109/JOE.2004.833220`.

[39] D. L. Jindrich and M. Qiao, "Maneuvers during legged locomotion", *Chaos*, vol. 19, no. 2, p. 26 105, 2009. DOI: `10.1063/1.3143031`.

[40] A. M. Wilson, J. C. Lowe, K. Roskilly, P. E. Hudson, K. A. Golabek and J. W. McNutt, "Locomotion dynamics of hunting in wild cheetahs", *Nature*, vol. 498, no. 7453, pp. 185–189, Jun. 2013. DOI: `10.1038/nature12295`.

[41] K. Davies, "Trajectory estimation of cheetah tail motions", 2020.

[42] T. Ismaila, R. Akmeliawati and M. J. E. Salami, "Artificial intelligent based friction modelling and compensation in motion control system", *Advances in Mechatronics*, Aug. 2011. DOI: `10.5772/23432`. [Online]. Available: `https://www.researchgate.net/publication/221915347_Artificial_Intelligent_Based_Friction_Modelling_and_Compensation_in_Motion_Control_System`.

[43] *Solve nonlinear least-squares (nonlinear data-fitting) problems - matlab lsqnonlin.* [Online]. Available: `https://www.mathworks.com/help/optim/ug/lsqnonlin.html`.

[44] *Friction in contact between moving bodies - matlab.* [Online]. Available: `https://www.mathworks.com/help/simscape/ref/translationalfriction.html`.

[45] M. I. Yuce and D. A. Kareem, "A numerical analysis of fluid flow around circular and square cylinders", *Journal - American Water Works Association*, vol. 108, E546–E554, 10 Oct. 2016, ISSN: 15518833. DOI: `10.5942/JAWWA.2016.108.0141`.

[46] thang010146, *(82) 2 dof spherical connection control 6 - youtube*, 2015. [Online]. Available: `https://www.youtube.com/watch?v=EvALwzHBlUQ&t=4s&ab_channel=thang010146`.

[47] M. Huang, W. Wang and J. Li, "Analysis and verification of aerodynamic characteristics of tianwen-1 mars parachute", *Space: Science and Technology*, vol. 2022, Jan. 2022. DOI: `10.34133/2022/9805457`.

[48] Traxxas, *Traxxas stampede rc monster truck*, Website. [Online]. Available: `https://traxxas.com/products/models/electric/base-stampede-36054-4`.

[49] *Lipo.* [Online]. Available: `https://www.rclipo.co.za/lipo/results,1-0`.

[50] *Cli380.* [Online]. Available: `https://braeckman-modellbau.de/product_info.php?products_id=14&language=en`.

[51] *Hobbywing xerun 4274sd brushless motor g3 2250kv - boyztoyz.* [Online]. Available: `https://boyztoyz.co.za/product/hobbywing-xerun-4274sd-brushless-motor-g3-2250kv/`.

[52] *C94-m8p | u-blox.* [Online]. Available: `https://www.u-blox.com/en/product/c94-m8p`.

[53] *Adafruit 9-dof absolute orientation imu fusion breakout - bno055 - micro robotics.* [Online]. Available: `https://www.robotics.org.za/AF2472`.

[54] *Sd-25b-12 mean well.* [Online]. Available: `http://surl.li/jzgac`.

[55] *Teensy® 4.0.* [Online]. Available: `https://www.pjrc.com/store/teensy40.html`.

[56] *Aeat-6010-a06 | broadcom absolute mechanical rotary encoder with a 6 mm plain shaft (not indexed), screw mount | rs.* [Online]. Available: `http://surl.li/jzfzt`.

[57] *Amazon.com: Jetson nano 4gb ram 16g emmc onboard for ai robotics machine learning (heat sink version) : Electronics.* [Online]. Available: `http://surl.li/jzfzi`.

[58] *Digi international xbee-pro s2c rf transceiver module 2.4ghz zigbee xbp24cauit-001 xbee | rs.* [Online]. Available: `http://surl.li/jzfyq`.

[59] R. M. Murray, Z. Li and S. S. Sastry, "A mathematical introduction to robotic manipulation", *A Mathematical Introduction to Robotic Manipulation*, pp. 1–456, Jan. 1994. DOI: `10.1201/9781315136370/MATHEMATICAL-INTRODUCTION-ROBOTIC-MANIPULATION-RICHARD-MURRAY-ZEXIANG-LI-SHANKAR-SASTRY`.

Chapter 9: Appendix A

9.1 Euler Lagrangian Theory

The Euler Lagrangian model development is textbook theory. This section details a brief overview of the theory. The model is developed using the Manipulator Equation as shown below in Equation 9.1. In this equation, the M is the mass matrix, the C is the Coriolis matrix, the G is the gravitational matrix, the τ is the input torque matrix, the Q is the external force matrix, and the $A^T\lambda$ is the constraint matrix.

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \ddot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \tau + \mathbf{Q} + \mathbf{A}^{\mathbf{T}}\lambda \tag{9.1}$$

The calculating of the energies is the basis of the model. The kinetic energy for each body in the model is given by the base Equation 9.2 and is expanded to translational and rotational forms in Equation 9.3.

$$\mathbf{E} = \frac{1}{2}m\mathbf{v}^2 = \frac{1}{2}\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \cdot m \cdot \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \tag{9.2}$$

$$T = \sum_{i=1}^{n} (T_{T_i} + T_{R_i}) = \frac{1}{2}\dot{\mathbf{r}}^{\mathbf{T}} m \dot{\mathbf{r}} + \mathbf{\frac{1}{2}}\omega_{\mathbf{B}}^{\mathbf{T}}\mathbf{I}_{\mathbf{B}}\omega_{\mathbf{B}} \tag{9.3}$$

In this equation, the inertia is made up of the sum of all the inertias that are dependent on the rotation of the model. The mass is given depending on the system. The translational velocity needs to be calculated, and this begins by determining the position of the body in the model, given by Equation 9.4.

$$\mathbf{r}_i = Rot^T \cdot \begin{bmatrix} x_i & y_i & z_i \end{bmatrix} \tag{9.4}$$

With the position vectors of each body defined, the velocity vectors are next calculated. This is done by taking the time derivative of the position vectors and ends up being the Jacobian of the position vector with respect to the state variables multiplied by the time derivative of the state variables. The equation can be seen in Equation 9.5.

$$\dot{\mathbf{r}} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}}\dot{\mathbf{q}} \tag{9.5}$$

The velocity vectors, as well as the masses obtained of each body, are then plugged into the translational energy half of the kinetic energy equation, as seen in Equation 9.3. To get the angular velocity vectors of the body in the body's axis, the angular velocity needs to be transformed by the rotation sequences of the body. This gets more complicated the more rotation sequences are involved, however, a simple method was used by [59] which uses something called the skew symmetric property. The skew symmetric is defined in Equation 9.6 where the output of the skew symmetric provides a square matrix which has the angular velocity in that body's frame in the various rows and columns as seen in the equation.

$$[\omega_{\mathbf{B}}]_\times = [\mathbf{R}^{\mathbf{I}}_{\mathbf{B}}]^{\mathbf{T}}\dot{\mathbf{R}}^{\mathbf{I}}_{\mathbf{B}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \quad \tilde{\omega}_{\mathbf{B}} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix} \tag{9.6}$$

With the kinetic energy calculated, the mass matrix could be calculated by taking the Hessian of the kinetic energy, which is seen in 9.7. The mass matrix is then multiplied by the time derivative of the state variables with the Jacobian of the kinetic energy to get the Coriolis matrix. This is seen in 9.8.

$$\mathbf{M}_{\mathbf{i},\mathbf{j}} = \frac{\partial T}{\partial \dot{\mathbf{q}}_{\mathbf{i}} \partial \dot{\mathbf{q}}_j} \tag{9.7}$$

$$\mathbf{C} = \dot{\mathbf{M}}\dot{\mathbf{q}} - \frac{\partial T}{\partial \mathbf{q}} \tag{9.8}$$

The next step is to get the gravitational matrix. This is done by taking the partial differential of the potential energy with respect to the state variables. This is seen in Equation 9.9. The potential energy is the sum of the gravitational potential energy of each body. This can be seen in 9.10.

$$\mathbf{G} = \frac{\partial V}{\partial \mathbf{q}} \tag{9.9}$$

$$V = \sum_{i=1}^{n} m_i g h_i \tag{9.10}$$

Looking back at the manipulator equation, Equation 9.1, the left-hand side had now been calculated. The right-hand side of the equation is the input torques, the external forces and the constraints. The input torques for this research's model was just the torque, T, applied to the motor in the direction of the state variable, $\vec{u}$. Equation 9.11 shows this relation for the torque applied by the motor multiplied by the direction applied, the state vector ϕ_2.

$$\tau = \sum_{i=1}^{n} (T_i \cdot \mathbf{u}) = T \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{9.11}$$

The external forces were then calculated. These are forces applied to the body by the environment. These forces were calculated by taking the partial differential of the position vector with respect to the state variables and having it dot multiply by the force vector and can be seen in 9.12.

$$\mathbf{Q_i} = \sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f_j} \cdot \frac{\partial \mathbf{r_j}}{\partial \mathbf{q_i}} \tag{9.12}$$

With all these terms determined, they can be put into whichever model environment is being used for the simulation. For this research, MATLAB's Simulink environment was used.

Chapter 10: Appendix B

10.1 Seeds for the lsqnonlin function

These are the initial guesses for the static tests at varying degrees as described in Chapter 4 in section 4.7.

Table 10.1: Variables used in the lsqnonlin function with their initial conditions and upper and lower bounds for the three tail at a 15° static position for varying winds speed of 10, 20, and $30m/s$.

Variable	**Conditions**	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
		$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$
C_D	x_0	1.1	0.95	1.0	1.6	1.3	1	1.2	1.0	0.9
	lb	0								
	ub	3								
C_L	x_0	0	0.02	0.04	0.04	0.02	0.14	0.2	0.23	0.16
	lb	0								
	ub	3								
ψ_3	x_0	0.01	0.05	0.05	0	0.01	0.05	0.01	0.04	0.06
	lb	-0.12								
	ub	0.12								
I_{in}	x_0	2.8	7.5	17.5	2.4	7.5	14	2.75	9	17
	lb	0								
	ub	30								

Table 10.2: Variables used in the lsqnonlin function with their initial conditions and upper and lower bounds for the three tail at a 30° static position for varying winds speed of 10, 20, and $30m/s$.

Variable	**Conditions**	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
		$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$
C_D	x_0	0.83	0.6	0.7	1.6	1.0	0.8	1.2	1.1	1.2
	lb	0								
	ub	3								
C_L	x_0	0.4	0.4	0.35	0.04	0.2	0.34	0	0.2	0.4
	lb	0								
	ub	3								
ψ_3	x_0	0.1	0.05	0.1	0	0.01	0	0.01	0.01	0
	lb	-0.15								
	ub	0.15								
I_{in}	x_0	0.65	4	12	2.4	5	13	2.75	5	11.5
	lb	0								
	ub	30								

Table 10.3: Variables used in the lsqnonlin function with their initial conditions and upper and lower bounds for the three tail at a 45° static position for varying winds speed of 10, 20, and $30m/s$.

Variable	**Conditions**	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
		$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$
C_D	x_0	0.4	0.3	0.6	1.1	0.8	0.71	0.8	0.95	1
	lb	0								
	ub	3								
C_L	x_0	0.4	0.5	0.35	0.4	0.52	0.41	0	0.46	0.4
	lb	0								
	ub	3								
ψ_3	x_0	0.1	0	0.3	0	0	0.3	0.01	0.05	0.04
	lb	-0.35								
	ub	0.35								
I_{in}	x_0	0.4	6	7.5	0.5	2	7	0.5	2.4	7.5
	lb	-10								
	ub	30								

Table 10.4: Variables used in the lsqnonlin function with their initial conditions and upper and lower bounds for the three tail at a 60° static position for varying winds speed of 10, 20, and $30m/s$.

Variable	**Conditions**	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
		$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$
C_D	x_0	0	0.15	0	0	0	0.1	0	0.5	0.6
	lb	0								
	ub	3								
C_L	x_0	0.1	0.2	0	0.1	0	0.1	0	0.2	0.2
	lb	0								
	ub	3								
ψ_3	x_0	0.1	0	0	0	0	0	0.01	0.15	0.3
	lb	-0.35								
	ub	0.35								
I_{in}	x_0	1.5	0.7	2	1.5	1.6	2	1.5	0.5	1.5
	lb	-10								
	ub	30								

Table 10.5: Variables used in the lsqnonlin function with their initial conditions and upper and lower bounds for the three tail at a 75° static position for varying winds speed of 10, 20, and $30m/s$.

Variable	**Conditions**	**Tail 1 Axis**			**Tail 2 Axis**			**Tail 3 Axis**		
		$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$	$10\frac{m}{s}$	$20\frac{m}{s}$	$30\frac{m}{s}$
C_D	x_0	1.7	0.35	0.15	1	0.4	0.16	1.7	0.358	0.16
	lb	0								
	ub	3								
C_L	x_0	0.2	0.16	0.06	0.4	0.25	0.08	0.1	0.154	0.1
	lb	0								
	ub	3								
ψ_3	x_0	0.12	0.35	0.35	0	0	0.1	0.17	0.35	0.35
	lb	-0.35								
	ub	0.35								
I_{in}	x_0	2.1	2.5	2.46	1.5	1.5	1.5	2.1	2.5	2.5
	lb	-10								
	ub	30								

10.2 Results of Optimisation for Static Tests

Table 10.6: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	0.950	0.042	0.000	1.962	4.587
	2	0.951	0.023	0.000	1.964	7.248
	3	0.956	-0.019	0.000	1.974	25.416
	4	0.982	0.020	0.000	2.029	41.250
	5	1.010	0.007	0.000	2.084	23.982
20 m/s	1	0.914	0.060	-0.012	7.551	23.992
	2	0.923	0.064	-0.013	7.631	26.502
	3	0.915	0.052	-0.014	7.559	95.303
	4	0.902	0.059	-0.014	7.455	57.157
	5	-0.012	-0.072	0.000	0.012	18575.640
30 m/s	1	0.554	0.040	0.002	10.411	518.899
	2	0.625	0.194	-0.047	11.653	599.622
	3	0.599	0.105	-0.035	11.211	2455.791
	4	0.877	0.065	-0.027	16.310	17.627
	5	0.876	0.064	-0.028	16.279	15.749

Table 10.7: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 2's major axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\boldsymbol{\psi_3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.630	-0.053	-0.039	2.583	27.305
	2	1.602	-0.055	-0.033	2.541	21.419
	3	1.596	-0.056	-0.041	2.529	29.555
	4	1.619	-0.054	-0.034	2.568	19.661
	5	1.603	-0.055	-0.030	2.544	16.155
20 m/s	1	1.384	0.048	-0.038	8.786	2.383
	2	1.326	0.027	-0.042	8.419	22.653
	3	1.410	0.047	-0.039	8.950	2.702
	4	1.401	0.058	-0.034	8.903	3.967
	5	1.407	0.058	-0.039	8.930	3.695
30 m/s	1	1.135	0.054	-0.006	16.307	39.188
	2	1.141	0.052	-0.007	16.385	38.437
	3	1.139	0.066	-0.004	16.369	34.628
	4	1.113	0.065	-0.010	15.994	26.449
	5	1.128	0.050	-0.005	16.201	45.808

Table 10.8: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 2's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.208	-0.071	-0.005	1.926	6.525
	2	1.226	-0.007	-0.005	1.955	6.310
	3	1.222	-0.005	0.000	1.948	4.957
	4	1.255	0.007	-0.002	2.001	5.367
	5	1.235	-0.037	-0.004	1.969	4.677
20 m/s	1	1.003	0.019	-0.016	6.416	15.390
	2	1.021	0.005	-0.012	6.527	4.218
	3	1.059	0.025	-0.012	6.768	36.404
	4	1.048	0.005	-0.014	6.698	7.701
	5	1.040	0.024	-0.018	6.645	3.681
30 m/s	1	0.761	0.002	0.008	11.007	115.613
	2	0.703	0.001	0.027	10.182	31.160
	3	0.706	0.022	0.018	10.224	10.468
	4	1.082	0.071	-0.003	15.562	54.007
	5	1.085	0.068	-0.004	15.594	31.091

Table 10.9: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 3's major axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\psi_\mathbf{3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.783	-0.053	-0.015	2.147	6.843
	2	1.765	-0.067	-0.016	2.127	6.113
	3	1.795	-0.101	-0.018	2.162	6.511
	4	1.797	-0.094	-0.015	2.165	5.456
	5	1.780	-0.079	-0.014	2.145	6.601
20 m/s	1	1.538	-0.031	-0.006	7.425	17.277
	2	1.531	-0.013	-0.006	7.394	12.275
	3	1.514	0.012	-0.012	7.311	4.195
	4	1.535	-0.017	-0.013	7.410	4.314
	5	1.508	0.001	-0.005	7.286	23.257
30 m/s	1	1.470	0.013	0.008	15.986	30.369
	2	1.476	0.027	0.012	16.050	37.334
	3	1.478	0.034	0.012	16.063	25.848
	4	1.465	0.027	0.011	15.930	26.541
	5	1.487	0.018	0.014	16.167	24.357

Table 10.10: Static Results for 0° Angle of Attack after optimisation of the model parameters. The results are for the tail 3's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	2.142	0.111	-0.014	2.574	18.647
	2	2.076	-0.138	-0.012	2.497	26.840
	3	2.117	-0.023	-0.016	2.545	20.941
	4	2.113	0.001	-0.015	2.541	30.515
	5	2.128	-0.041	-0.016	2.558	24.298
20 m/s	1	1.479	-0.017	-0.027	7.136	4.287
	2	1.473	-0.034	-0.024	7.110	7.144
	3	1.497	-0.032	-0.023	7.228	3.330
	4	1.470	-0.032	-0.026	7.095	4.184
	5	1.474	-0.038	-0.028	7.112	2.372
30 m/s	1	1.069	0.001	-0.022	11.682	17.087
	2	0.995	0.004	0.000	10.900	23.470
	3	0.986	0.000	-0.009	10.806	24.854
	4	1.576	0.041	-0.002	17.121	14.620
	5	1.568	0.053	-0.003	17.039	19.354

Table 10.11: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's major axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.065	0.165	-0.071	1.674	2.211
	2	1.054	0.184	-0.061	1.669	3.329
	3	1.046	0.172	-0.081	1.635	2.532
	4	1.066	0.191	-0.086	1.680	1.580
	5	1.057	0.186	-0.065	1.673	3.204
20 m/s	1	0.923	0.191	-0.074	7.202	3.958
	2	0.916	0.178	-0.077	7.113	4.044
	3	0.935	0.167	-0.082	7.223	3.067
	4	0.914	0.173	-0.075	7.088	10.510
	5	0.892	0.191	-0.081	6.948	6.646
30 m/s	1	0.896	0.215	-0.075	16.485	56.627
	2	0.896	0.206	-0.067	16.486	63.404
	3	0.894	0.213	-0.073	16.459	55.497
	4	0.903	0.206	-0.072	16.578	61.745
	5	0.900	0.206	-0.074	16.520	49.002

Table 10.12: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	0.840	0.201	0.092	1.243	1.451
	2	0.829	0.135	0.077	1.195	1.354
	3	0.818	0.119	0.074	1.167	1.424
	4	0.845	0.132	0.062	1.234	1.379
	5	0.859	0.144	0.040	1.275	1.400
20 m/s	1	1.006	0.214	-0.014	7.981	5.846
	2	0.869	0.168	-0.017	6.808	1687.114
	3	1.015	0.223	-0.014	8.068	7.381
	4	1.006	0.223	-0.012	7.997	5.709
	5	1.012	0.220	-0.012	8.038	17.287
30 m/s	1	0.492	0.134	0.091	8.976	27810.792
	2	0.475	0.108	0.015	8.701	17.865
	3	0.638	0.151	-0.037	11.763	138.300
	4	0.777	0.177	-0.047	14.324	20.579
	5	0.776	0.184	-0.049	14.327	13.957

Table 10.13: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 2's major axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\boldsymbol{\psi_3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.352	0.131	-0.037	1.571	11.363
	2	1.273	0.148	-0.033	1.458	11.619
	3	1.304	0.152	-0.047	1.502	11.129
	4	1.288	0.134	-0.042	1.473	11.821
	5	1.258	0.119	-0.038	1.422	10.939
20 m/s	1	1.197	0.175	-0.049	7.093	102.000
	2	1.211	0.179	-0.056	7.169	123.714
	3	1.201	0.172	-0.056	7.099	82.800
	4	1.198	0.173	-0.057	7.082	110.978
	5	1.200	0.173	-0.055	7.096	94.683
30 m/s	1	0.987	0.208	-0.011	13.984	23.874
	2	0.982	0.209	-0.010	13.922	23.477
	3	0.978	0.196	-0.013	13.817	26.635
	4	0.997	0.201	-0.014	14.100	31.896
	5	0.996	0.200	-0.012	14.073	27.165

Table 10.14: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 2's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\boldsymbol{\psi_3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	0.956	0.196	0.053	0.992	2.065
	2	0.998	0.168	0.034	1.050	3.219
	3	0.921	0.070	0.016	0.896	20.524
	4	0.951	0.200	0.059	0.984	2.721
	5	0.972	0.093	0.015	0.983	3.274
20 m/s	1	0.904	0.137	-0.020	5.281	4.499
	2	0.908	0.156	-0.015	5.340	3.669
	3	0.901	0.170	-0.016	5.317	4.004
	4	0.929	0.161	-0.009	5.472	4.598
	5	0.923	0.153	-0.018	5.422	3.497
30 m/s	1	0.598	0.098	0.049	8.226	83.651
	2	0.654	0.120	0.033	9.098	99.691
	3	0.788	0.069	0.117	10.469	279426.170
	4	0.769	0.152	-0.007	10.807	21.161
	5	0.631	0.114	0.032	8.760	53.932

Table 10.15: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 3's major axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\psi_\mathbf{3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.458	0.168	0.001	1.210	2.765
	2	1.487	0.136	-0.004	1.233	2.543
	3	1.460	0.144	-0.008	1.205	3.034
	4	1.416	0.174	-0.007	1.164	2.300
	5	1.428	0.176	-0.031	1.175	2.040
20 m/s	1	1.344	0.169	-0.004	5.971	4.114
	2	1.331	0.185	-0.006	5.931	3.040
	3	1.354	0.188	-0.010	6.040	4.384
	4	1.332	0.188	-0.013	5.937	2.742
	5	1.342	0.170	-0.017	5.957	3.615
30 m/s	1	1.339	0.211	0.006	14.184	14.801
	2	1.336	0.225	0.014	14.186	10.095
	3	1.327	0.222	0.009	14.087	12.814
	4	1.342	0.216	0.008	14.231	9.549
	5	1.310	0.231	0.007	13.936	18.492

Table 10.16: Static Results for 15° Angle of Attack after optimisation of the model parameters. The results are for the tail 3's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	$\boldsymbol{\psi_3}$	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	1.782	0.075	-0.035	1.549	7.344
	2	1.766	0.162	-0.019	1.561	6.993
	3	1.724	0.217	-0.022	1.530	8.209
	4	1.787	0.155	-0.028	1.581	7.323
	5	1.734	0.121	-0.008	1.513	7.398
20 m/s	1	0.108	0.000	0.120	0.063	30558.592
	2	1.346	0.142	-0.037	5.930	2.161
	3	1.365	0.138	-0.036	6.012	8.096
	4	1.349	0.157	-0.039	5.956	3.986
	5	1.336	0.144	-0.031	5.888	4.095
30 m/s	1	0.866	0.124	0.002	9.050	18.493
	2	1.049	0.186	-0.032	11.088	79.310
	3	1.053	0.190	-0.032	11.150	44.647
	4	1.046	0.184	-0.033	11.058	47.845
	5	1.044	0.183	-0.034	11.032	15.846

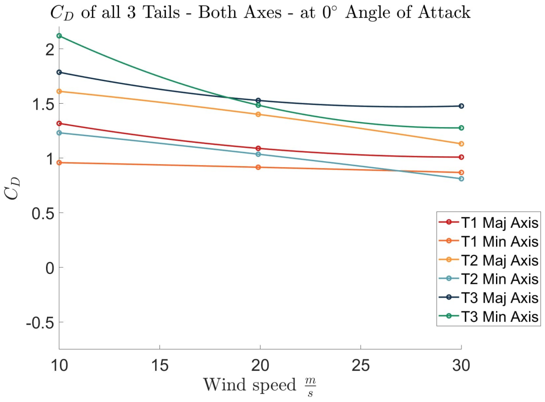
Table 10.17: Static Results for 30° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's major axis.

Wind Speed	Test No.	C_D	C_L	ψ_3	I_{in}	Resnorm
10 m/s	1	0.824	0.341	-0.120	0.677	5.721
	2	0.818	0.343	-0.120	0.669	3.651
	3	0.812	0.348	-0.120	0.664	2.886
	4	0.834	0.360	-0.120	0.712	2.322
	5	0.821	0.358	-0.120	0.688	3.151
20 m/s	1	0.843	0.331	-0.146	5.722	8.024
	2	0.830	0.339	-0.150	5.650	6.748
	3	0.823	0.344	-0.150	5.625	6.528
	4	0.840	0.347	-0.146	5.764	6.240
	5	0.827	0.341	-0.150	5.643	7.681
30 m/s	1	0.843	0.331	-0.146	5.722	8.024
	2	0.830	0.339	-0.150	5.650	6.748
	3	0.823	0.344	-0.150	5.625	6.528
	4	0.840	0.347	-0.146	5.764	6.240
	5	0.827	0.341	-0.150	5.643	7.681

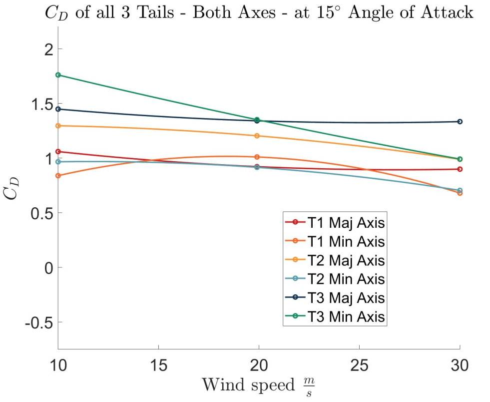
Table 10.18: Static Results for 30° Angle of Attack after optimisation of the model parameters. The results are for the tail 1's minor axis.

Wind Speed	**Test No.**	$\mathbf{C_D}$	$\mathbf{C_L}$	ψ_3	$\mathbf{I_{in}}$	**Resnorm**
10 m/s	1	0.805	0.000	0.120	0.316	1489.978
	2	0.715	0.292	0.148	0.432	7.832
	3	0.707	0.272	0.150	0.396	3.723
	4	0.717	0.279	0.150	0.420	5.861
	5	0.699	0.288	0.150	0.400	4.211
20 m/s	1	0.875	0.346	-0.010	6.267	9.028
	2	0.844	0.346	-0.002	6.057	7.103
	3	0.929	0.323	0.132	6.297	61375.788
	4	0.858	0.337	0.005	6.121	6.184
	5	0.854	0.348	-0.009	6.131	7.260
30 m/s	1	0.875	0.346	-0.010	6.267	9.028
	2	0.844	0.346	-0.002	6.057	7.103
	3	0.929	0.323	0.132	6.297	61375.788
	4	0.858	0.337	0.005	6.121	6.184
	5	0.854	0.348	-0.009	6.131	7.260

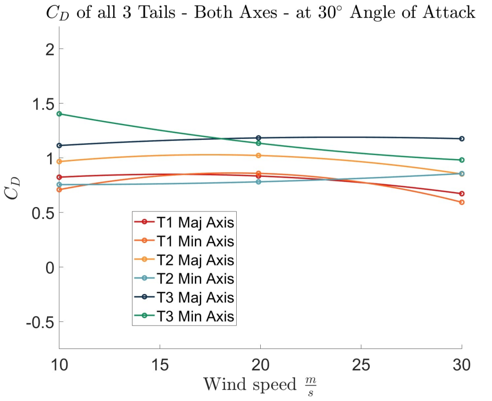
10.3 Final Drag and Lift Coefficients for Static Tests



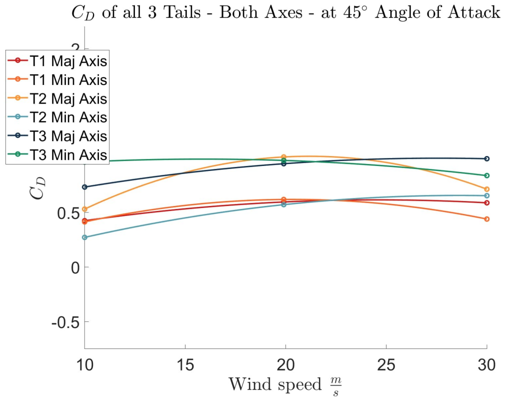
(a) 0° Angle of Attack



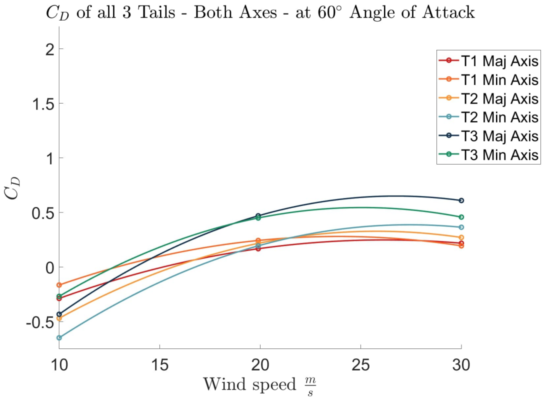
(b) 15° Angle of Attack



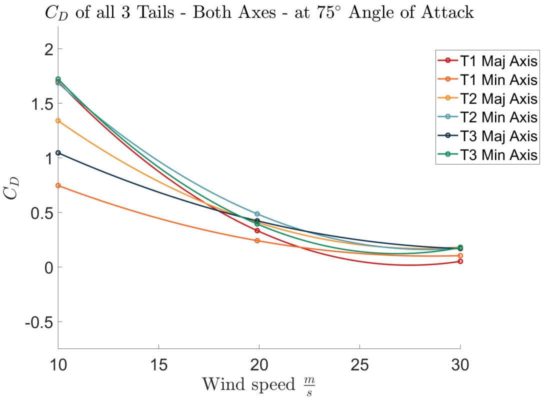
(c) 30° Angle of Attack



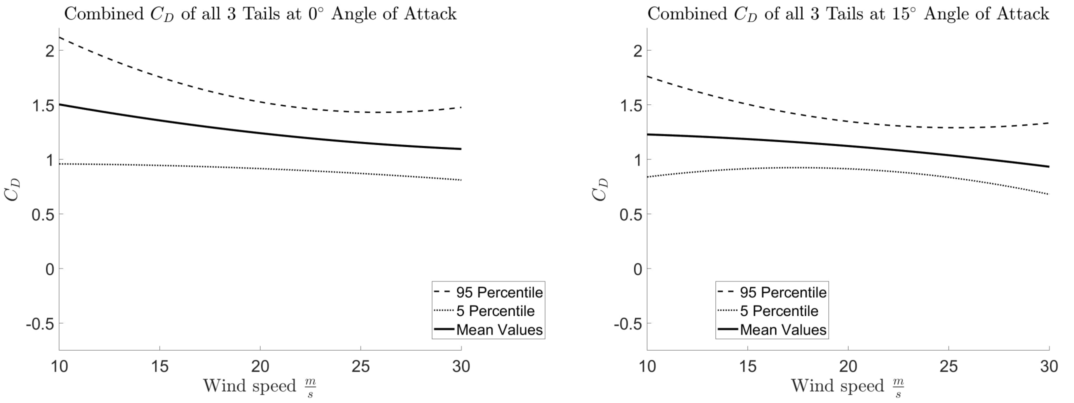
(d) 45° Angle of Attack



(e) 60° Angle of Attack

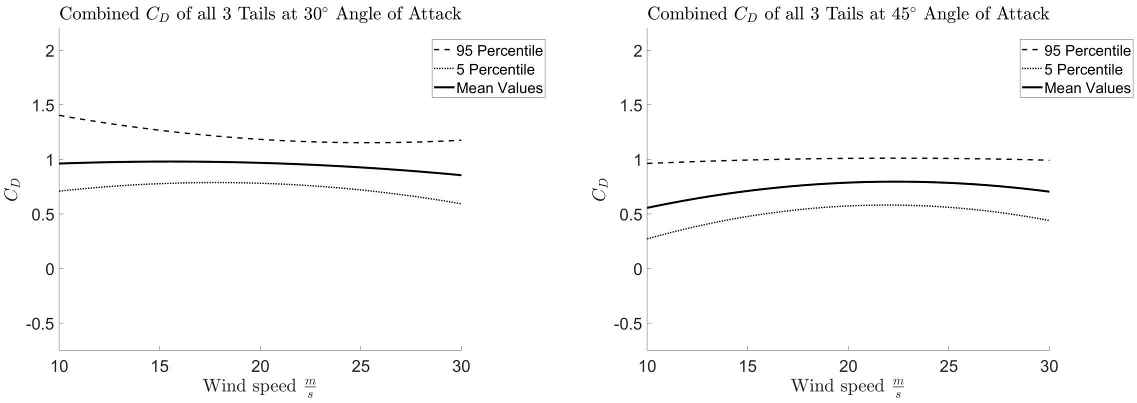


(f) 75° Angle of Attack



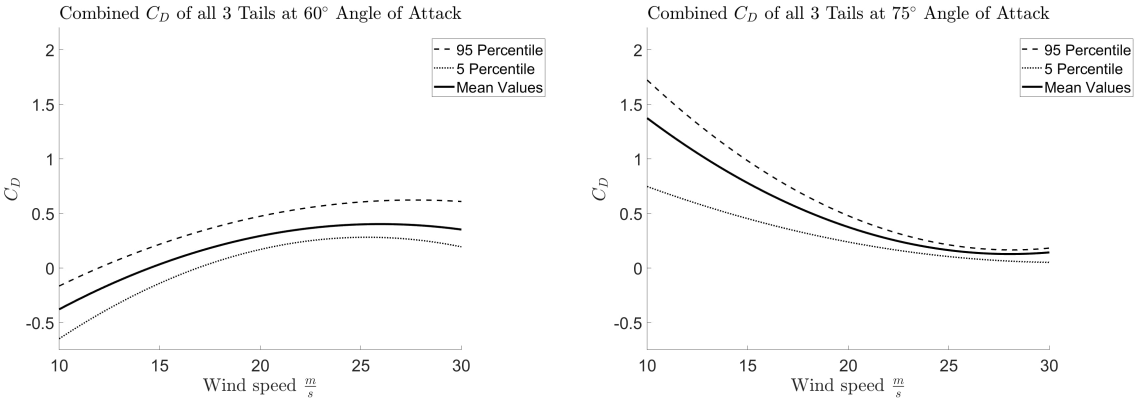
(a) 0° Angle of Attack

(b) 15° Angle of Attack



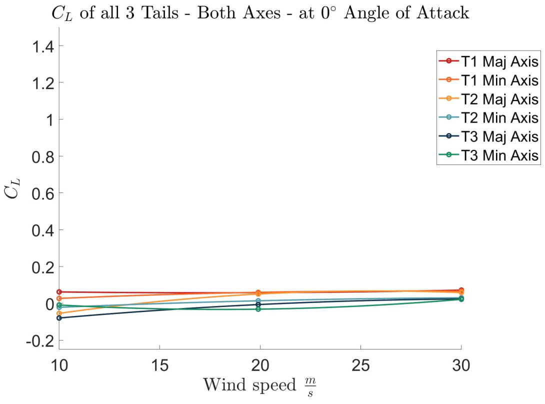
(c) 30° Angle of Attack

(d) 45° Angle of Attack

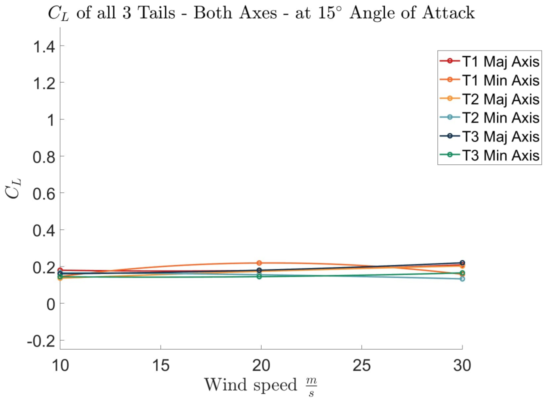


(e) 60° Angle of Attack

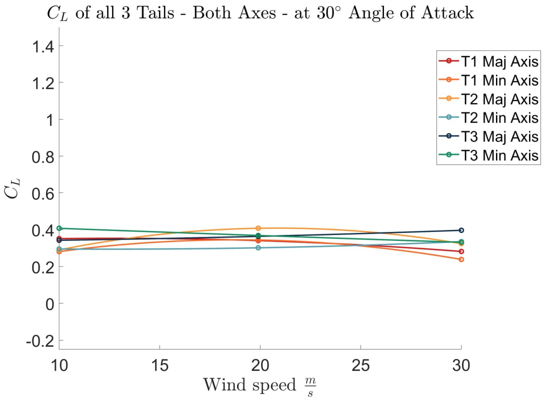
(f) 75° Angle of Attack



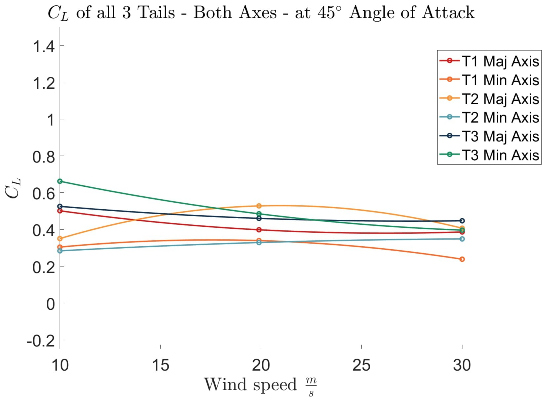
(a) 0° Angle of Attack



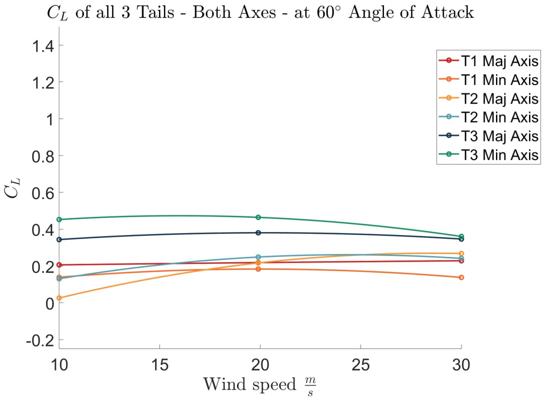
(b) 15° Angle of Attack



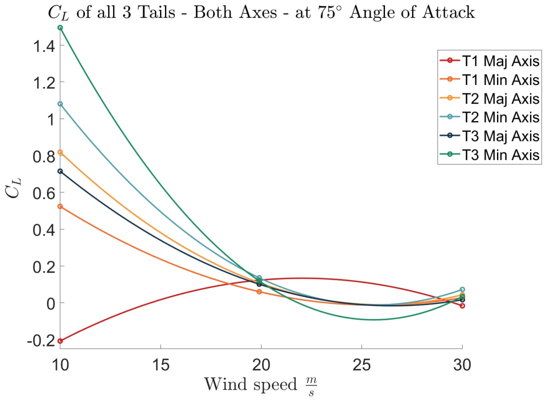
(c) 30° Angle of Attack



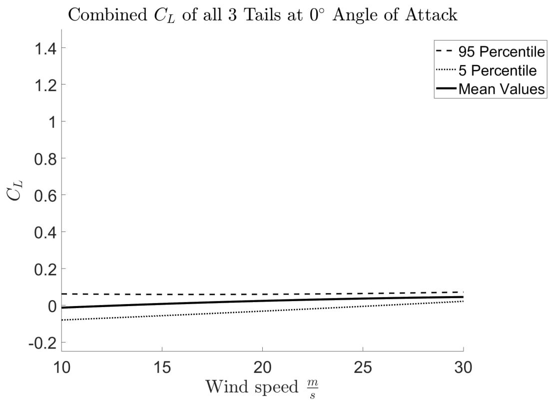
(d) 45° Angle of Attack



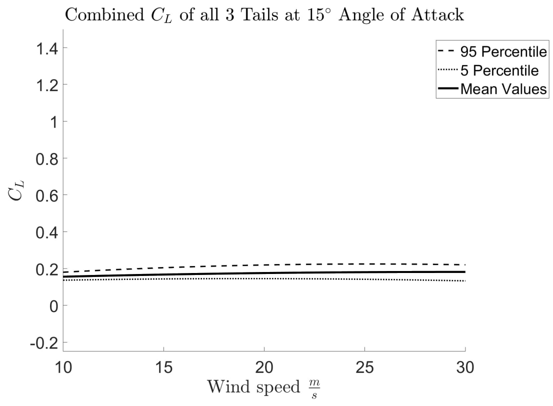
(e) 60° Angle of Attack



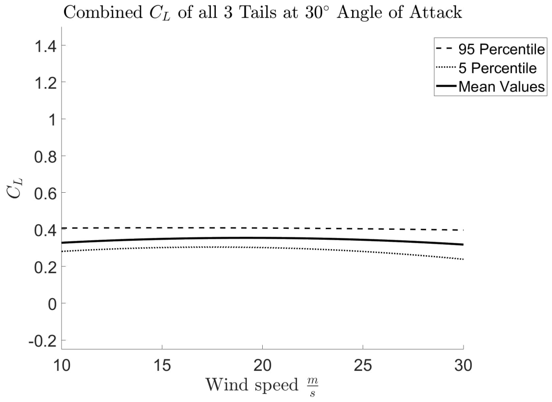
(f) 75° Angle of Attack



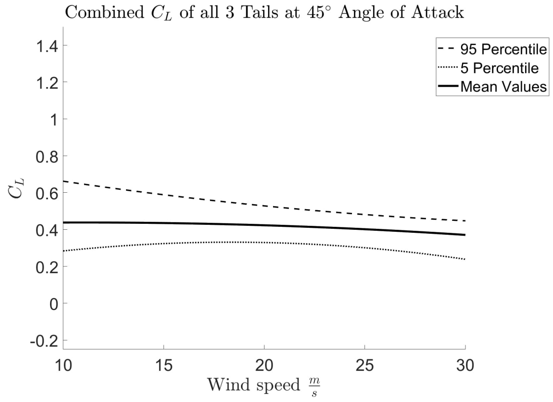
(a) 0° Angle of Attack



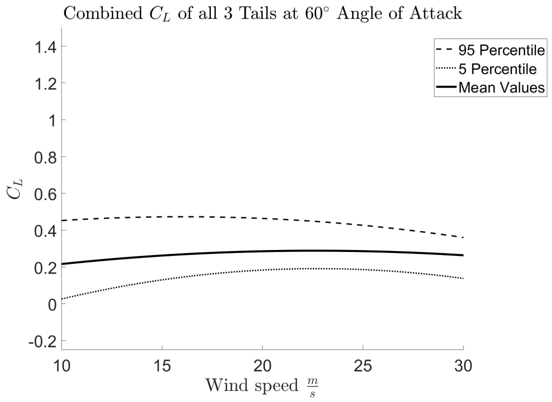
(b) 15° Angle of Attack



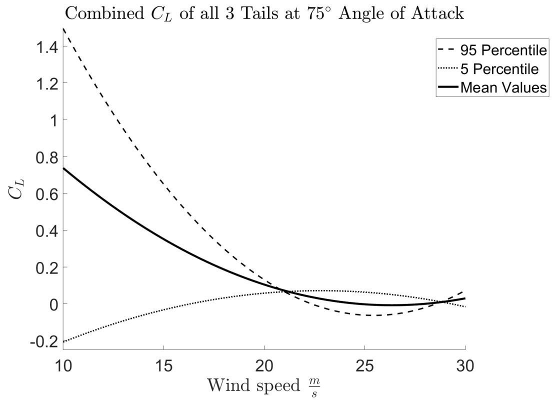
(c) 30° Angle of Attack



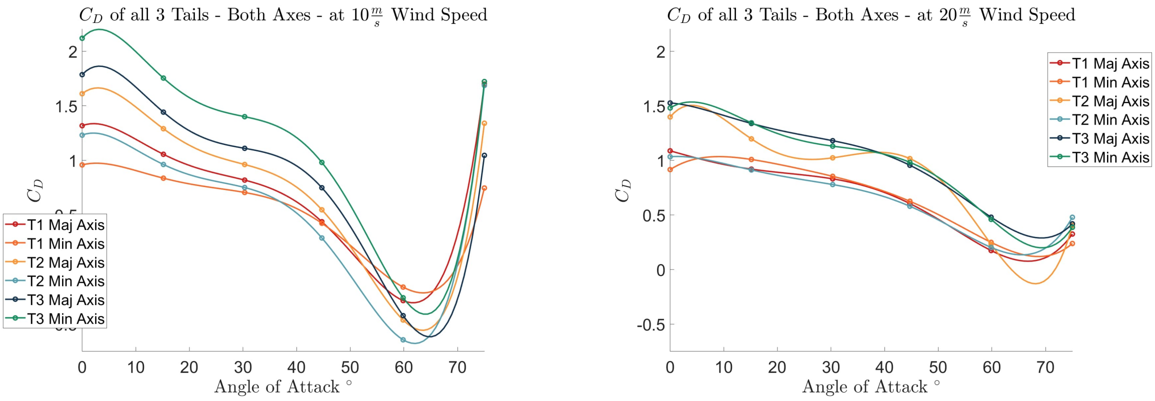
(d) 45° Angle of Attack



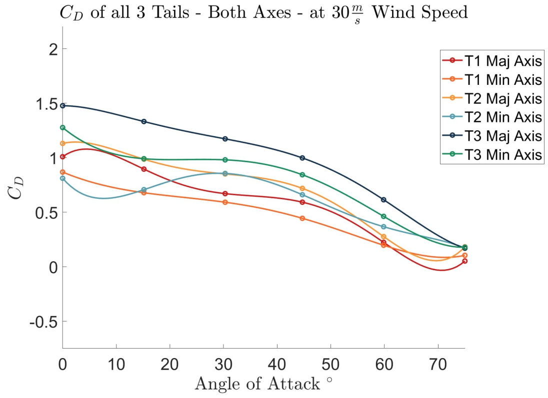
(e) 60° Angle of Attack



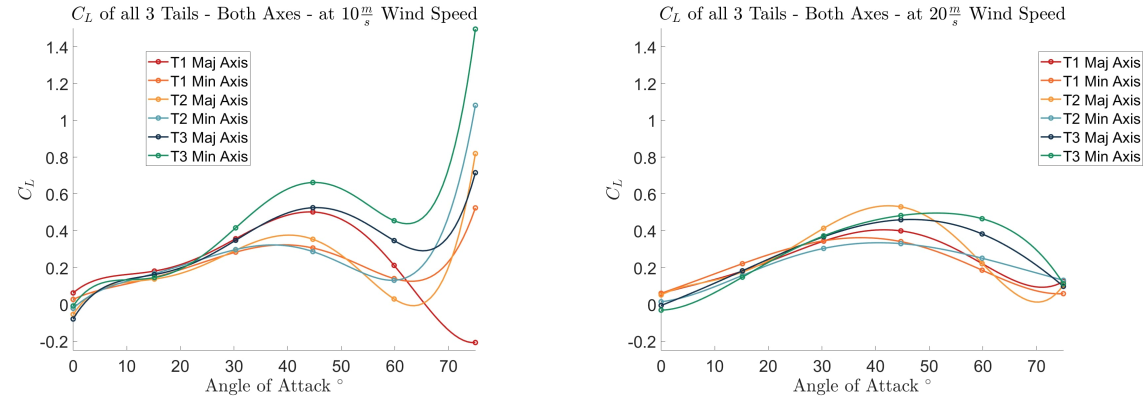
(f) 75° Angle of Attack



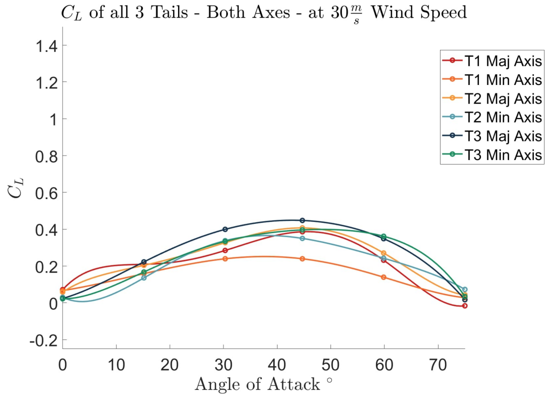
(a) 10m/s Wind speed **(b)** 20m/s Wind speed



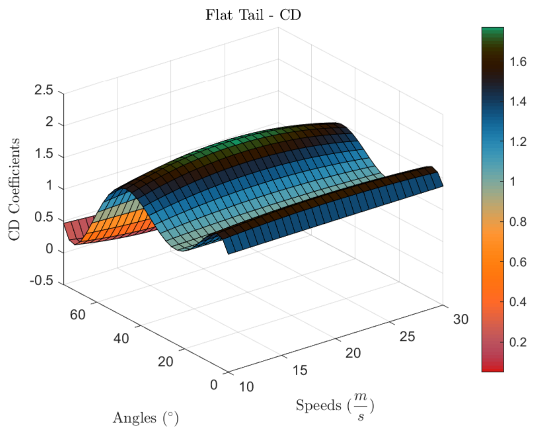
(c) 30m/s Wind speed



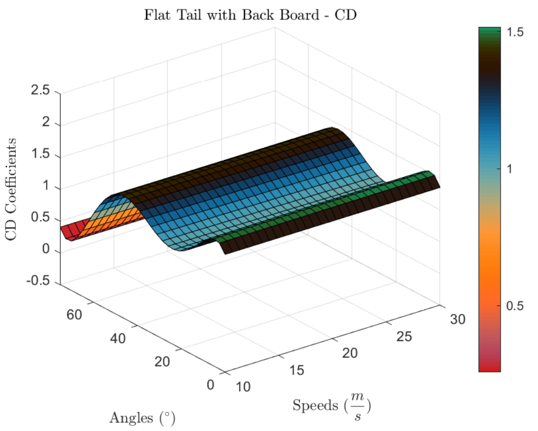
(a) 10m/s Wind speed **(b)** 20m/s Wind speed



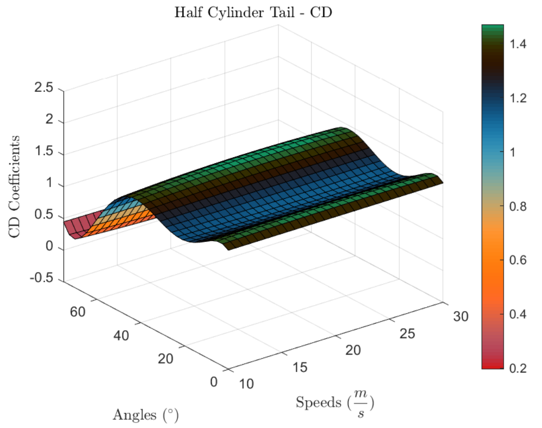
(c) 30m/s Wind speed



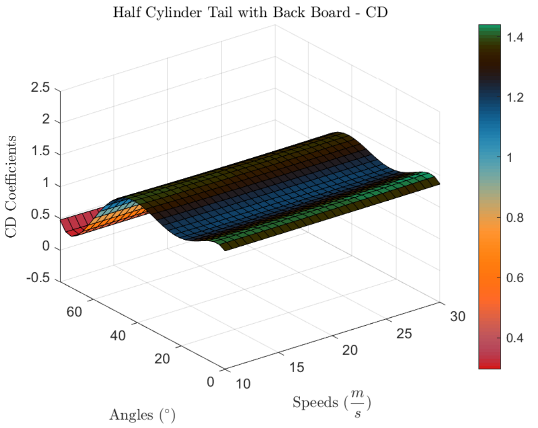
(a) Flat Tail Design



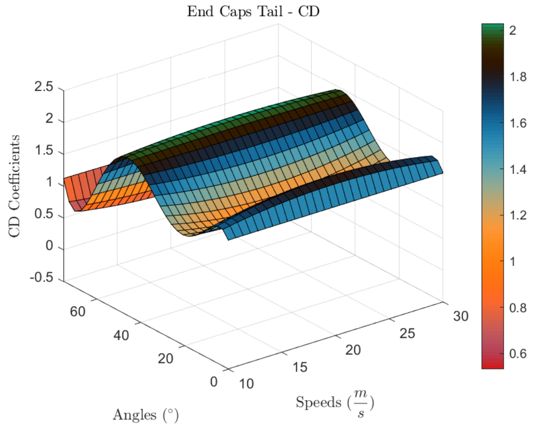
(b) Flat Tail with Backing Board Design



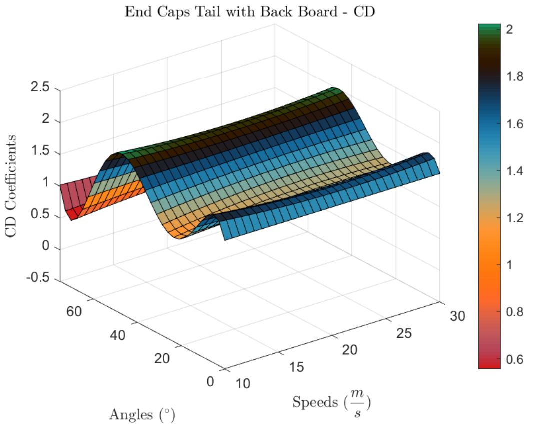
(c) Half Cylinder Tail Design



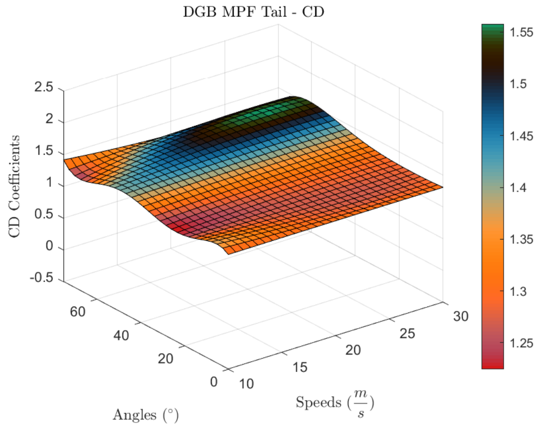
(d) Half Cylinder Tail with Backing Board Design



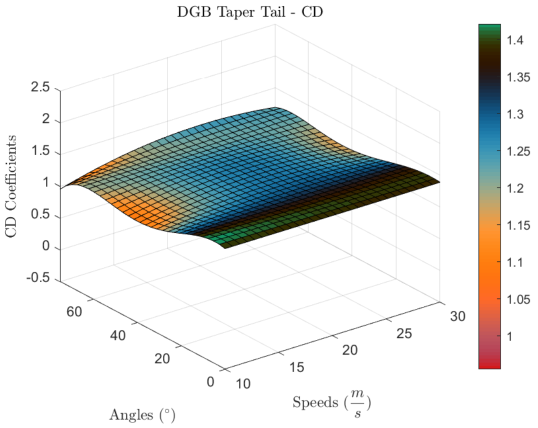
(e) End Caps Tail Design



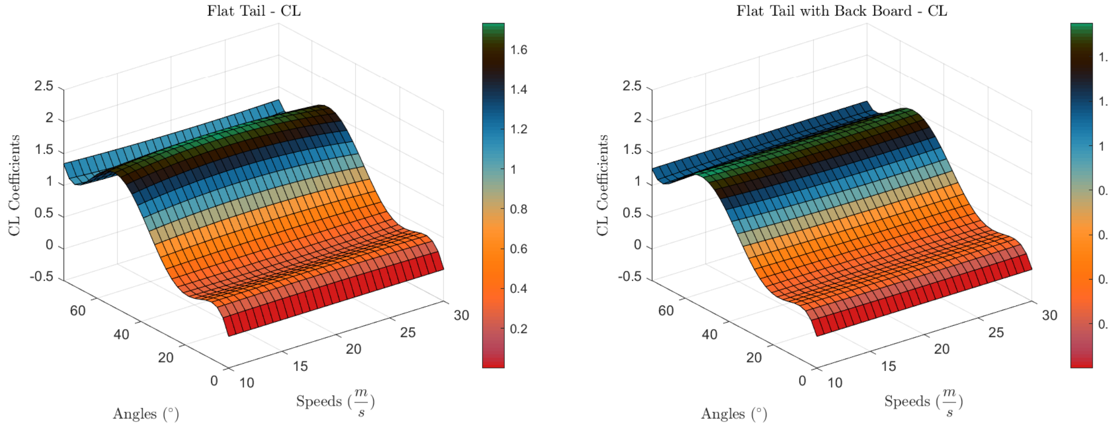
(f) End Caps Tail with Backing Board Design



(g) DGB MPF Tail Design

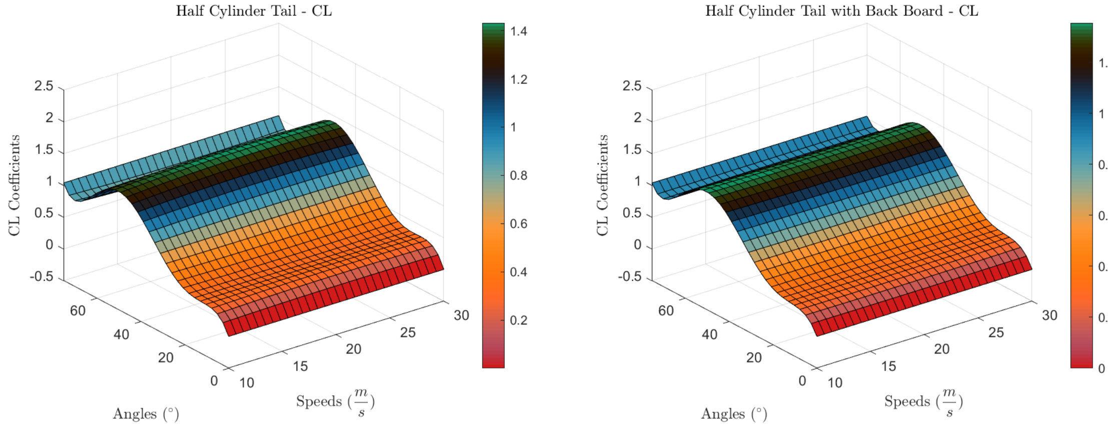


(h) DGB Taper Tail Design



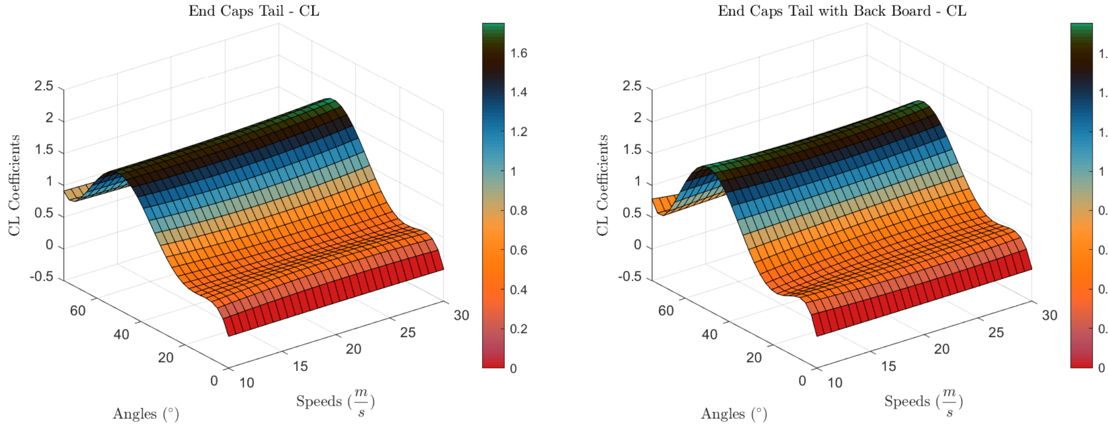
(a) Flat Tail Design

(b) Flat Tail with Backing Board Design



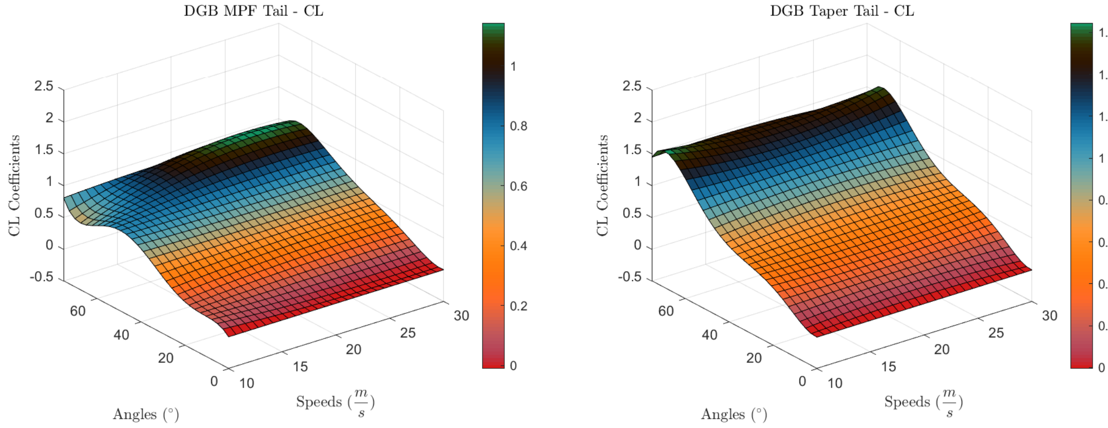
(c) Half Cylinder Tail Design

(d) Half Cylinder Tail with Backing Board Design



(e) End Caps Tail Design

(f) End Caps Tail with Backing Board Design



(g) DGB MPF Tail Design

(h) DGB Taper Tail Design

Chapter 11: Appendix C

11.1 Results of Optimisation for Dynamic Swing Tests

These are the results of the optimisation function that was run for the dynamic swing tests. The results measured the added mass constants and the resnorm for each test.

Table 11.1: Optimised added mass constants and resnorms for each tail swung in $10m/s$ wind speed. The weighted average of the added mass constants are also included.

Tail	**Variable**	**Swing** 1	2	3	4	5	**Weighted Average**
Tail 1 Maj	Added Mass	0.0201	0.0185	0.0163	0.0183	0.0157	0.02
	Resnorm	192.41	238.86	248.70	160.74	229.03	*N/A*
Tail 1 Min	Added Mass	0.0167	0.0060	0.0083	0.0101	0.0081	0.0097
	Resnorm	357.73	387.55	283.31	400.95	267.83	*N/A*
Tail 2 Maj	Added Mass	0.0480	0.0415	0.0462	0.0472	0.0451	0.0456
	Resnorm	253.27	245.09	219.63	214.23	271.18	*N/A*
Tail 2 Min	Added Mass	0.0399	0.0433	0.0348	0.0351	0.0382	0.0381
	Resnorm	380.19	303.16	263.15	286.44	265.10	*N/A*
Tail 3 Maj	Added Mass	0.0506	0.0531	0.0549	0.0521	0.0511	0.0524
	Resnorm	220.51	197.24	204.08	186.02	259.03	*N/A*
Tail 3 Min	Added Mass	0.0623	0.0597	0.0595	0.0554	0.0604	0.0592
	Resnorm	301.44	267.14	181.57	210.43	239.30	*N/A*

Table 11.2: Optimised added mass constants and resnorms for each tail swung in 20m/s wind speed. The weighted average of the added mass constants are also included.

Tail	**Variable**	**Swing** 1	2	3	4	5	**Weighted Average**
Tail 1 Maj	Added Mass	-0.032	-0.028	-0.002	-0.005	-0.012	0.0676
	Resnorm	1157.16	832.84	1218.92	1195.25	947.05	*N/A*
Tail 1 Min	Added Mass	-0.038	-0.016	-0.027	-0.047	-0.029	-0.033
	Resnorm	856.78	1396.16	963.00	756.85	742.62	*N/A*
Tail 2 Maj	Added Mass	0.0247	0.0247	0.0265	0.0190	0.0164	0.0220
	Resnorm	490.28	467.00	442.23	394.56	395.47	*N/A*
Tail 2 Min	Added Mass	0.0441	0.0383	0.0444	0.0420	0.5000	0.0460
	Resnorm	447.42	830.84	504.55	474.64	18093	*N/A*
Tail 3 Maj	Added Mass	-0.003	-0.003	-0.001	-0.001	-0.006	-0.003
	Resnorm	408.51	449.28	500.56	510.75	622.63	*N/A*
Tail 3 Min	Added Mass	0.0669	0.0670	0.0686	0.0688	0.0665	0.0676
	Resnorm	1079.70	795.04	820.92	783.50	914.72	*N/A*

11.2 Drag Coefficients from Simulated Swing Tests

These are the results of the drag coefficients for the simulated swing tests done for the aerodynamic robotic tail.

Table 11.3: Results from the simulated tail model for a wind speed of 5m/s with a Kp gain of 10.

Relative Velocity (m/s)	Test Conditions: Wind Speed = 5m/s, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
9	1.30	1.50	1.20	1.50	1.50
10	1.30	1.50	1.40	1.70	1.70
11	2.00	2.00	2.00	2.00	2.00
12	2.00	2.00	2.00	2.00	2.00
13	2.00	2.00	2.00	2.00	2.00
Resnorm	50.64	43.03	43.93	141.95	168.20

Table 11.4: Results from the simulated tail model for a wind speed of 7.5m/s with a Kp gain of 4.

Relative Velocity (m/s)	Test Conditions: Wind Speed = 5m/s, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
6	1.90	1.90	2.00	1.70	1.70
7	1.90	1.90	2.00	1.70	1.70
8	2.00	2.00	2.00	2.00	2.10
9	2.00	2.10	2.00	2.10	2.20
10	2.20	2.30	2.10	2.30	2.30
Resnorm	103.56	108.51	26.85	25.48	40.29

Table 11.5: Results from the simulated tail model for a wind speed of 7.5m/s with a Kp gain of 10.

Relative Velocity (m/s)	Test Conditions: Wind Speed = 5m/s, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
6	2.00	1.80	1.60	1.60	1.60
7	2.00	1.80	1.60	1.60	1.60
8	2.20	2.20	2.00	2.00	2.00
9	2.20	2.20	2.20	2.20	2.20
10	2.40	2.40	2.40	2.40	2.40
11	2.50	2.50	2.50	2.50	2.50
12	2.50	2.50	2.50	2.50	2.50
13	2.50	2.50	2.50	2.50	2.50
Resnorm	580.02	543.06	471.53	32.99	151.73

Table 11.6: Results from the simulated tail model for a wind speed of $10m/s$ with a Kp gain of 4.

Relative Velocity (m/s)	Test Conditions: Wind Speed = $5m/s$, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
9	1.30	1.50	1.20	1.50	1.50
10	1.30	1.50	1.40	1.70	1.70
11	2.00	2.00	2.00	2.00	2.00
12	2.00	2.00	2.00	2.00	2.00
13	2.00	2.00	2.00	2.00	2.00
Resnorm	50.64	43.03	43.93	141.95	168.20

Table 11.7: Results from the simulated tail model for a wind speed of $10m/s$ with a Kp gain of 10.

Relative Velocity (m/s)	Test Conditions: Wind Speed = $5m/s$, Kp = 4				
	Swing 1	Swing 2	Swing 3	Swing 4	Swing 5
8	1.50	1.50	1.50	1.50	1.50
9	1.30	1.30	1.30	1.30	1.30
10	1.30	1.30	1.30	1.30	1.30
11	2.00	2.00	2.00	2.00	2.00
12	2.00	2.20	2.20	2.20	2.20
13	2.00	2.20	2.20	2.20	2.20
14	2.10	2.10	2.10	2.40	2.40
15	2.20	2.20	2.20	2.50	2.50
Resnorm	108.43	101.19	338.68	29.18	43.99