

COVID-19 Impact Statement

Impact on the Research from COVID-19

In March 2020, experimental work was halted for 6 months due to the closure of the university facilities. This caused a major interruption to the experimental work, as setup for the twisted string actuator experiments in chapter 3 was already in progress, and setup for the payload stability experiments in chapter ?? was close to commencement. In October 2020 limited access was allowed into the lab, so experimental work could resume, albeit at a significantly slower rate of progress due to the access limitations. It was not until September 2021 that full access was restored, and by then, coupled with various logistics issues that were also the result of COVID-19, experimental progress was at least 6 months behind schedule. While the twisted string actuator experiments were completed in June 2021, there was still significant work required to complete the payload stability experiments. The essential experiments for this were completed by Christmas break in December 2021, and the thesis was submitted by the end of March 2022. However, the compressed timeframe meant that control issues with the robot tail that would have ideally been investigated further and corrected were instead overlooked in favour of sub-optimal but nevertheless usable results.

Planned Work not Undertaken due to COVID-19

The experimental work for the payload stability experiments was significantly curtailed due to the lack of time available and the slow progress due to limited lab hours and logistics issues. Initial plans were to experiment with the arm and tail in different y axis positions, hence the use of rails on the static rig. This would have given more credibility to the results of these experiments, showing the system working for a range of different mobile robot shapes and sizes. The ultimate goal of the research was to evolve from a static rig to a mobile robot in simulated real world conditions, which would have provided the most solid proof of the efficacy of using a robot tail for stability when carrying a payload.

Actions taken to Mitigate the effects of COVID-19 on the Research

During the university shutdown, efforts were redirected onto simulation research, however it was decided that all planned experimental work considered essential to complete the PhD to a doctoral standard would be completed, with time extensions sought if necessary (and in the end a 6 month extension was granted). In particular, work was done to improve and

characterise the control system for the twisted string actuator experiments using a simulated model, using a borrowed myRIO® to verify the LabVIEW™control system operated in the same fashion as the system designed in Simulink for the same purpose. ~~The multi-segment optimisation work in chapter ?? was conducted as an extra piece of research in Summer 2020 while waiting for the lab to reopen.~~

Robotic Tails for Enhanced Mobile Robot Performance

A thesis submitted to the University of Manchester for the degree of
Doctor of Philosophy
in the Faculty of Science and Engineering

2022

Damian Crosby
Department of Mechanical, Aerospace and Civil Engineering

Contents

Word Count: 25356

List of figures

List of tables

List of publications

icra

List of abbreviations

Abstract

Maintaining stability is of vital importance for mobile robots working in the field, where they may be in hazardous or confined environments, out of the reach of operators. Robotic tails have already been investigated for use in maintaining robot stability in a number of different scenarios, but the action of transporting a payload is as yet unexplored. This presents unique challenges for robot stability, due to the sudden shift in center of mass. Developing a robotic tail that can respond to this change and maintain robot stability would allow mobile robots to transport heavier and more unbalanced payloads in a safe and stable manner. This research project first investigated the existing uses of robotic tails in terrestrial mobile robots using selected literature from the first systematic search conducted in the field, finding a diverse set of tail functions that maintain stability or enhance the motion of mobile robots. The findings from this research, mainly the rarity of multi-segment tails, went on to inform the direction of later parts of the project. A potential actuation method for the tail joint based on the twisted string actuator (TSA) in an “antagonistic triad” configuration was designed and built, and successfully tested with robust control, able to accurately track a 2 degree of freedom joint trajectory within a range of $\pm 11^\circ$ to an accuracy of 1.8° . This twisted string actuator was then compared with alternative actuators to characterise its performance, which was found to be dependant on the actuator position. A simulation study was then conducted to investigate if there was any advantage to using multi-segment tails over a single segment tail when the segments were tracking the center of mass position trajectory of all of the segments combined. Two optimisation algorithms were used to minimise the velocity 2-norm and torque 2-norm of all the tail joints. Unfortunately, the results were inconclusive due to issues with one of the algorithms that prevented minimisation of the torque 2-norm, so a single segment design was eventually selected. A configurable payload, consisting of a container filled with eight cubes of various materials with different densities, was designed and constructed to be able to generate a variety of different simulated payloads with a range of mass and center of mass for later experiments. The configurable payload was mathematically defined, and test point sets were created from the extrema of the set of mass and center of mass that encompassed all possible configurations of the payload, as well as target mass and center of mass where the closest configuration was found using a search method. Finally, a static test rig was constructed which was able to measure stability using four load cells. A robot arm was then used to pick, hold, and place the configurable payload with and without a single segment 2 DOF tail with a weighted tip, which used the load cell data to minimise the value of the center of pressure by moving the tail. Overall, the research in this thesis demonstrates a clear improvement in stability of up to 87% when using a robot tail to carry a payload. This research lays the groundwork for the implementation of a robot tail on a mobile robot for simulated tasks involving transporting various payloads. It has also developed a novel new actuated universal joint based on the twisted string actuator, which has applications for robotic tails and beyond.

Declaration of originality

I hereby confirm that no portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright statement

- i The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.
- ii Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made *only* in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.
- iii The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.
- iv Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see <http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=24420>), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.library.manchester.ac.uk/about/regulations/>) and in The University’s policy on Presentation of Theses.

Acknowledgements

The journey to the completion of this PhD has been long and not without obstacles, especially due to the disruption caused by COVID-19. I wish to thank the people who mentored, assisted, and supported me along the way:

To my supervisor Andrew Weightman, and my co-supervisors Joaquin Carrasco and William Heath. Andy has provided invaluable guidance, inspiration and emotional support. He has pulled me up when I felt overwhelmed, and set me straight when I was overconfident. Coupled with a cheery disposition and good sense of humour, he has been an excellent supervisor. Joaquin Carrasco was essential in making sure any equations I wrote actually made sense, and filled in a lot of the gaps in my knowledge around control theory. He also provided the vital inspiration for the twisted string actuator control system, that allowed stable, robust control to be achieved.

To my masters supervisor, Dmitriy Makhnovskiy, who during his time at the university of Plymouth, initially inspired me to look at twisted string actuators for use in an actuated universal joint. To finally finish what I started nearly a decade ago, and get it submitted to a prestigious robotics conference, is an immense feeling of achievement.

To my friends and colleagues that I met along the way, that provided companionship and advice. Tapiro Väistö and Hang Gu were there for me in the early days, and the lunches we had together will not be forgotten. Lavanand Vengadasalam, Thomas Shearwood, Mobin Malik and others while working as a GTA for Andy and Mark Quinn, whose sharp wit and sarcasm made his classes very memorable. Alejandro Macario Rojas and Gunter Just, who were the only faces I saw for months when the lab reopened. And towards the end once the university had fully reopened, the rest of the lab group in J2, Muhammed Yilmaz, Lutong Li, Athia Haron, Harrison Mcaleese and others, who I wish the the best of luck on their on-going research. For the rest I have forgotten, I apologise!

To Stephen Burley for his help with sourcing and adjusting an instrument amplifier that was required for the load cells.

To the workshop staff, in particular Christopher Eaton and Stuart McIntyre, who were approachable and helpful with any requests for manufacturing or 3D printing.

To my former colleagues in the MTRL, Paul English, Daniel Wilson, Neil Irvine, Jacqueline Grant, Anastasia Vasileiou and others, I promise to come visit when I'm finished!

And of course, to my kind and supportive parents, Lynne and John, who on several occasions came and cleaned my house up, took me out to dinner, lifted my spirits and allowed me to rest. They also provided proofreading for my work, and let me stay with them for 6

weeks to focus on my thesis. Without their support, this would have been a lot more difficult.

And finally, to the takeaway chefs and Deliveroo drivers that sustained me in the final weeks of my PhD, when I really *really* couldn't be bothered to cook.

Here's to the next chapter of my life, wherever it may take me!

Chapter 1

Introduction

This chapter gives a brief overview of the use of mobile robots for transporting payloads in adverse and challenging environments. It then describes the problem of maintaining balance in mobile robots, and how the addition of a payload changes the balanced margins of the robot. It then proposes the use of a robotic tail for compensating for this change, and provides case studies from the animal kingdom that shows how tails are used for two different kinds of locomotion problems. It then outlines the aims and objectives of the thesis, and finally details the structure of the thesis.



Figure 1.1: Boston Dynamics Spot® working with the Royal Air Force. *Public Domain*

In recent years, there has been a significant increase in the use of mobile robots for industrial, military and service applications, such as moving stock around in warehouses, inspecting equipment in hazardous or remote environments, performing search and rescue operations or acting as infantry support. With advances in battery technology, material fabrication and onboard processing power, mobile robots have been able to fill an increasing number of roles. Modern legged robots such as the *Boston Dynamics* Spot® pictured in figure 1.1 have allowed mobile robots into less rigidly controlled environments, able to navigate rough terrain and stairs. Some of these robots are also incorporating robot arms or other mechanisms that allow them to pick up, carry, and set down payloads. This can be useful for a number of reasons, such as collecting samples from a hazardous environment, clearing away obstructions that impede progress, moving products in a warehouse or providing vital supplies to trapped person. However, payload mass and center of mass offset is limited not just by the strength of the robot arm, but by the stabilityability of the robot itselfto keep itself upright, i.e. to remain balanced. An object that is too heavy, or that is highly unbalanced, will cause the robot to lose stabilitybalance and fall over, in a similar fashion to a forklift truck or crane.

There are a number of ways this can be mitigated. The *Boston Dynamics* Handle, pictured in figure 1.2, is a bipedal “wheel-leg” robot designed for warehouse tasks that uses a counterweight that can be swung around the robot body in order to maintain stabilitybalance. This design, while reasonably effective, is unsuitable for most other mobile robot designs, which are closer to the ground and have four or more legs or wheels. It is also possible to use an adjustable mass on the robot body **gor2018fault**, however the range of compensation that can be provided is necessarily limited by the size of the robot’s torso.

One option that has yet to be explored is the use of a robotic tail, in a more traditional sense than what is used in the Handle, which resembles something closer to what is found in the animal kingdom. Robotic tails of this kind have already been extensively studied for other areas of robot stabilitylocomotion control, as will be explored in chapter 2.



Figure 1.2: Boston Dynamics Handle loading a pallet. © Boston Dynamics

1.1 Research Concepts

The following section gives an overview of important concepts necessary to understand the issue of mobile robot **stability balance** when carrying a payload and the proposed solution.

1.1.1 Mobile Robot **StabilityBalance**

Stability Balance is a significant issue for mobile robot design. Loss of **stability balance** can mean the robot is unable to move and must be reorientated or retrieved, which may be difficult or impossible in some extreme environments, such as in outer space or a nuclear fuel pool. In the worst case it can result in severe damage or destruction of the robot, and any objects it is carrying. This has become more of an issue as mobile robots have become increasingly fast and agile, often running, jumping and hopping around less controlled environments.

In many ways the consequences of **stability loss of balance** in mobile robots are analogous to other situations. A human that falls over has to pick themselves up before continuing on, and if they are infirm they may require assistance. They could also suffer injury, or if next to a long drop, fall to cause severe injury or death. A forklift truck or other piece of heavy plant can topple, injuring the driver and causing the damage or destruction of vehicles and materials.

~~In general, stability from a biomechanical perspective can be divided into two different types, this introduction, the biomechanical definitions of static balance and dynamic balance. While there are numerous ways to define the difference between the two, such as the maximum lyupanov exponent for dynamic stability ekizos2018maximum will be generalised to be~~

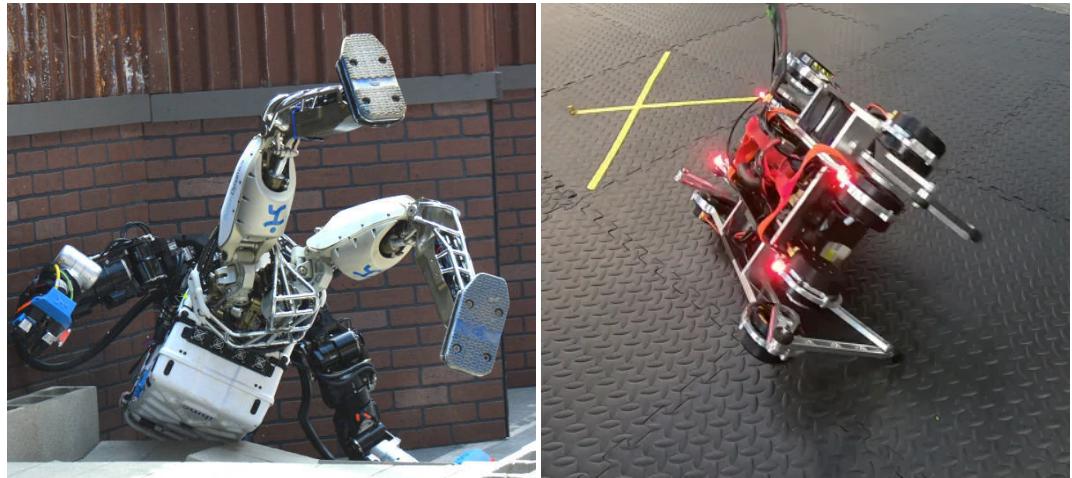


Figure 1.3: Examples of loss of **stability** in bipedal and quadrupedal robots. *Left Image: Public Domain, Right Image: © Google Robotics*

applicable to mobile robots. In human biomechanics, static balance is considered as “quiet standing” where the person is required to maintain a static pose with no external disturbances, and dynamic balance is where the person is required to maintain a stable position while undertaking a prescribed movement, which could be rejecting an external disturbance karimi2011rel (such as compensating for an unstable surface while balancing on one leg hrysomallis2006relations). This can be generalised for a mobile robot: static balance is the balance maintained when the robot is not in motion and not subject to external disturbances, and dynamic balance is when the robot is moving, either to accomplish a goal or reject an external disturbance. When considering this definition from the perspective of the forces acting on the robot, the following definitions will be used: can be used, and will be the definitions used thereafter in this introduction:

- **Static stabilitybalance** only considers the uniform force of gravity and assumes no other forces are acting on the **objectrobot**.
- **Dynamic stabilitybalance** considers other forces and torques on the **objectrobot**, both internal and external, as well as gravity.

A stationary object Therefore, a stationary robot that has no external forces or torques being applied needs to be statically **stablebalanced**, a moving **object**, or an **object robot**, or a **robot** that is having a force or torque applied to it other than gravity, needs to be dynamically **stablebalanced**.

1.1.1.1 Static StabilityBalance

To determine if the robot is statically **stablebalanced**, the gravity axis projection of the center of mass , commonly referred to as the , needs to fall within a defined “support polygon” on the plane perpendicular to the gravity axis plane, as shown in figure 1.5¹. The center of

¹center of mass and center of gravity are used interchangeably here. This assumes a uniform gravitational vector g , which is a suitable assumption for terrestrial robots due to the overwhelming dominance of the earth’s gravitational field. If there is a non-uniform gravitational field, then the center of mass and center of gravity will not be equivalent, and the terms can not be used interchangeably.

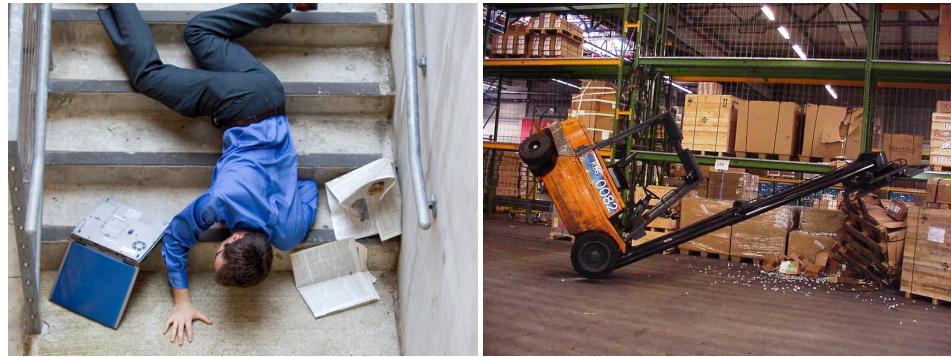


Figure 1.4: Examples of loss of **stability balance** (static or dynamic) in a human and a forklift truck.

mass can be calculated using **equation 1.1**

$$\text{COM} = \frac{\sum_{i=1}^n m_i \mathbf{p}_i}{\sum_{i=1}^n m_i} \quad (1.1)$$

for n bodies of masses $m_{1\dots n}$ and center of mass positions $\mathbf{p}_{1\dots n}$. If the gravity vector is parallel to any of the basis vectors, then the perpendicular components of the center of mass can be used to determine the static **stabilitybalance**.

$$\text{COM} = \frac{\sum_{i=1}^n m_i \mathbf{p}_i}{\sum_{i=1}^n m_i}$$

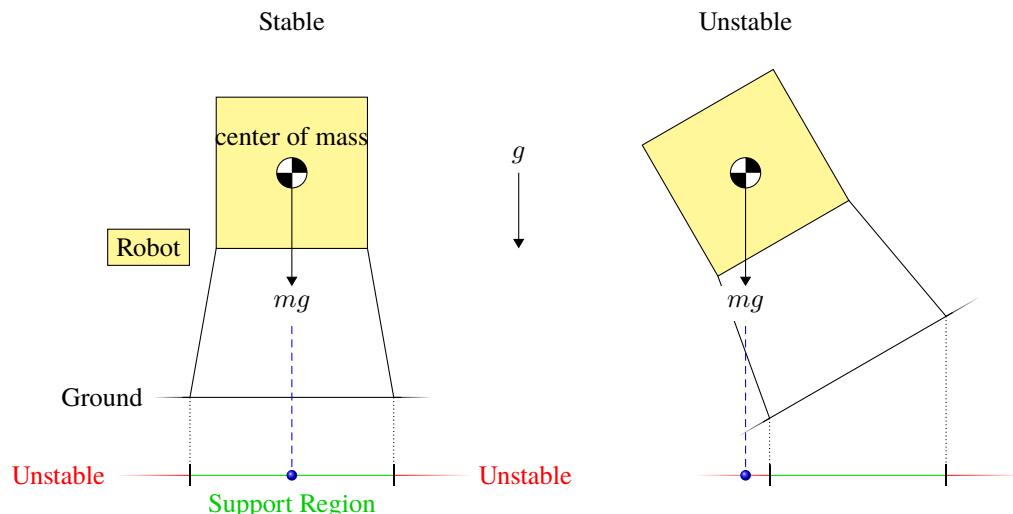


Figure 1.5: 2D representation of the static **stability balance** of a legged robot, with the support region defined by the contact points of the legs with the ground. Notice how the orientation of the robot with respect to the gravity axis can move it in and out of the **static stability statically balanced** region.

Fundamentally, there are two different methods to maintain static **stabilitybalance**:

- Change the region or shape of the support region so the gravity axis projection of the center of mass remains within the bounds of the support region.
- Move the gravity axis projection of the center of mass so it remains within the bounds

of the support region.

The former method can be considered equivalent to a person using a walking pole to steady yourself themselves on uneven ground when hiking, the leg of the pole acts as a new vertex that is used to calculate the support polygon, expanding it sufficiently, or placing your foot in front of you when walking. The latter method can be considered equivalent to leaning back to remain upright when falling forward. Leaning back moves the centre of gravity to keep it within the support polygon.

However, just because the center of mass falls outside of the support polygon does not mean that loss of overall stability balance is inevitable, as long as a force or torque applied to the body counteracts the forces and torques induced by the force of gravity in order to maintain overall stabilitybalance. This is similar to applying a torque to your ankle when standing on one leg a person leaning backwards or forwards to maintain balance when accelerating or decelerating during a sprint, or a strong wind keeping you them upright when leaning forward. If this is the case, then dynamic stabilitybalance is maintained while static stabilitybalance is lost. If the force or torque is removed, then overall stability is lost balance is lost completely. Conversely, forces and torques can also cause loss of overall stabilitybalance even if the center of gravity center of mass does not fall outside of the support polygon. This is similar to being pushed over, or stopping too quickly decelerating too quickly when sprinting and falling forward. So dynamic stability can maintain overall stability balance can maintain balance even if static stabilitybalance is lost, but static stability cannot maintain overall stability if dynamic stabilitybalance cannot maintain balance if dynamic balance is lost.

1.1.1.2 Dynamic StabilityBalance

To determine if the robot is dynamically stablebalanced, it is not enough to only consider the position of the gravity axis projection of the center of mass, as it is for static stabilitybalance. This is where the concept of the zero moment point **vukobratovic2004zero** is useful for mobile robots to check if dynamic stabilitybalance will be maintained. It extends the calculation used for static stabilitybalance by including inertial forces caused by accelerations of the bodies, as shown in figure 1.6. Though it has mostly been utilised for bipedal robots to ensure stability maintain balance while walking, it is also applicable to quadruped robots **montes2005detecting**, **akbas2012zero**, **winkler2017fast**, and has even been investigated for the development of a stability warning system in road vehicles **lapapong2020experimental**. The zero moment point is formally defined as the point at which the sum of the horizontal inertia and gravity forces is equal to zero. It can be thought of as a *dynamically augmented* version of the gravity axis projection of the center of mass. Equation 1.2 defines the The

position of the **ZMP**-zero moment point is given by

$$\tau_i = \mathbf{R}_i \left(\mathbf{I}_i \ddot{\theta}_i - (\mathbf{I}_i \dot{\theta}_i) \times \dot{\theta}_i \right)$$

$$\text{ZMP} = \frac{\mathbf{n} \times \sum_{i=1}^n (\mathbf{p}_i \times m_i \mathbf{g} - \mathbf{p}_i \times m_i \ddot{\mathbf{p}}_i - \tau_i)}{\mathbf{n} \cdot ((\sum_{i=1}^n m_i \mathbf{g}) - (\sum_{i=1}^n m_i \ddot{\mathbf{p}}_i))} \quad (1.2)$$

for a robot or vehicle with n bodies of masses $m_{1\dots n}$, center of mass positions $\mathbf{p}_{1\dots n}$ and center of mass accelerations $\ddot{\mathbf{p}}_{1\dots n}$, in contact with a planar surface of normal vector \mathbf{n} (ZMP cannot be calculated for non-planar surfaces). $\tau_{i\dots n}$ defines the torque acting on each center of mass, which can be calculated from the inertia matrix for the body \mathbf{I}_i , the angular velocity $\dot{\theta}_i$ and acceleration $\ddot{\theta}_i$ of the body, and the transformation matrix \mathbf{R}_i from the origin to the location of the center of mass for the body.

$$\tau_i = \mathbf{R}_i \left(\mathbf{I}_i \ddot{\theta}_i - (\mathbf{I}_i \dot{\theta}_i) \times \dot{\theta}_i \right)$$

$$\text{ZMP} = \frac{\mathbf{n} \times \sum_{i=1}^n (\mathbf{p}_i \times m_i \mathbf{g} - \mathbf{p}_i \times m_i \ddot{\mathbf{p}}_i - \tau_i)}{\mathbf{n} \cdot ((\sum_{i=1}^n m_i \mathbf{g}) - (\sum_{i=1}^n m_i \ddot{\mathbf{p}}_i))}$$

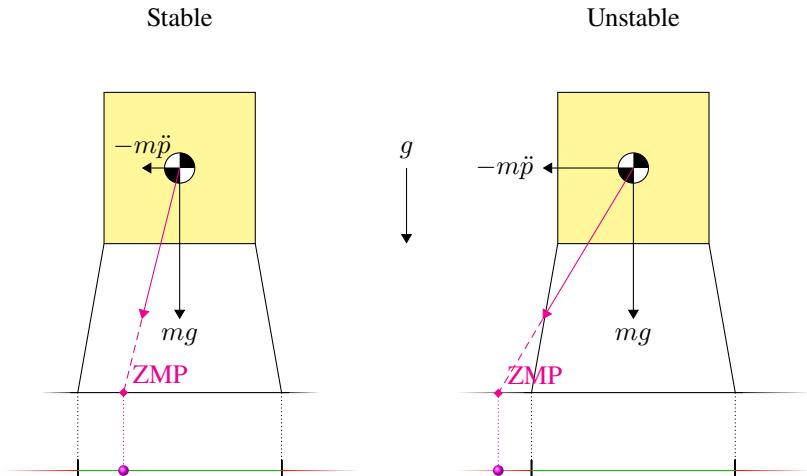


Figure 1.6: 2D representation of the ZMP of a legged robot under horizontal acceleration, with the support region defined by the contact points of the legs with the ground. Notice how increasing the horizontal acceleration of the robot can make it dynamically unstableunbalanced.

1.1.1.3 Addition of a Payload

When a payload is added to the robot, it is equivalent to instantaneously adding an extra body to the robot of mass m_p and position p_p , thus changing the center of mass. Initially the payload will also create an extra contact point with the ground, changing the support polygon so the robot remains statically stablebalanced. However, as soon as contact between the payload and ground is severed, the robot can become statically unstableunbalanced, as shown in figure 1.7. This does not mean the robot will immediately lose stabilitybalance, the dynamic forces created picking up the payload may keep the robot dynamically stablebalanced,

but once the robot is stationary without other forces acting upon it, it may lose **stability balance**. The best way to compensate for this is to change the position of the other bodies so the center of mass remains within the support polygon. This is akin to leaning back when carrying something heavy, but leaning generally has limited range, so can only compensate for lighter payloads.

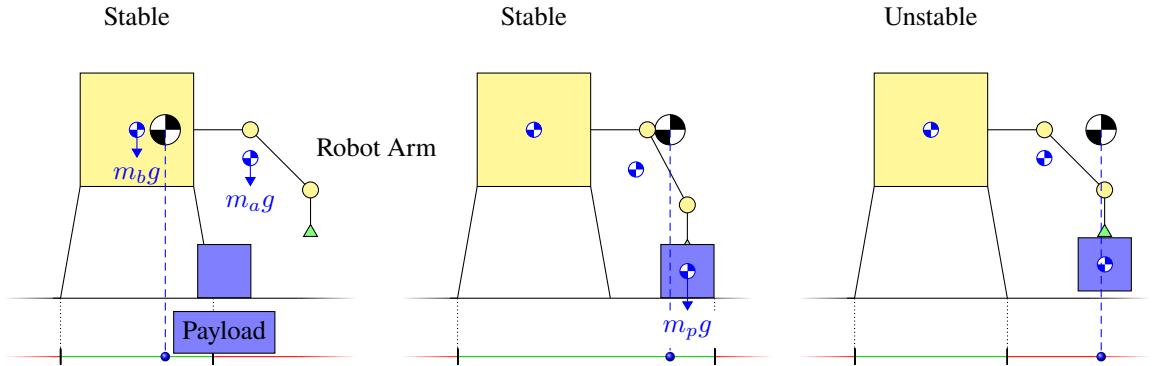


Figure 1.7: 2D representation of the static **stability balance** of a legged robot picking up a payload, with the support region defined by the contact points of the arm and legs with the ground. Notice how the payload acts as a contact point until it is lifted off the ground, preventing loss of **stability balance** even as the COM is translated due to the mass of the payload.

1.1.1.4 Use of a Tail for Maintaining Robot **Stability Balance** with Addition of a Payload

One option to change the position of other bodies is to attach a “tail” to the robot at the opposite end of the robot arm. When the robot is not carrying a payload, the tail center of mass remains close to the body center of mass, and therefore has a smaller contribution to the overall center of mass. When the robot picks up the payload and the center of mass shifts forward, the tail can swing out as shown in figure 1.8, moving the center of mass further from the body which pulls the overall center of mass back towards the center of the body, **maintaining the stability of the robot keeping the robot balanced**.

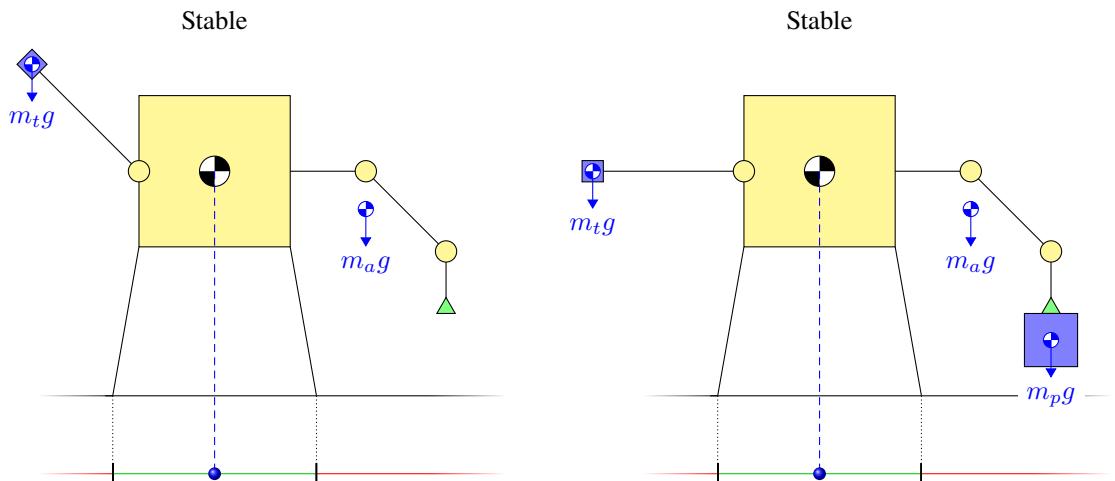


Figure 1.8: 2D representation of the static **stability balance** of a legged robot with a weighted tail picking up a payload. By changing the angle of the tail, the position of its COM can be adjusted in order to compensate for the payload.

1.1.2 Studies of Tails in Animals

Tails are a common sight in vertebrate animals, a natural extension of the spinal column. While some tails are used purely for grasping, communication or decoration, many have a significant function in ~~maintaining stability during locomotion~~ ~~locomotion control~~. The following research studies demonstrate the importance of the tail for ~~stability locomotion control~~ in two animals, by measuring their performance before and after surgical operations to disable the function or remove their tail. While these studies clearly have ethical issues, and only represent a small subset of animal tail functions for locomotion ~~stabilitycontrol~~, they do provide clear proof of the ~~improvements in stability locomotion control improvements~~ tails in animals can allow since they provide a control experiment without the tail.

1.1.2.1 Balancing Ability of a Cat Before and After Partial Tail Paralysis

In **walker1998balance**, experiments were conducted to observe how domestic cats use their tails for balance when walking along a narrow beam, which was shifted laterally at a certain velocity by 2.5 cm or 5 cm while the cat was traversing it. Four cats were trained to walk across the beam, before and after a surgical procedure that severed the nerves in the spinal cord just above the tail, severely affecting its function by causing paralysis of the tail muscles. As can be seen from figure 1.9, this procedure caused the cats to fall from the beam far more often than before surgery.

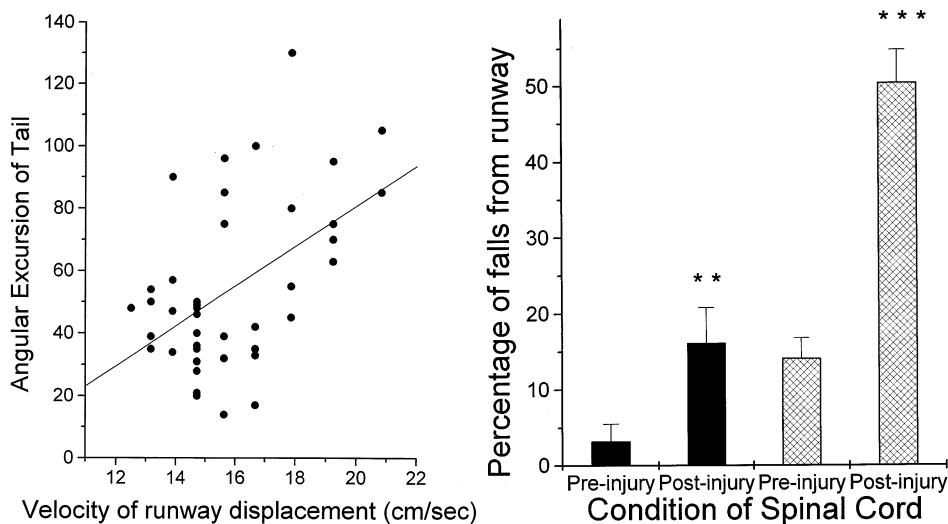


Figure 1.9: Charts from **walker1998balance** showing how the cat's tail is used to maintain balance on the beam when it is shifted, and how impairing it causes a major loss of ~~stabilitybalance~~. Dark bars are a 2.5 cm displacement, cross-hatched bars are a 5 cm displacement.

1.1.2.2 Aerial ~~Stability Orientation Control~~ of a Jumping Lizard Before and After Tail Removal

In **gillis2009losing**, the aerial ~~stability orientation control~~ of the arboreal lizard is examined, with an intact tail and with their tail removed. Lizards with the tail removed are unable to maintain their body orientation and do not land cleanly, as can bee seen in figure

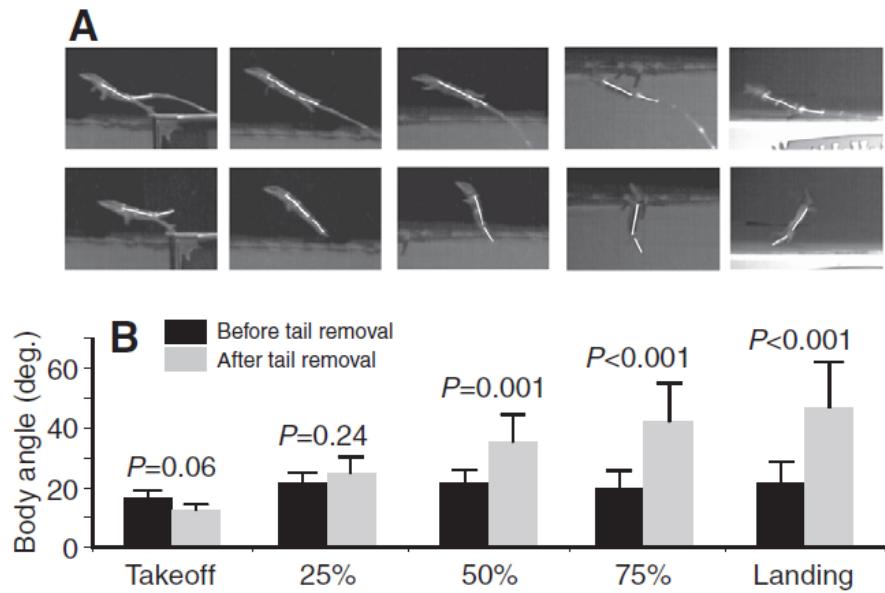


Figure 1.10: Image from [gillis2009losing](#), showing the body angle of a lizard during a jump, before and after tail removal. Section A shows still frames of the lizard at various stages when jumping, with the spinal column highlighted. Section B shows the resulting body angle at these snapshots.

1.2 Aims and Objectives

This research investigates the use of a robotic tail in maintaining the static ~~stability~~balance of a terrestrial mobile robot when carrying a payload, as described in section 1.1.1.4. This is done by measuring the ~~stability~~static balance of a robotic system when carrying a payload using a robotic arm, with and without the addition of a robotic tail designed to adjust the center of mass to ~~maintain stability~~improve static balance. In order to accomplish this, a number of different strands of research need to be brought together. As such, the aim of this research is:

“To develop a robotic tail that can be used to improve stability in a terrestrial mobile robot”

In order to accomplish this aim, the following key objectives are required:

- Investigate the current uses of robotic tails in terrestrial mobile robots in order to gain insight into further research.
- Investigate potential actuation methods for the robotic tail and select a suitable method.
- ~~Investigate if there is any advantage to using a multi-segmented tail.~~
- Design a configurable payload that can simulate a range of mass and center of mass.
- Design a system that can measure ~~stability~~static balance in a way that could be easily used on a mobile robot platform.

- Compare the ~~stability~~static balance performance of a system that is able to pick up, carry and set down the payload with and without an active tail that uses the measured ~~stability~~static balance as a closed loop control signal.

1.3 Thesis Organisation

The rest of this thesis is organised into the following chapters:

Chapter 2: A Literature Review of Terrestrial Robots with Robotic Tails and Their Functions This chapter conducts a “state of the art” review into the existing literature for robotic tails on mobile robots, with an eye towards ~~robot stability~~controlling robot locomotion.

Chapter 3: A Novel Triad Twisted String Actuator for Controlling a Two Degrees of Freedom Joint: Design and Experimental Validation This chapter proposes and experimentally validates a potential design for the tail actuator, based on the twisted string actuator.

Chapter ??: ~~This chapter simulates single and multiple segment tails of the same mass controlling their , which is what contributes to static stability, as mentioned in section 1.1.1.4.~~

Chapter ??: ?? This chapter describes the design and examines the performance of a dummy payload that can have its mass and center of mass easily adjusted.

Chapter ??: ?? This chapter describes the design of an apparatus that measures ~~stability~~static balance using load cells, using a design that could be easily adapted to the majority of mobile robots. Then it proceeds to test the efficacy of the chosen tail design for improving ~~stability~~static balance by using a robot arm to carry various configurations of the payload designed in chapter ??.

Chapter ??: ?? This chapter examines the findings from the previous chapters and considers the impact of the research. It also examines some of the limitations and proposes future work not covered in other chapters.

Chapter 2

A Literature Review of Terrestrial Robots with Robotic Tails and Their Functions

In this chapter, a literature review methodology is described that produces a collection of relevant publications in the field of terrestrial mobile robots with tails. A categorisation system is developed based on the function of the tail and how it contributes to the robot's dynamics. In each category, selected publications are then summarised with accompanying figures from the source material. In the discussion, broader findings about the qualities of the robot tails found in the literature are explained, which include what performance comparisons were made between different tail designs, motions, or control methods within the same publication for the same tail function, any bio-inspiration the publication took from the animal kingdom, be it through direct performance comparison, or loose "inspiration", and finally if multiple segment tails confer any performance advantage in specific tail applications that are experimentally investigated in the literature.

2.1 Introduction

The field of terrestrial robots with robotic tails is incredibly diverse, reflecting the many functions of tails in the animal kingdom. Even discounting tails used for fluid dynamics, e.g. swimming, water walking and flying robots, and focusing only on robots that use their tail during “terrestrial” locomotion, broadly defined as when a robot is moving along a contiguous surface, or jumping from one surface to another surface, there are many applications of robotic tails. In order to make sense of the state of the field, an abstract categorisation system is considered based on the environment the robot is in when the tail is active, the specific action the robot is taking to move itself in space, and what the specific function of the tail is with respect to the robot dynamics. Using this categorisation system, various examples are explored from a set of research articles selected using specific keywords. From this, conclusions can be derived about the general design and operation of robotic tails, which can be used to influence and guide the research covered in chapters ?? and ???. While a literature review has been conducted in this field before **saab_rone_ben-tzvi_2018**, this focused on a narrow selection of mobile robots with tails, and therefore does not cover the same range of literature that is analysed in this chapter, which takes a more systematic approach as can be seen in section 2.2.

2.2 Literature Review Methodology

Using three online publication repositories, *IEEE Xplore*, *Scopus* and *Web of Science*, a search query was conducted to find relevant publications. The query was tailored to include all publications with **tail*** or **appendage** in the title along with **robot*** (* indicates a wildcard suffix), but to exclude publications that concerned swimming, water walking, or flying robots, as using a tail as a rudder to influence fluid dynamics was outside of the scope of this research. Further exclusions were added upon experimentation with the query in order to remove false positives in areas such as chemistry (as molecules are often described as having “tails”) or medicine (as it did not pertain to mobile robots and usually concerned biological structures such as proteins and cells). The date range was set from January 1980 to June 2021, when the search was conducted, to exclude outdated publications:

(Tail* **OR** Appendage) **is contained in Document Title** & Robot* **NOT** (Fish **OR** Swim) **NOT** (Surgery **OR** Medic* **OR** Tumour) **NOT** (Helicopter **OR** Unmanned Aerial Vehicle **OR** UAV) **NOT** Underwater **NOT** (Chemical **OR** Chemistry) **NOT** Tailor* **is contained in Document Title**

As a result **498** publications were discovered. After duplicates were removed and after screening abstracts and full texts, a total of **84** unique publications were selected for inclusion. A flowchart of this process can be found in figure 2.1.

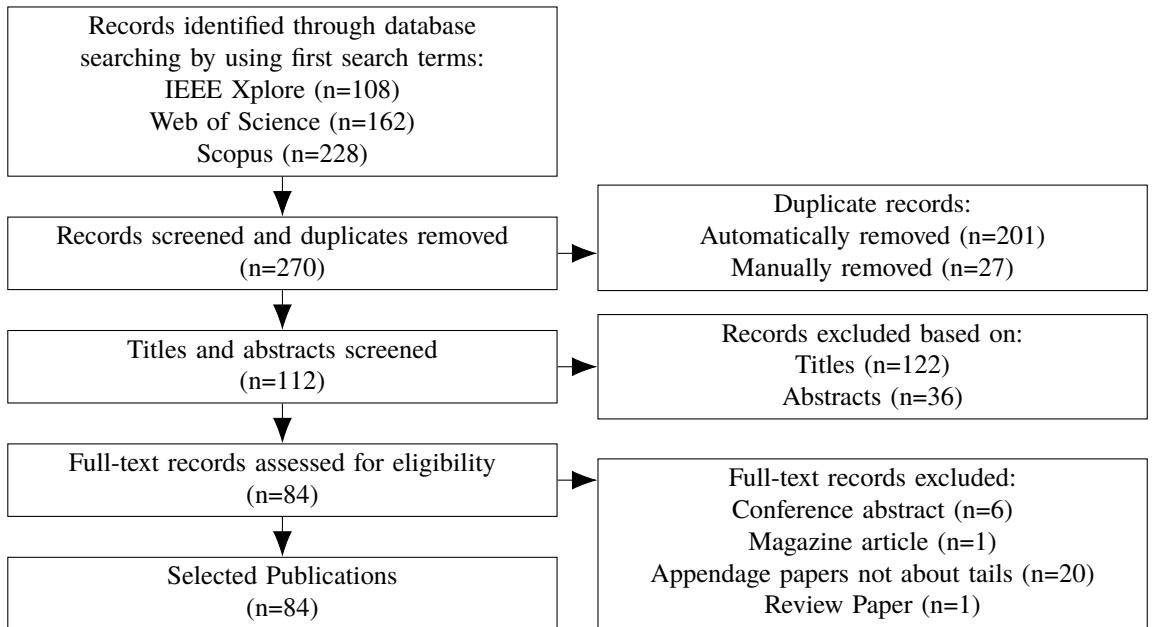


Figure 2.1: PRISMA Flowchart of the publication selection process.

The final selection included 55 *experimental* publications, which included experimental results from a physical prototype, and 29 *simulation* publications which only included results from dynamics simulations and/or analytic functions. This review will focus on experimental publications, as they have stronger evidence for the efficacy of their research.

2.3 Tail Functions of Terrestrial Robots

2.3.1 Categorisation System for Robotic Tail Functions

The resulting publications represented a wide array of different robot designs and multiple forms of locomotion. Therefore, a simple categorisation system was required in order to better understand the majority of the functions of these tails. After careful analysis of the literature, three questions could be asked that provided common answers:

1. Is the robot on the ground, in the air or transitioning between those states when the tail is active?
2. What precisely is the robot doing when the tail is active?
3. What does the tail do to the dynamics of the robot (i.e. what would happen if there *wasn't* a tail present)?

Each question can then assign a tiered category to a publication based on the answer, the three tiers were named *Environment*, *Action* and *Function*.

1. **Environment:** The general domain the robotic system is operating in when the tail is active. From the reviewed literature, three categories have been created:

- *Terrestrial*: A robot with an active tail when the robot is touching a surface, such as a robot driving along the ground.
- *Aerial*: A robot with an active tail when the robot is in free space, such as a robot which *has jumped* into the air, or is falling off a ledge.
- *Transition*: A robot with an active tail when the robot is just about to transition from between the two previous environments, such as a robot *just about* to jump into the air.

2. **Action:** The specific action the robot is performing when the tail is active. Most actions are unique to each environment, except for *hopping*.

- *Straight*: A robot travelling across a surface maintaining its direction of travel.
- *Accelerating*: A robot changing its velocity in the direction of travel across a surface. In the literature, this was a robot coming to complete stop, and starting from stationary.
- *Turning*: A robot changing its direction of travel across a surface, such as a robot turning a corner.
- *Balancing*: A robot undergoing external disturbances while travelling across a surface, typically due to adverse terrain, such as a robot navigating a rough and uneven surface without falling over.
- *Hopping*: A robot executing a sequence of periodic jumps in order to travel across a surface, similar to the method of locomotion of a kangaroo.
- *Jumping/Falling*: A robot executing isolated non-periodic jumps, falling off a ledge or launching off a ramp, typically to transition from one surface to another at a different altitude and/or orientation.

3. **Function:** The purpose of the tail when the robot is performing the action. These categories usually apply to multiple actions. A visual illustration of these categories is shown in figure 2.2.

- *Stability*: The tail is used to *maintain* some aspect of the robot's position and/or orientation from the start of the action ([\(i.e. rejecting a disturbance\)](#)).
- *Initiation*: The tail is used to *change* some aspect of the robot's position and/or orientation from the start of the action.
- *Amplification*: The tail is used to *amplify* the effects of an action by other parts of the robot (such as the legs) which changes the position and/or orientation.

For example, a robot with a tail that helps it increase its apex when hopping, is in the *Transition* environment as it is about to transition from being on the ground to in the air when

the tail is active. The robot itself locomotes by hopping, so it is performing a *Hopping* action, and since without the tail the robot would still be capable of hopping, but would not have such a tall apex, the tail can be considered to be performing *Amplification* of the robot's existing capabilities. Overall, this results in the categorisation *Transition* → *Hopping* → *Amplification*.

Another example is a robot with a tail that prevents it from falling over on rough terrain. The robot is in the *Terrestrial* environment as it is on the ground when the tail is active. The robot itself is performing a *Balancing* action as it is trying to remain upright during locomotion, and since without the tail the robot would fall over, the tail can be considered to be maintaining the *Stability* of the robot. Overall, this results in the categorisation *Terrestrial* → *Balancing* → *Stability*.

This categorisation system does not include robots that have a tail but are not mobile, or where the tail is used for non-locomotion tasks, such as self righting **kessens2017ground** or dragging objects **kim2018bound**. It also does not include robots where the tail is in constant contact with the ground as essentially an extra leg **mcinroe2016tail**. Figure 2.3 shows this categorisation system in a tree format.

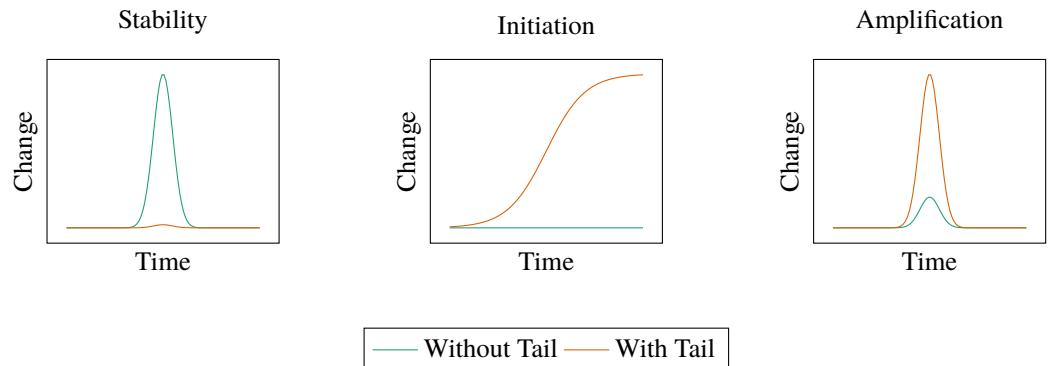


Figure 2.2: Abstract graphs of the different functions of the tail, with the magnitude representing some kind of change in the robot's position and/or orientation, showing how the introduction of a tail improves the performance of this function. The **Stability** function aims to reduce the magnitude of a disturbance that causes the robot to deviate from a set position and/or orientation. The **Initiation** function aims to change the robot's set position and/or orientation that *cannot* be done with an equivalent robot design without a tail. The **Amplification** function aims to increase the magnitude of a change of the robot's set position and/or orientation that *can* be done with an equivalent robot design without a tail.

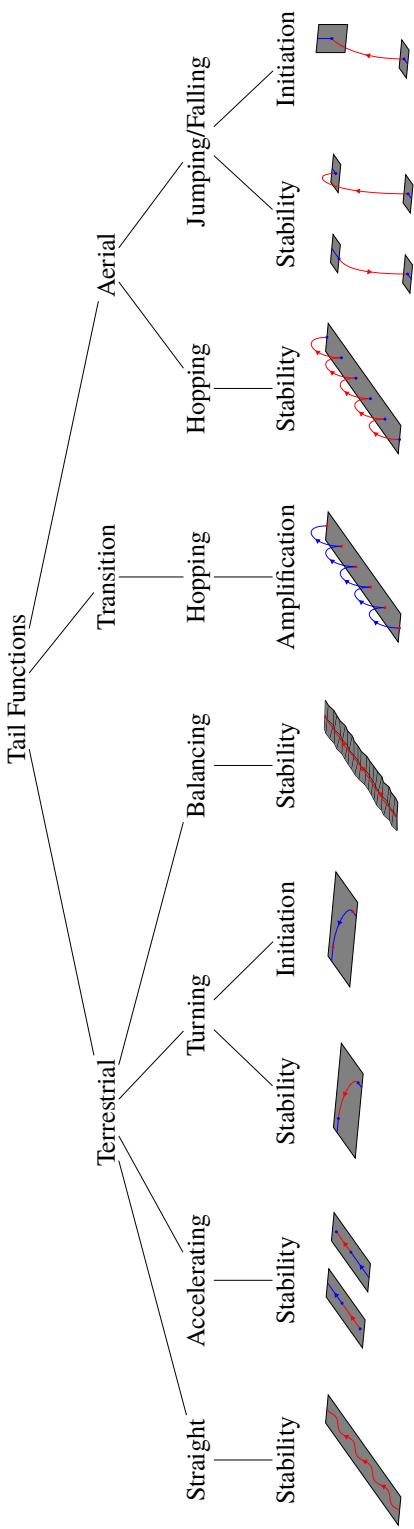


Figure 2.3: Tree diagram of all the categorisations found in the publications with accompanying visual diagrams.

2.3.2 Examples of Robotic Tails from each Category of the Categorisation System

2.3.2.1 *Terrestrial* → *Straight* → *Stability*

zhang2016effects uses a quadrupedal robot with a gait controlled by a central pattern generator.

Upon initial experiments with a trotting locomotion with no active tail, the robot would not maintain a set heading, it would instead slowly begin to drift in a circle. Visual observations noted that the robot would topple onto its front left leg that was in “swing” phase (lifted off the ground), and it would drag on the ground until in “stance” phase. This resulted in a difference between the ground reaction force of the left and right feet which caused the drift in locomotion path.

By implementing a swinging tail that imparted an opposing torque to the direction of the topple, the differences between the left and right ground reaction force were reduced, and the robot could maintain its heading, as can be seen in figures 2.4 and 2.5.

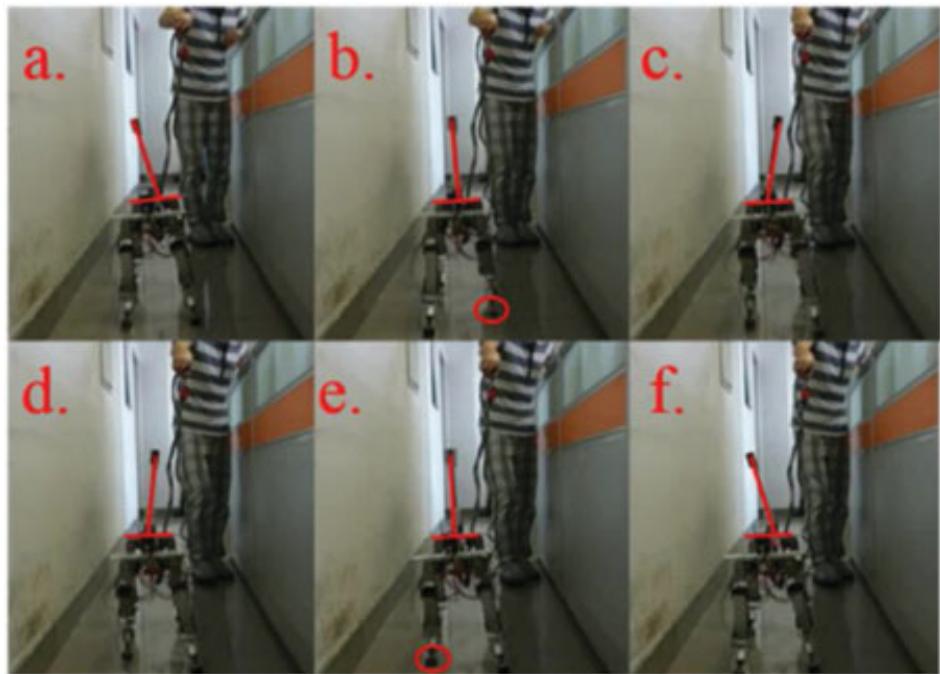


Figure 2.4: Image from **zhang2016effects** showing how the tail moves during the gait in order to correct for heading drift.

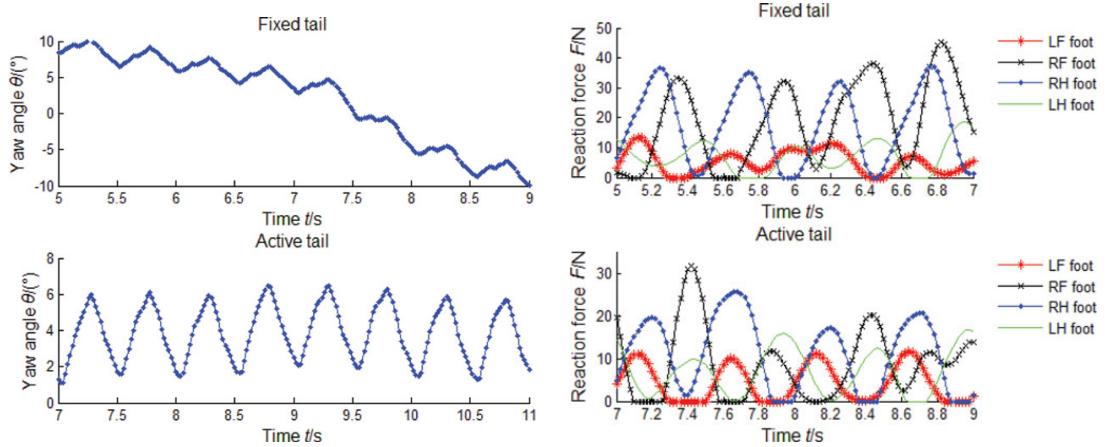


Figure 2.5: Data from **zhang2016effects** showing the effects of a static and dynamic tail on maintaining robot heading in a trotting gait.

2.3.2.2 *Terrestrial → Accelerating → Stability*

patel2014rapid used inspiration from the cheetah to improve the acceleration and braking capabilities of a wheeled robot. The research is based on findings from **williams2009pitch**, which shows that quadruped acceleration and deceleration in the animal kingdom is limited by their ability to constrain body pitch to prevent toppling over. It can be considered analogous to a motorcycle: accelerate too fast and the vehicle will “pop a wheelie” and potentially flip backwards, decelerate too fast and the opposite may occur, as shown in figure 2.6.

Using a combined state feedback and proportional integral controller based on the angular position of the tail, and the angular velocity of the tail and body, the researchers were able to increase the acceleration and braking capabilities of the robot by using the tail to generate an opposing torque to the direction of body pitch, as can be seen in figures 2.7 and 2.8. This was verified by running a series of experiments, increasing the acceleration/braking magnitude until the robot failed to complete the test by toppling over.

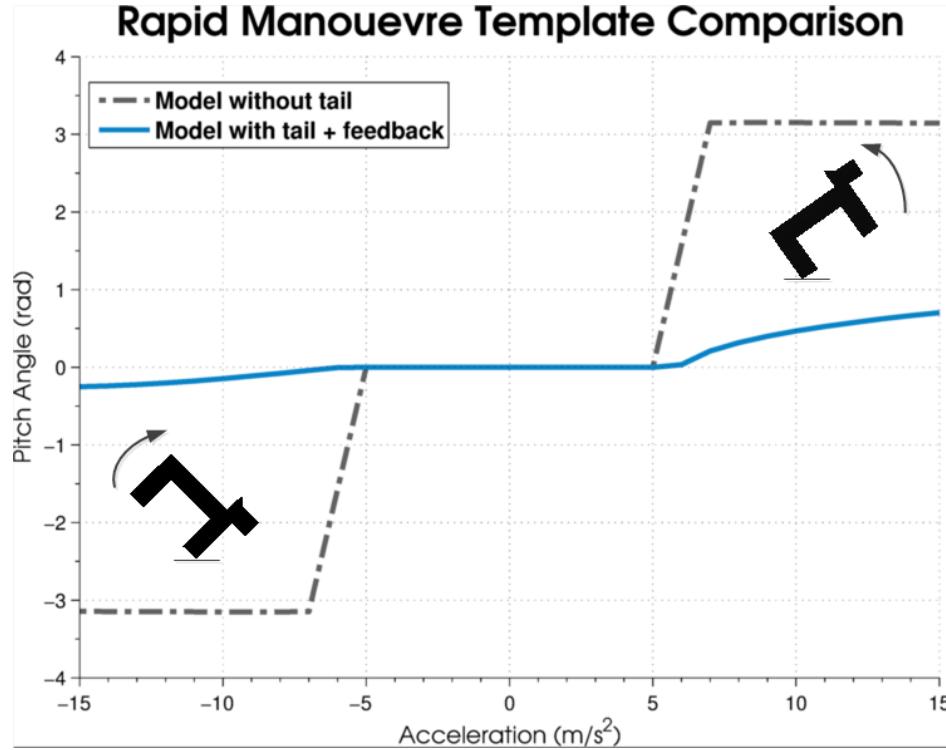


Figure 2.6: Simulation Data from **patel2014rapid** showing how the body pitch would be reduced when accelerating or braking with an active tail.



Figure 2.7: Images from **patel2014rapid** showing the robot performing a rapid acceleration test with the tail.

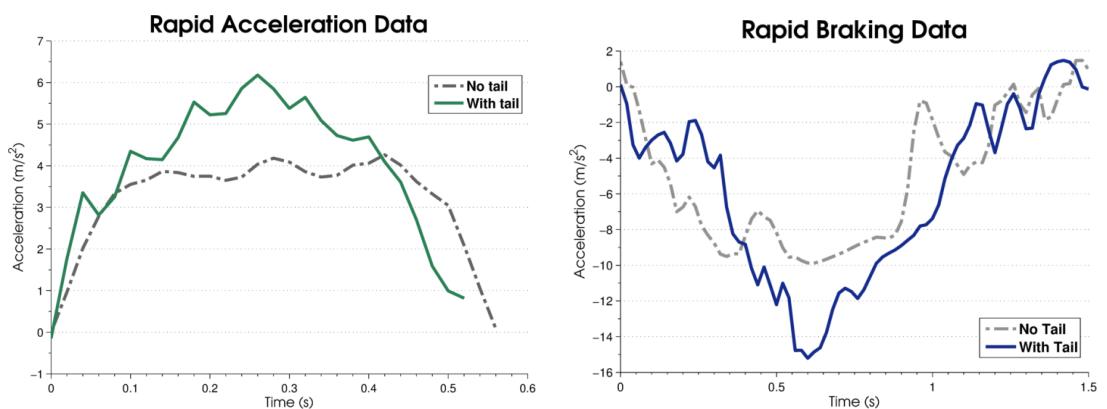


Figure 2.8: Experimental Data from **patel2014rapid** showing the maximum acceleration achieved with and without an active tail.

2.3.2.3 Terrestrial → Turning → Stability

patel2013rapid, **patel2015conical** took similar inspiration from the cheetah to allow for tighter turns by allowing greater lateral acceleration. **patel2013rapid** swings the tail out in a single motion in the direction of the turn, producing an opposing torque to the centrifugal force that would otherwise topple the robot during the turn. In contrast, **patel2015conical** moves the tail constantly in a conical motion, the direction of rotation in the direction of the turn, as shown in figure 2.9. This also produces an opposing torque in the same fashion, but was not limited in duration, as in the first strategy the tail would eventually contact the ground. This allowed for turns of longer duration to be stabilised, as can be seen in figure 2.10.

The control system and experimental procedures were similar to those in **patel2014rapid**.

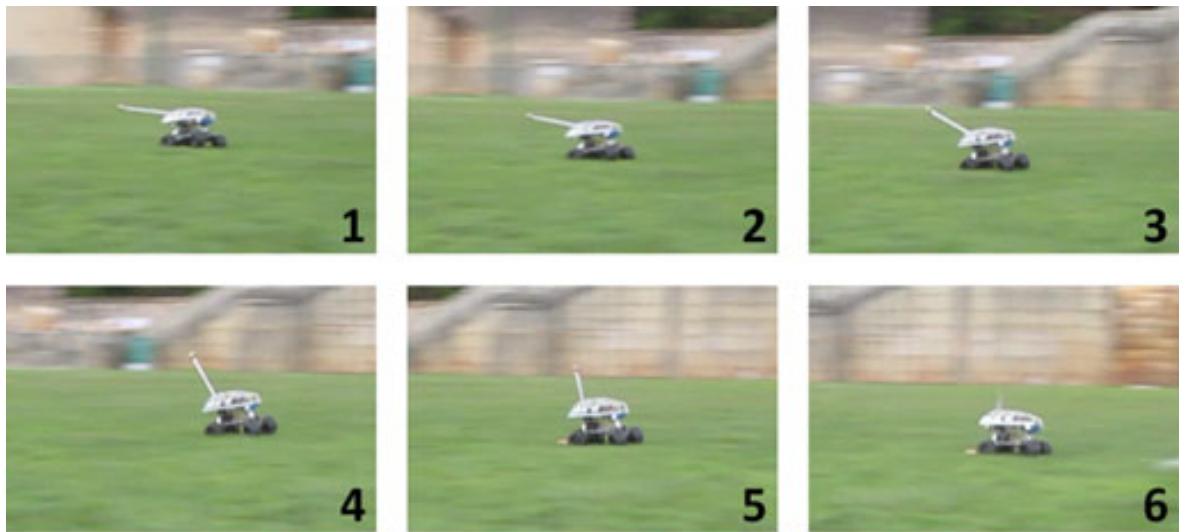


Figure 2.9: Images from **patel2015conical** showing the robot performing a turn with the tail.

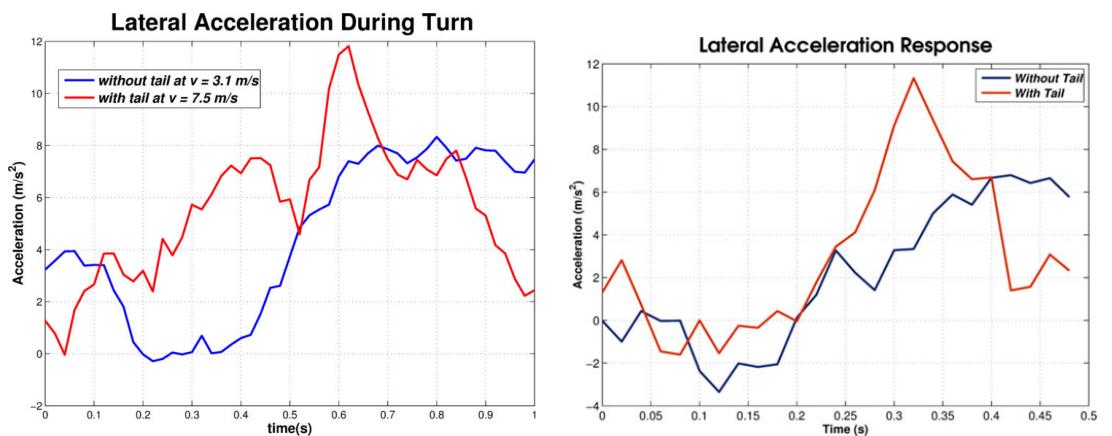


Figure 2.10: Experimental Data from **patel2013rapid** (left) and **patel2015conical** (right) showing the maximum lateral acceleration achieved with and without an active tail. Note the right graph manages similar results to the left graph.

2.3.2.4 Terrestrial → Turning → Initiation

pullin2012dynamic, **kohut2013precise** both use similar robot designs, insect like robots that locomote using 6-8 pairs of small legs (in this case **pullin2012dynamic** used a robot with eight legs, and **kohut2013precise** used six legs) as shown in figure 2.11. Both robots are designed to be very light (52 g in **pullin2012dynamic** and 46 g in **kohut2013precise**) so the legs have a low friction force with the ground. A suitably weighted tail, when swung out in a horizontal motion, can overcome this friction force and impart enough torque to rotate the body of the robot to a new heading, as shown in figure 2.12.

kohut2013precise compared an open and closed loop response of the tail, as shown in figure 2.13, while **pullin2012dynamic** only compared open loop tail responses at different frequencies and amplitudes, as can be seen in figure 2.14.

Both experiments were able to greatly increase the turning rate of the robot over a turn using a differential drive, at $360^{\circ} \text{ s}^{-1}$ for **kohut2013precise** and $400^{\circ} \text{ s}^{-1}$ for **pullin2012dynamic**.

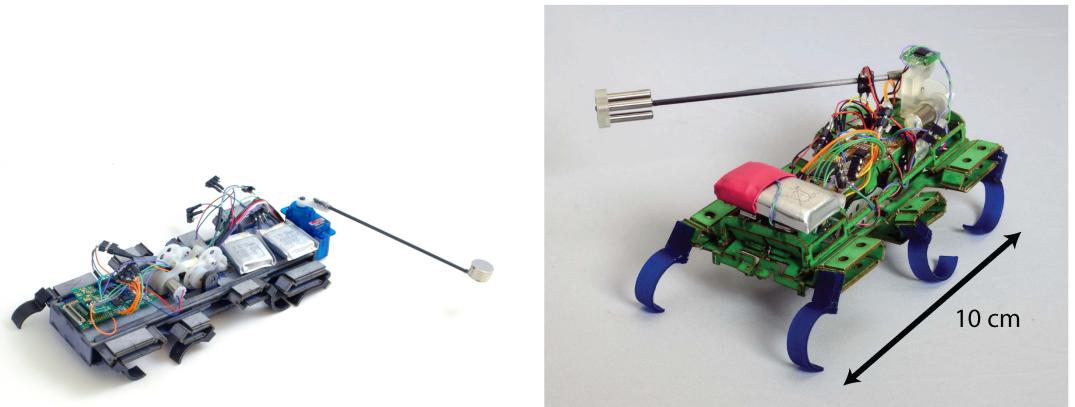


Figure 2.11: Images of the hexapedal and octopedal insect like robots in **pullin2012dynamic** (left) and **kohut2013precise** (right).

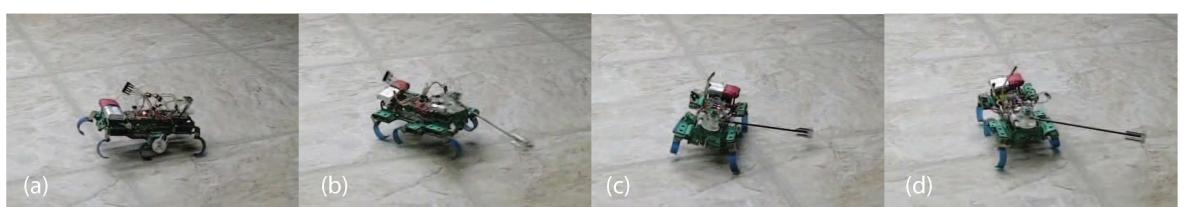


Figure 2.12: Images from **kohut2013precise** showing the robot using its tail to make a turn.

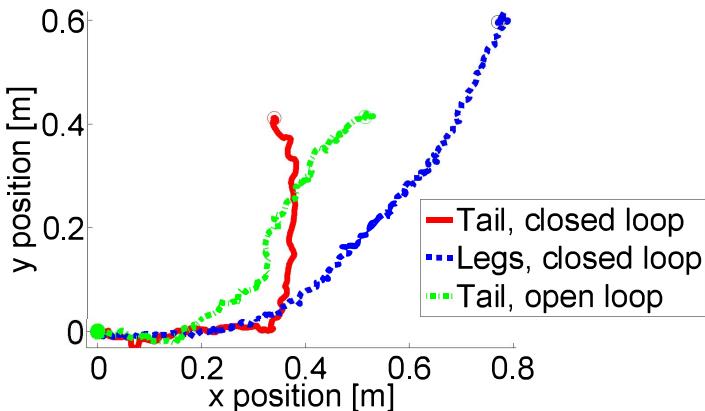


Figure 2.13: Data from **kohut2013precise** comparing the robot path on an XY plane for a 90° differential drive turn, open loop tail turn, and closed loop tail turn. Note the vastly increased sharpness of the turn when a tail is used.

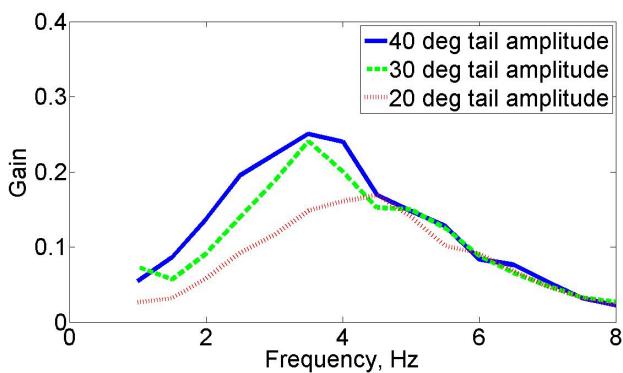


Figure 2.14: Data from **pullin2012dynamic** showing the gain in yaw rate for various open loop tail trajectories and different frequencies and amplitudes.

2.3.2.5 *Terrestrial → Balancing → Stability*

The forces that may cause the robot to topple over when balancing can result from both *internal forces*: ~~robot design or joint inertia when moving at high speed, caused by the acceleration and deceleration of motors which create inertial forces on the robot body~~, and *external forces*: ~~uneven terrain or impact, caused by some kind of external disturbance, such as a sudden change in orientation due to uneven terrain, or another body colliding with the robot~~. **takita2002development** is concerned with internal forces, whereas **briggs2012tails** is concerned with external forces.

takita2002development used a dinosaur like bipedal robot with a long neck and tail. The neck and tail then swung from side to side during the gait, maintaining the stability of the robot. Two experiments were conducted using different strategies for maintaining stability. The “static” method swung the neck and tail in a trapezoidal motion in order to keep the center of mass within the area of the current foot on the ground. The “dynamic” method calculated the zero moment point of the robot and instead constrained the motion of the neck and tail to keep the zero moment point within the area of the foot. This resulted in a smaller motion of the neck and tail which enabled a faster gait, from 19.5 mm s^{-1} up to a theoretical 208 mm s^{-1} , though in practice the velocity was limited by the motor performance. Figure 2.15 shows how the trajectory of the center of mass changes between the static and dynamic walk, and figure 2.16 shows the trajectory of the center of mass and zero

moment point for the dynamic walk.

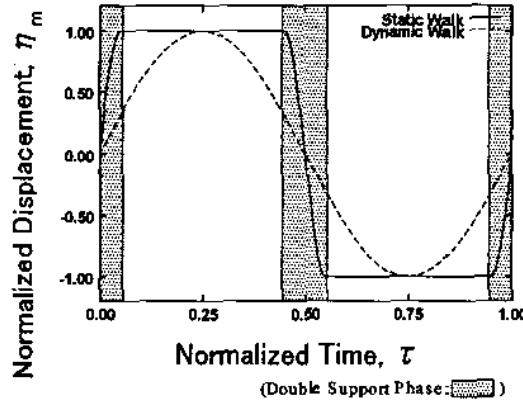


Figure 2.15: Data from **takita2002development** showing the change in COM for a static and dynamic walk over a single walk cycle, normalised to the full amplitude of the static walk COM trajectory.

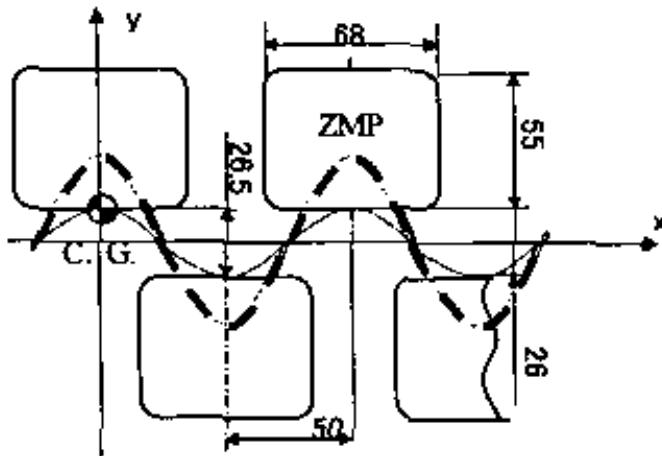


Figure 2.16: Data from **takita2002development** showing the trajectory of the COM and ZMP for a dynamic walk. For a static walk the COM trajectory would be approximately equal to the ZMP trajectory.

briggs2012tails uses a quadrupedal robot closely modelled on a cheetah which is hit in the torso by a “wrecking ball” to simulate a disturbance. In the control experiment the weighted tail remains static, in the active tail experiment the tail responds in an open loop trajectory when triggered by an accelerometer that senses the impact, as shown in figure 2.17. The active tail experiment was able to significantly reduce the hip displacement after impact, as can be seen in figure 2.18.

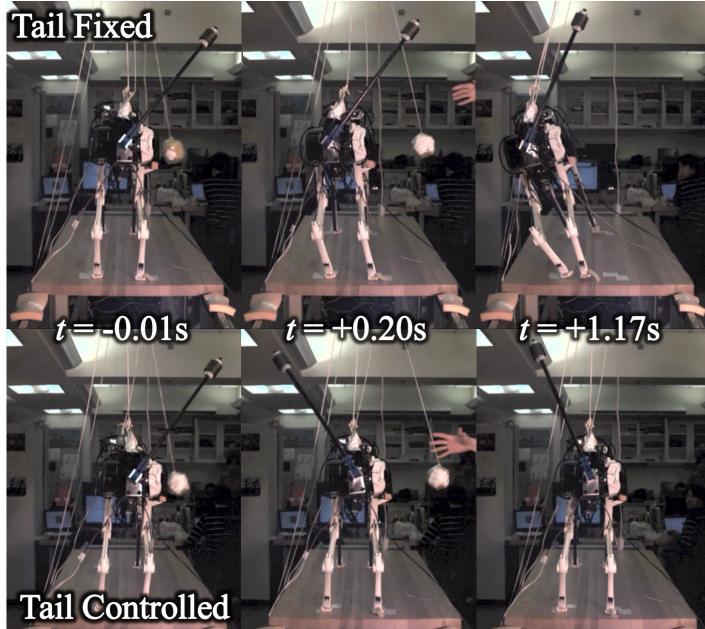


Figure 2.17: Images from **briggs2012tails** showing how the tail or body deflect when hit by the wrecking ball, depending on if the tail is active.

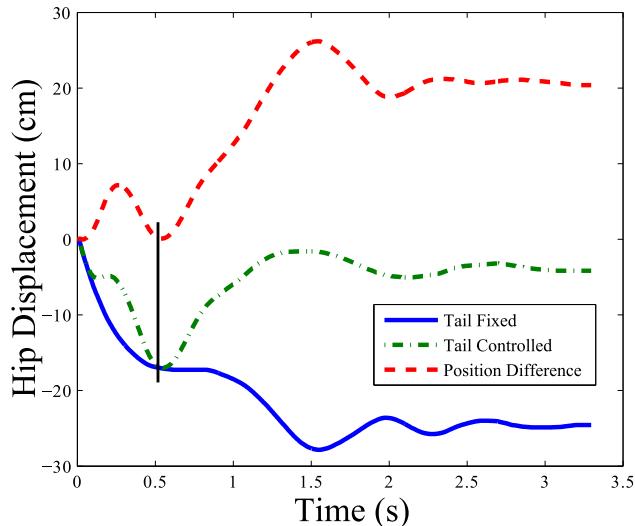


Figure 2.18: Data from **briggs2012tails** showing the difference in hip displacement between a fixed and controlled tail.

2.3.2.6 Transition → Hopping → Amplification

sato2016development, **heim2016designing** both use a tail to increase the magnitude of a hopping gait. **sato2016development** increases the *height* of the hop, while **heim2016designing** increases the *length*.

sato2016development uses a bipedal robot with a long, flexible tail. The tail consists of six segments, connected together by passive spring revolute joints. The only active joint connects the tail assembly to the robot body. This creates a “whip-like” motion in the tail when the joint is actuated, as shown in figure 2.19. By experimenting with different spring constants, an optimum value that maximises jumping height can be found, as can be seen in figure 2.20. Results showed a significant increase in jump height, 256 mm over 240 mm for

a model with a rigid tail, and 210 mm for a model with an inactive tail, demonstrating the superiority of a flexible tail in this application.

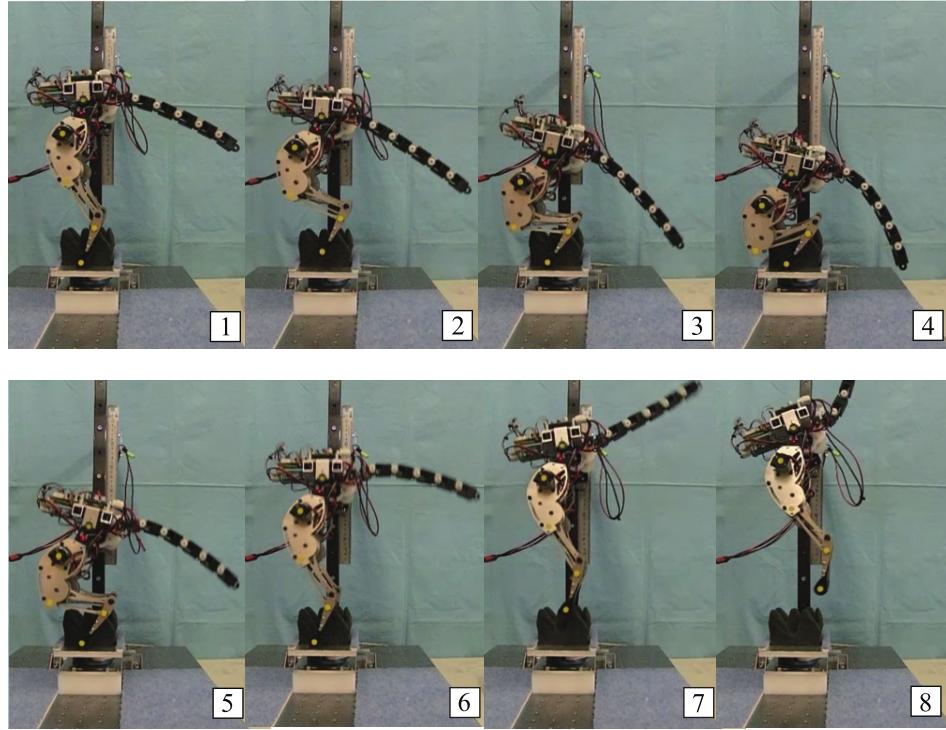


Figure 2.19: Images from **sato2016development** showing the motion of the tail during a jump.

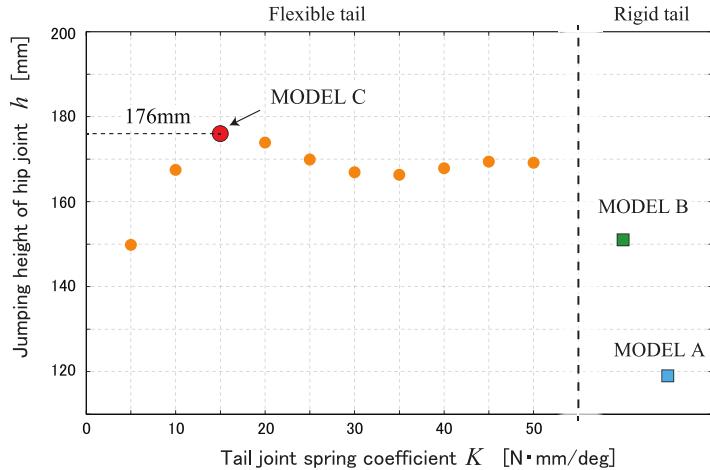


Figure 2.20: Data from **sato2016development** showing the optimal spring constant for maximising jumping height.

heim2016designing uses a quadrupedal robot that locomotes in a galloping motion, as shown in figure 2.21. By using a weighted tail in an open loop trajectory, the robot is able to increase its forward velocity while also reducing body pitch. Two different tail lengths and masses were used for experiments, a long light tail (31 g/168 mm), and a short heavy tail (53 g/128 mm). Different open loop amplitudes, 35° and 65°, were also used for each experiment. Results showed an increase in forward velocity and reduction in body pitch, with the best forward velocity of 0.558 m s⁻¹ from the short heavy tail at 35°, and the best reduction in body pitch per stride of 4.6° from the long light tail at 65°. These results are compared to a forward velocity of 0.479 m s⁻¹ and a body pitch per stride of 8.5° for a fixed “passive” tail, as can be seen in figure 2.22.

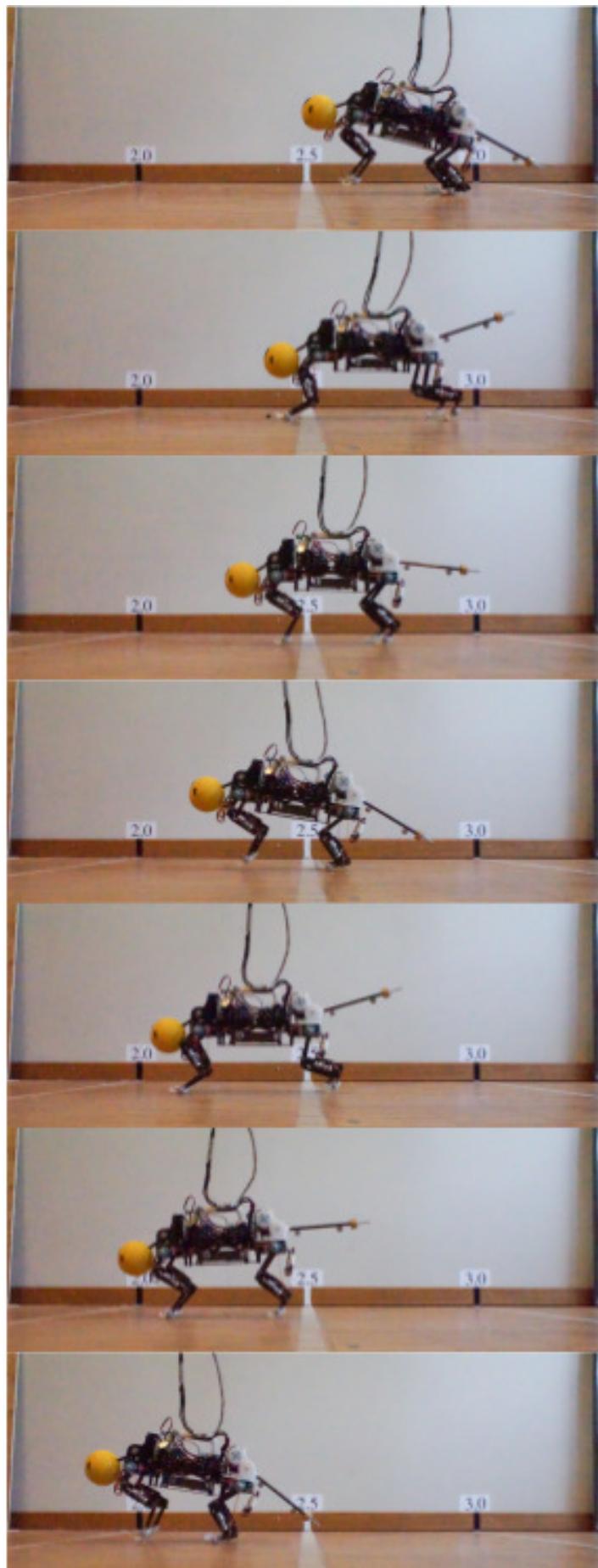


Figure 2.21: Images from **heim2016designing** showing the motion of the body and tail over a single stride cycle.

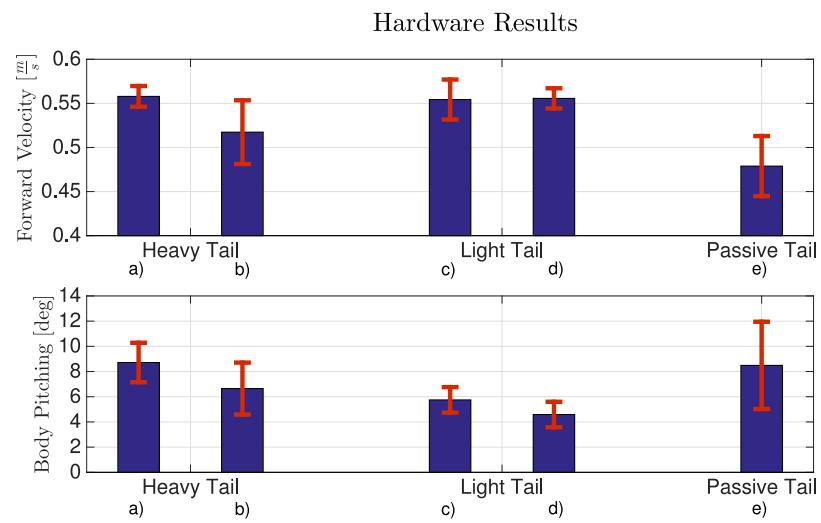


Figure 2.22: Data from **heim2016designing** showing the increase in forward velocity and/or reduction in body pitch for the different tail lengths and masses. Note that a) and c) correspond to a 35° tail amplitude, while b) and d) are a 65° amplitude.

2.3.2.7 Aerial → Hopping → Stability

heim2016designing also used a tail to reduce body pitch, which prevented the robot from falling over forward when hopping. **liu2014bio** used a kangaroo like robot with a tail to also decrease body pitch as shown in figure 2.23, in both an open and closed loop tail trajectory. The open loop tail was able to reduce the body pitch range to 5.17° and root mean square error to 1.17° , and the closed loop tail to 4.49° with an root mean square of 0.96° , as compared to 7.18° with an root mean square of 2.24° for a stationary tail (in this case, *range* refers to the difference between the maximum and minimum pitch angle during the hop).

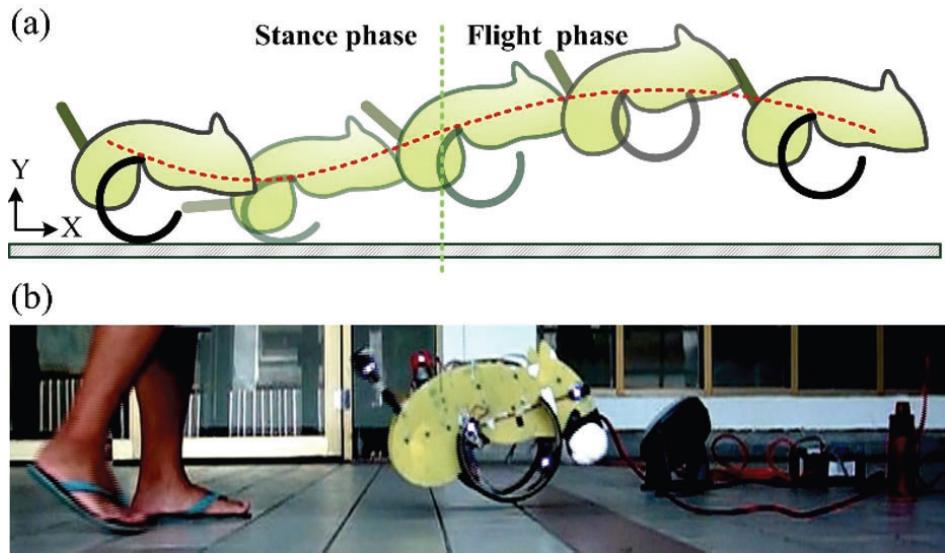


Figure 2.23: Images from **liu2014bio** showing a single stride of the robot, and an image of the robot just before landing.

an2020development also developed a bipedal robot with a 2 degree of freedom tail that was used to maintain a stable hopping motion.

2.3.2.8 Aerial → Jumping/Falling → Initiation

chang2011lizard used a wheeled robot with a weighted tail to reorient the robot in mid-air. Two experiments are carried out to this end as shown in figure 2.24, where the robot is released from a vertical position on a wall and attempts to land horizontally, as can be seen in figure 2.25, and where the robot drives off a ledge and attempts to land on a 45° sloped surface, as can be seen in figure 2.26. The tail used a proportional derivative feedback controller with body orientation as input. In both experiments, the robot was able to successfully reorient itself.

zhao2015msu used a robot that was able to wheel itself to a specific position, then rotate itself using the tail as an appendage into a different orientation which allowed it to launch itself into the air. Then once in the air it would use the tail to orient itself back so it landed on its wheels.

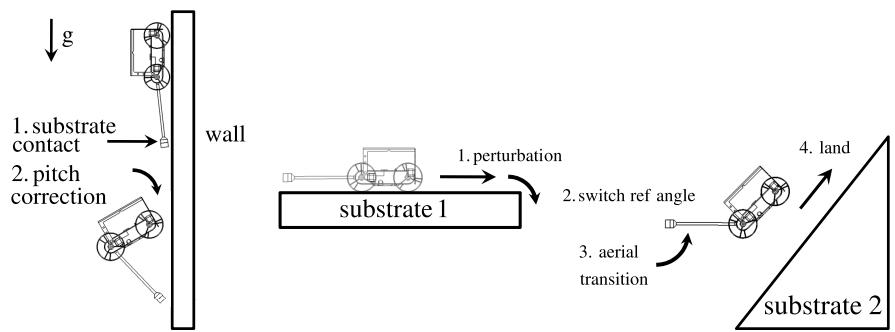


Figure 2.24: Diagrams from **chang2011lizard** outlining the two reorientation experiments.

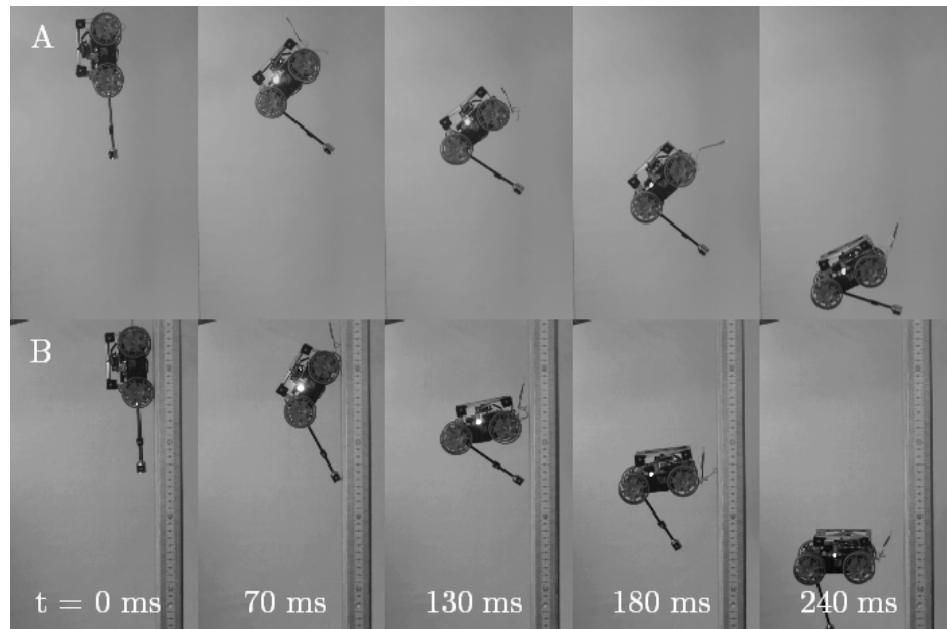


Figure 2.25: Images from **chang2011lizard** showing the wall reorientation experiment.



Figure 2.26: Images from **chang2011lizard** showing the slope reorientation experiment.

2.3.2.9 Aerial → Jumping/Falling → Stability

chang2011lizard also conducted an experiment where the robot would hit a small obstacle when driving along a surface, causing it to go airborne, as shown in figure 2.27. Experiments were conducted without a tail, with a static tail, and with a closed loop controlled tail, as shown in figure 2.28.

libby2016comparative had a hexapedal robot with a tail that was able to remain upright when running off a ledge.

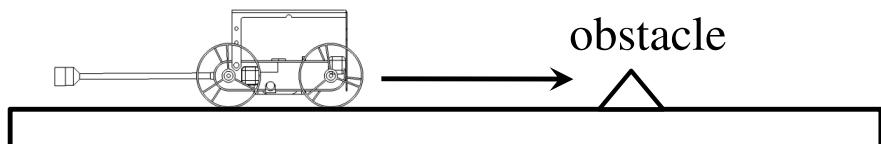


Figure 2.27: Diagram from **chang2011lizard** showing the obstacle impact test.

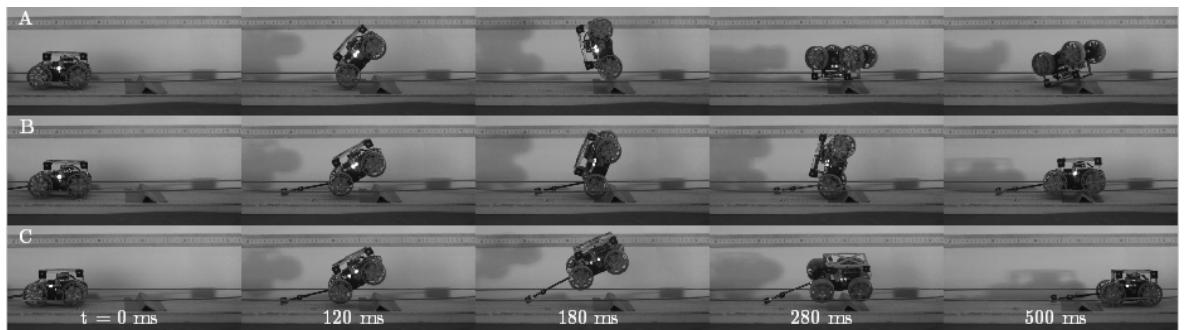


Figure 2.28: Images from **chang2011lizard** showing the slope obstacle impact test without a tail, with a static tail, and with a dynamic tail with closed loop feedback.

2.4 Discussion

2.4.1 Tails with Multiple Functions

Some robot tails were designed to be multi functional, as evidenced by experiments of different functions within the same publication, such as **heim2016designing**, and/or multiple publications that included the same robot, such as **patel2013rapid**, **patel2014rapid**, **patel2015conical**.

2.4.2 Performance Comparisons in the Selected Publications

A number of experimental publications varied their tail design **heim2016designing**, **sato2016develop**, or control system **liu2014bio**, **zhao2015msu**, and were able to record improvements in performance for various metrics. Figure 2.29 shows selected publications' performance comparisons, normalised to the best performing variation for each metric, for a unitless comparison. For control systems, the more complex and advanced the better the performance,

understandable since a publication is unlikely to test a more complex control system unless there is high confidence in a performance improvement. For tail design, the picture is more complex. While **sato2016development** demonstrates a clear improvement in performance for a flexible tail, a more complex design, **heim2016designing** shows how tail design can result in trade-offs between different metrics, with different tail designs being superior. However, it is possible to see the *Long Light Tail* (65°) is the optimal choice, since the reduction in *Forward Velocity* was slight, but the reduction in *Body Pitch* (a negative metric, where a larger value is worse) was significant, assuming both metrics are equally weighted. Figure 2.29 shows these performance improvements normalised to the best performing configuration at 100%, e.g. a measured result of [5, 10, 30, 50] would be represented as [10%, 20%, 60%, 100%].

Figure 2.29: Graphs of various publications that varied tail design or control, normalised to the best performing variation (= 100%) for each metric, for a unitless comparison.

2.4.3 Bio-Inspiration in Robotic Tails for Terrestrial Mobile Robots

The selected publications contained varying degrees of bio-inspiration, which can be fitted into a hierarchy as shown in figure 2.30. Publications with the least influence mentioned an animal in their research only in a vague fashion, mainly as an introduction to their research, possibly to provide loose justification for their approach. Other publications did take this one step further, and made direct observations on the motion of the animal tail in order to closely approximate them in a robot. For example, **patel2013rapid**, **patel2014rapid**, **patel2015conical** all observe the tail motion of a cheetah when performing certain actions (accelerating, decelerating and turning), and while they do not measure the exact trajectory of those observations, they use a similar motion for their robot tail when the robot is performing the same action. A more intensive approach does take data from an animal performing an action, and compares the data to a robotic equivalent. For example, **libby2012tail** makes a direct comparison between the body rotation sensitivity from a perturbation between an Agama Lizard, a robot, and a simulated Velociraptor. What has not been observed in the selected literature is an attempt to replicate the anatomical structure of an animal tail using mechanical components. This is likely due to a number of reasons, complexity of design and control, redundant actuation, and technological limitations among others. While this approach has been conducted before **nyakatura2019reverse**, it has not been seen in the selected literature in this field. This kind of intensive bio-inspiration could be useful in the field of robotic tails in order to explore the potential for performance improvements, and could also help answer questions about why naturally evolved tails are generally more complex and redundant than their robotic counterparts.

Examination

Publication examines the anatomical structure of part of the animal, and uses this as a basis for a mechanical design element in the robot.

Comparison

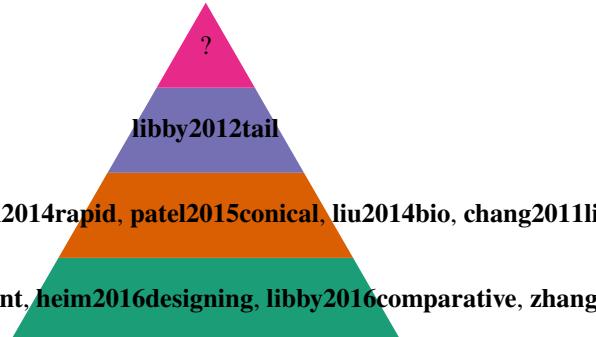
Publication takes measurements of the animal in motion, and compares the performance for specific criteria during a given task with the robot.

Observation

Publication sources visual observations on biomechanics **patel2013rapid, patel2014rapid, patel2015conical, liu2014bio, chang2011liu**, and approximates said motions using the robot.

Inspiration

Publication mentions **takita2002development, sato2016development, heim2016designing, libby2016comparative, zhang2016tail** any data from that animal in the research.



Note: **briggs2012tails** does use *observation* for the simulation, but does not for the experimental work.

Figure 2.30: Hierarchy of bio-inspiration with examples from the selected literature, from least to most influential on the research.

2.4.4 Use of Multiple Segment Tails

Only a couple of publications **simon2018development, sato2016development** experimented with mobile robots using tails with multiple segments. Other publications **rone2018design, saab2018discrete, saab2018modeling, rone2014continuum, liu2019cable, liu2020design** only experimented with multi-segment tails in isolation, some with speculation on future inclusion in a mobile robot. As mentioned in subsections 2.4.2 and 2.3.2.6, **sato2016development** managed to achieve a 25 mm improvement in jump height using a sprung flexible tail instead of a rigid one. The relative paucity of experimental multi-segment tails could be explained due to complexities in dynamics and control, as even in **sato2016development** only the first joint was actuated and the rest passively spring-loaded, resulting in a simple design and control system. **simon2018development** attempted the same objective with a continuum cable driven design with flexible tail segments, but was unable to realise an improvement in jump height, with the extra mass of the tail resulting in a lower jump height over no tail at all. If the future research of some isolated multi-segment tail experiments is realised, then more data will be available.

2.5 Conclusion

This chapter has analysed a curated selection of the literature on terrestrial mobile robots with robotic tails from a wider systematic review, focused around experimental papers that generated results from physical prototypes. It has discovered a commonly occurring set of tail functions that assist in the dynamics of the robot, in the air, on the ground, and transitioning between the two environments. Some publications used bio-inspiration, such as the kangaroo and cheetah for hopping and turning in **liu2014bio, patel2015conical** and some did not, such as the addition of a tail to insect like hexapod robots in **pullin2012dynamic, kohut2013precise** for inducing steering, which does not occur in nature. It has discovered by far the most common design for a tail is a rigid single segment with 1-2 degree of freedom, unlike in the animal kingdom where multi-segment tails are the norm. However,

one publication that did use multi-segment tails did find an improvement in performance **sato2016development**, even when only using passive spring joints for all segments except the first.

There are still several gaps in this research field, which has very limited experimental studies into multiple segment tails on mobile robots, and no quantitative research in the use of a tail for controlling robot dynamics when carrying a payload. **mu2018multi** does use a hexapedal robot with a tail to carry a payload, but does not indicate how effectively, if at all, the tail contributed to the robot stability.

While simulation studies **iwamoto2015jumping**, **iwamoto2018distributed**, **rone2014continuum**, **rone2015static**, **liu2021dynamic** and studies with tails in isolation have been conducted **rone2018design**, **saab2018discrete**, **saab2018modeling**, **rone2014continuum**, **liu2019cable**, **liu2020design** using multi-segment tails, there is still a relative paucity of experimental research, the only notable result still using mostly passive joints. This is most likely down to technical limitations, the additional mass and bulk of additional actuators is more of an issue on a mobile robot than a static tail or simulation study. There is also the increase in control complexity, which will make stable control of the tail more of a challenge in most of the functions discussed. The development of a compact, light 2 degree of freedom actuator could assist in making multi segment tails on mobile robots more feasible. A proposed actuator based on the twisted string actuator was investigated in chapter 3. A simulation study was also conducted in chapter ??, in order to see if any improvement can be gained using a multi-segment tail for the specific application in this thesis.

The dynamics of picking up, carrying, and setting down a payload pose a unique challenge for a robotic tail. However the literature has proven that using a robotic tail for maintaining stability, by keeping the robot body at a certain angle in the air, or preventing the robot toppling over on the ground, is viable. A closed loop control system that is able to measure the stability of the robot and control the tail with said measurement during the transport of a payload as feedback is structurally similar to a system that uses body angle or acceleration, both of which have been described in the literature. In chapter ??, a measurement system that does this based on four load cells was devised, and a closed loop controller is connected to a robotic tail.

Chapter 3

A Novel Triad Twisted String Actuator for Controlling a Two Degrees of Freedom Joint: Design and Experimental Validation

Actuated universal joints, or equivalent joint systems, are found in a number of robotic applications, in particular mobile snake robots, snake-arm robots and robotic tails. Such joints use a variety of actuation methods, including direct drive motors, linear screw drives, cable based systems, and hydraulics/pneumatics. In this paper the authors present a novel design for a mechanism that utilises the twisted string actuator in an antagonistic triad to actuate the universal joint, using orientation sensors and load cells to create a robust cascading closed loop control system. This design realises a light, compact, high-performance actuation system that avoids the extra mass and hardware complexity that alternative actuation methods present, with the additional challenge of nonlinearity.

3.1 Introduction

Actuated universal joint mechanisms are found in a wide range of robotic applications, such as confined space inspection using continuum robots **buckingham2012nuclear**, highly manoeuvrable mobile snake robots **luo2018**, and biomimetic robot tails for stability **rone2018**. Mobile snake robots must usually incorporate electric actuators inline with their joints. This results in an actuated universal joint having to shift the mass of the follower segments and all the actuators inside the follower segments, which results in high torque requirements. Continuum robots and robotic tails can reduce the mass and size of the actuated universal joint by moving their actuators away from the actuated universal joints and use cables to transfer the force to the joints, or use hydraulic or pneumatic actuators which tend to be lighter than equivalent electric motors. This comes at the expense of increased mass and bulk at the base of the arm or tail.

First developed by **wurtz2010** **wurtz2010** in **wurtz2010**, the twisted string actuator uses two or more strings between two fixtures as a linear actuator. When one fixture is rotated (typically by an electric motor), the looped string twists into a helix, decreasing the distance between them. twisted string actuator actuators have been used for a hand orthosis **muehlbauer2021twisted**, elbow joint **park2020control** and foldable robot arm **suthar2021design** among other functions.

The primary advantage of twisted string actuator over similar linear actuators such as a leadscrew is the reduction (lower velocity, higher torque) the twisted string actuator provides is not proportional to the mass of the actuator, in fact it is slightly inversely proportional. Generally, to increase the reduction in an actuator requires the addition of a gearbox which increases mass, but in the case of the twisted string actuator, by decreasing the string cross-section radius, the reduction increases given a constant unwound length and motor angle, resulting in a greater reduction with no increase, or even a slight decrease, in actuator mass.

While the reduction in a leadscrew can be increased by decreasing the lead on the thread, which also has no increase in mass, this has a limited range and can quickly run up against manufacturing tolerances or material strength requirements. In order to achieve greater or more robust reductions, the screw radius has to be increased, or the driving motor has to have a larger reduction before driving the screw, both of which usually result in more material (typically steel) and therefore more mass.

However, twisted string actuator does have some disadvantages, the most significant of which is a nonlinear reduction equation, which is also dependent on the motor angle (and therefore actuator position). The reduction decreases in a nonlinear fashion as the angle increases, with the derivative decreasing as the angle increases. There is also the compliance of the strings to consider, depending on the thickness and material chosen, which becomes a sig-

nificant factor under high forces. Both of these issues can be addressed with accurate modelling **nedelchev2020** and/or a robust control strategy, as demonstrated in **wurtz2010**. What is more of an issue is the unidirectional force of the twisted string actuator, which can only impart force in tension. This means that for an actuated universal joint, which is a 2 degree of freedom joint, a minimum of three twisted string actuator are required, unless spring return mechanisms are used, which would impart additional force on the twisted string actuator and therefore reduce performance. However, the potential high force to mass ratio of the twisted string actuator due to the non-proportional reduction may adequately compensate for the additional actuator requirement.

The focus of this research is to investigate if the twisted string actuator is a suitable candidate for control of an actuated universal joint considering both the benefits and drawbacks. To this end, the objective is to simulate a model and then construct a physical experimental prototype to validate the proposed control system.

~~Model coefficients. Coefficient Value Coefficient Value l_1 J l_2 K_L r f_{\min} l_u ω_s r_s I_s m K_t C τ_s α_s Coefficient I~~ The use of twisted string actuator as an actuator for an actuated universal joint is an understudied area of research. **bombara2022physics** have proposed a similar design using a flexible core with continuous curvature as opposed to a rigid universal joint, however currently this research has not demonstrated control of both axes of motion with multiple twisted string actuator. For the first time the authors demonstrate a robust closed loop control of an actuated universal joint in both axes of motion using three twisted string actuator in an “antagonistic triad” configuration. The result is a light, compact actuated universal joint design that has the potential to significantly improve upon exiting inline actuation options.

3.1.1 Twisted String Actuator

This publication will first give an outline of the twisted string actuator based on the existing literature, and the concept of an antagonistic triad. Then a detailed explanation of the control system is given, followed by results from the simulation and experimental system. Finally, a theoretical analysis of the twisted string actuator actuated universal joint compared to a similar actuated universal joint using leadscrews is conducted, followed by a discussion and conclusion.

This work was previously published as a conference proceeding at the International Conference on Robotics and Automation 2022 **crosby2022novel**. This publication extends that research by including additional experimental results with increased follow mass and joint velocity, as well as adjusting a parameter in the control system to increase the joint range. It also details additional preliminary work that was done to characterise each individual twisted string actuator under force control, including exploration of alternative twisted string actuator inner loop control systems. Finally, it adds additional content to the discussion and proposes further improvements to the design for future work.

3.1.1 A Twisted string actuator

First developed by [wurtz2010](#) in 2010, twisted string actuator uses two or more strings between two fixtures as a 1 degree of freedom linear actuator. When one fixture is rotated (typically by an electric motor), the strings twist into a helix, decreasing the distance between the fixtures, as shown in figure [??](#). Given the unwound length l_u , and the cross-section radius of the string r_s (or $r_s + r_c$ when there are more than two strings) as shown in figure [??](#), the actuator length is given by

$$l_s(\theta_s) = \sqrt{l_u^2 - \theta_s^2 r_s^2} \quad (3.1)$$

where θ_s is the motor angle, as shown in figure [??](#).

$$\theta_s = 0 \quad \theta_s = 2\pi \quad \theta_s = 20\pi$$

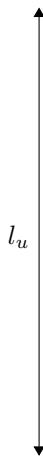


Figure 3.1: The value of θ_s increases the number of twists in a string bundle with a string length l_u .

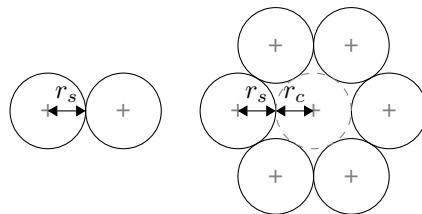


Figure 3.2: The location of r_s and optionally r_c in a string bundle.

This equation assumes an infinite string stiffness, so is only reasonably accurate under low tension. Although theoretically the stroke of the twisted string actuator can be the entire domain of $[0, l_u]$, in reality the thickness of the string prevents a geometric helix from forming once the helix pitch $q < 4r_s$ (or $q < 2nr_s$ for n strings) as mentioned in [wurtz2010](#). This limits the lower bound of the stroke as follows,

$$l_{\min\min} = \frac{l_u}{\sqrt{\frac{\pi^2}{2} + 1}} \approx 0.46 l_u \quad (3.2)$$

or approximately 46% of l_u for a two string twisted string actuator.

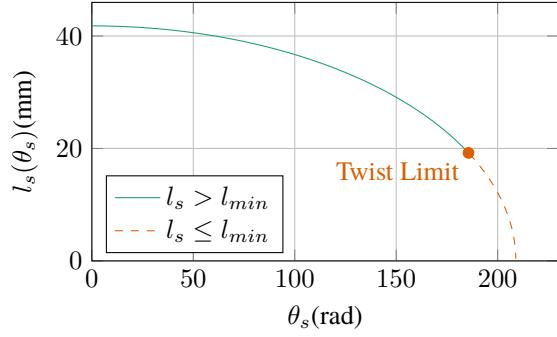


Figure 3.3: twisted string actuator string length against motor angle with coefficients from table ??.

3.1.2 Antagonistic Triad

As mentioned in the introduction, because the twisted string actuator provides only tensile force, a minimum of three actuators are required for a 2 degree of freedom actuation system. These can be arranged in a triangular configuration to create an “antagonistic triad”, akin to the antagonistic pairs of muscles found in animals. ~~In a pair arrangement, one actuator contracts while the other relaxes, but in a triad, up to two actuators can share the same action, and one actuator can be inactive if the other two are performing different actions. Where a revolute joint would be found between the connecting ends of the actuator, a universal joint is found instead.~~ The geometric structure of the system as shown in figure ?? can be described with two equilateral triangles of ~~inradius exradius~~ r on two planes separated in the z axis. The centroids are then connected via a universal joint from each plane normal to an intersecting point, let the vector $\theta = [\theta_1 \ \theta_2]$ denote the rotation of the second plane relative to the first, in the y and x and y axes around the intersecting point, and let l_1 and l_2 denote the normal distance from the intersection to the first and second plane centroids respectively. When $\theta = [0 \ 0]$ the triangles are parallel to each other. The distance between the vertex pairs of each triangle is then denoted as $[\lambda_1 \ \lambda_2 \ \lambda_3]$ for the “top”, “left” and “right” vertices of the triangles. When θ is changed, this will change λ_1 , λ_2 and λ_3 respectively.

To calculate the lengths of the strings for a given θ of the universal joint, ~~we define a transformation matrix for each string~~

$$\begin{aligned}
R_1(\theta_1) &= \begin{bmatrix} \cos \theta_1 & 0 & \sin \theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_1 & 0 & \cos \theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
R_2(\theta_2) &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & \cos \theta_2 & -\sin \theta_2 & 0 \\ 0 & \sin \theta_2 & \cos \theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
P_l(l) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
P_r &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
R_r(\phi) &= \begin{bmatrix} \cos \phi & -\sin \phi & 0 & 0 \\ \sin \phi & \cos \phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{3.3}$$

$$T_n(\boldsymbol{\theta}) = P_l(l_1) R_1(\theta_1) R_2(\theta_2) P_l(l_2)$$

$$T_1(\boldsymbol{\theta}) = P_r^{-1} T_n(\boldsymbol{\theta}) P_r$$

$$T_2(\boldsymbol{\theta}) = R_r\left(\frac{2\pi}{3}\right) P_r^{-1} R_r\left(-\frac{2\pi}{3}\right) T_n(\boldsymbol{\theta}) R_r\left(\frac{2\pi}{3}\right) P_r$$

$$\underline{T_3(\boldsymbol{\theta}) = R_r\left(-\frac{2\pi}{3}\right) P_r^{-1} R_r\left(\frac{2\pi}{3}\right) T_n(\boldsymbol{\theta}) R_r\left(-\frac{2\pi}{3}\right) P_r}$$

can be computed, which follows a path from one vertex to the other via the plane normals. This matrix incorporates the universal joint rotations $R_1(\theta_1)$ and $R_2(\theta_2)$, the normal distance translations $P_l(l_1)$ and $P_l(l_2)$, the triangle exradius translation P_r and triangle exradius rotation $R_r(\phi)$, where $\phi = [0, \frac{2\pi}{3}, -\frac{2\pi}{3}]$ to represent the three vertices of the triangle, hereafter referred to as *top*, *left* and *right*.

Then, to get the distances between the vertex pairs, the euclidean norm for the position components of each transformation matrix \mathbf{T} can be calculated. These distances can then be combined into a vector function $\Lambda(\boldsymbol{\theta}) = [\lambda_1(\boldsymbol{\theta}) \ \lambda_2(\boldsymbol{\theta}) \ \lambda_3(\boldsymbol{\theta})]$ as follows. where

$$\ell(\mathbf{T}) = \left\| \begin{bmatrix} t_{14} & t_{24} & t_{34} \end{bmatrix} \right\|_2$$

$$\Lambda(\boldsymbol{\theta}) = \begin{bmatrix} \ell(T_1(\boldsymbol{\theta})) & \ell(T_2(\boldsymbol{\theta})) & \ell(T_3(\boldsymbol{\theta})) \end{bmatrix}$$

$$= \begin{bmatrix} \lambda_1(\boldsymbol{\theta}) & \lambda_2(\boldsymbol{\theta}) & \lambda_3(\boldsymbol{\theta}) \end{bmatrix}$$

where:

$$\lambda_1(\boldsymbol{\theta}) = \sqrt{l_1^2 + 2l_1l_2 \cos(\theta_1) \cos(\theta_2) + 2l_1r \sin(\theta_2) \cos(\theta_1) + l_2^2 + 2l_2r \sin(\theta_2) - 2r^2 \cos(\theta_2) + 2r^2}$$

$$\lambda_2(\boldsymbol{\theta}) = \sqrt{(a-b+c)^2 + (l_1-d)^2 + e^2} \sqrt{a+b+c-d} \quad (3.4)$$

$$\lambda_3(\boldsymbol{\theta}) = \sqrt{(a+b-c)^2 + (l_1+d)^2 + e^2} \sqrt{a-b-c+d}$$

where:

$$a = -l_1^2 + 2l_1l_2 \cos(\theta_1) \cos(\theta_2)$$

$$b = \sqrt{3}l_1r \sin(\theta_1) - l_1r \sin(\theta_2) \cos(\theta_1) + l_2 \cos \theta_2 \sin \theta_1^2$$

$$c = d = +\sqrt{3}l_2 \cos \theta_1 \cos \theta_2 r \sin(\theta_1) \cos(\theta_2) - l_2r \sin(\theta_2)$$

$$ed = \frac{\sqrt{3}r^2 \sin(\theta_1) \sin(\theta_2)}{2} - \frac{3r^2 \cos(\theta_1)}{2} - \frac{r^2 \cos(\theta_2)}{2} + l_2 \sin \theta_2 2r^2.$$

Surface plots of the functions in (??) are shown in figure ?? for a domain of $[-\frac{\pi}{2}, \frac{\pi}{2}]$.

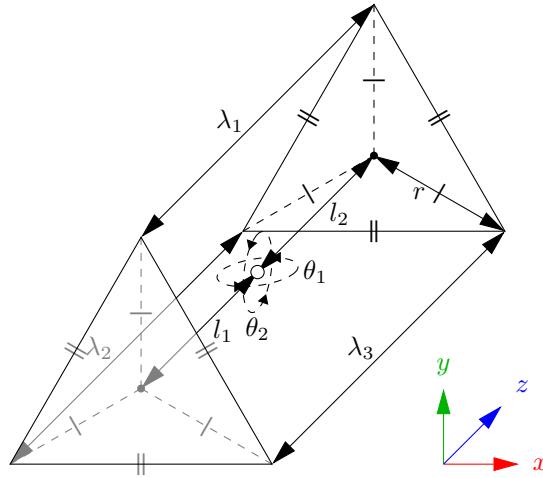


Figure 3.4: Kinematic diagram of the antagonistic triad, where the universal joint rotation is defined by $\theta_{1,2}$ on the y and x axes respectively, and the actuator lengths are defined by $\lambda_{1,2,3}$ for the **top**, **left**, and **right** actuators. r and $l_{1,2}$ define the anchor points of the actuators.

3.2 Control System

The initial experiments with a position based control system proved unsuccessful, a

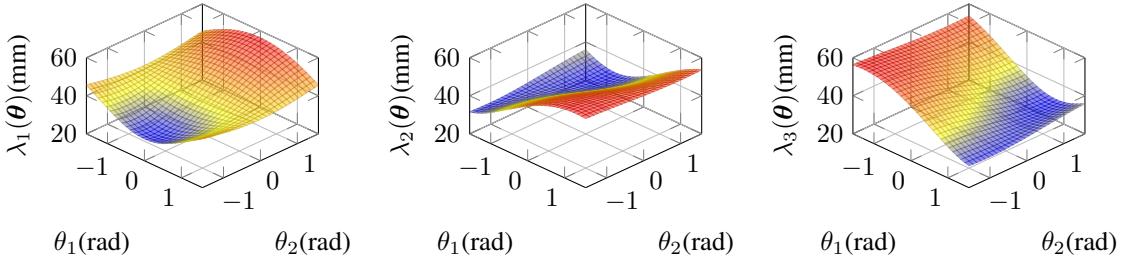


Figure 3.5: Surface plots of each element of the vector function $\Lambda(\theta)$, assuming coefficient values from table ???. Note that λ_2 and λ_3 are symmetric.

force based control system was designed, which uses the dynamics of the actuated universal joint to convert angular velocity from a PID controller into angular torque, which is then turned into force setpoints for each twisted string actuator using an optimising algorithm. Each twisted string actuator has a tensile force when $l_s(\theta_s) < \lambda_n(\theta)$, which is calculated as the difference in triad length and twisted string actuator contraction length multiplied by the load stiffness $(\lambda_n(\theta) - l_s(\theta_s)) K_L$. When all of the twisted string actuator tensile forces are equal, there is no torque on the actuated universal joint, but when they are unequal, a torque is generated, and if these torques do not match the torques required by the current dynamics to maintain the actuated universal joint orientation, the actuated universal joint will rotate. As the tensile force is measured using load cells, the control system does not need to know the value of K_L since a setpoint force can be maintained using a P controller.

This control system is a four layer cascade design, joining an inverse dynamic control system spong2020robot, to the triad force controller in dessen1986, to a proportional controller for each twisted string actuator. It uses feedback signals of the joint position from the accelerometers and twisted string actuator force from the load cells. A second order setpoint trajectory q is used as the input, which can either be pre-defined or generated dynamically from user input. Feedback is provided by the actuated universal joint angular position θ as shown in figure ??, angular velocity $\dot{\theta}$, and twisted string actuator tension force \hat{f} . Figure ?? shows a complete block diagram of the control system.

The control system can be defined by four functions as follows:

1. C_1 actuated universal joint Position PID Controller with Acceleration Feedforward
2. C_2 Inverse Dynamics
3. C_3 twisted string actuator Force Optimisation Algorithm
4. C_4 twisted string actuator Force P Controller

Functions $C_{1..4}$ are then combined into a cascade function $C_4(C_3(C_2(C_1(\dots), \dots), \dots), \dots)$.

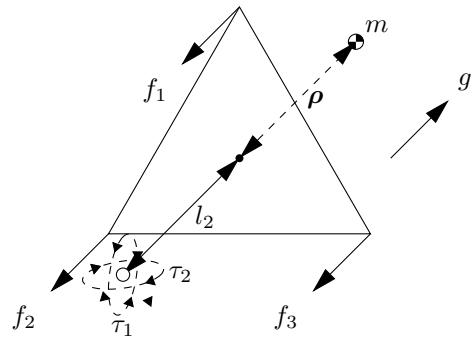


Figure 3.6: Dynamics of an antagonistic triad based on figure ??, where the follower mass is m , the follower center of mass relative to the universal joint pivot is ρ , the forces on each anchor point are $f_{1,2,3}$ and the universal joint torque is $\tau_{1,2}$.

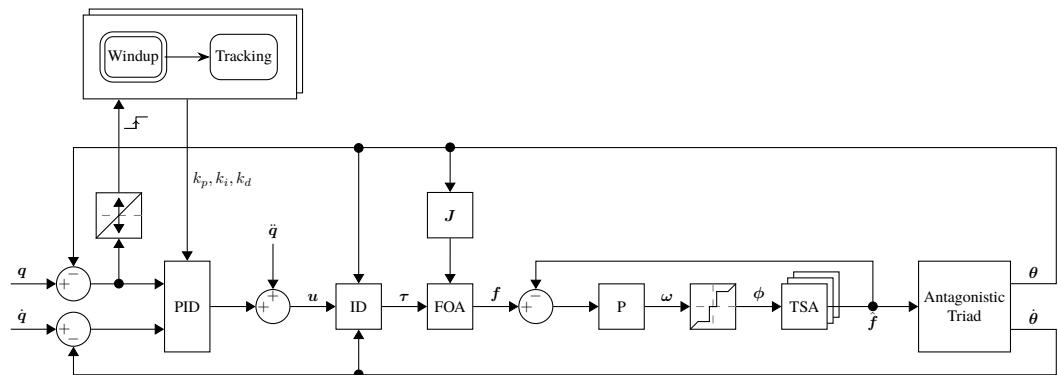


Figure 3.7: Block diagram of the complete experimental control system, excluding the hardware velocity controllers for the motors.

3.2.1 AUJ Position PID Controller with Acceleration Feedforward

Firstly, a PID controller is used to generate a control signal u with the input q as the set-point, and the actuated universal joint angular position θ and velocity $\dot{\theta}$ as feedback, plus the addition of a feedforward term for the input acceleration \ddot{q} , i.e.

$$u = k_p (q - \theta) + k_i \left(\int_0^t (q - \theta) dt \right) + k_d (\dot{q} - \dot{\theta}) + \ddot{q}.$$

In the discrete implementation used for fixed step simulation and experimental system control, the integral term is replaced by the trapezoidal rule.

3.2.2 Inverse Dynamics

The control signal u from the PID controller is then converted to the desired actuated universal joint torque τ as follows. This is achieved using the Euler-Lagrange formulation which takes into account the dynamic properties of the actuated universal joint in its current state, as shown in figure ?? . For simplicity, the mass of the universal joint is ignored and the value of ρ only has a z component, ρ_3 , which is included in the experiments in section

$$\begin{aligned}
T^\tau(\boldsymbol{\theta}) &= R_1(\theta_1) R_2(\theta_2) P_t(l_2 + \rho_3) \\
J_v(\boldsymbol{\theta}) &= J([t_{14}^\tau \quad t_{24}^\tau \quad t_{34}^\tau]) = \begin{bmatrix} (l_2 + \rho_3) \cos \theta_1 \cos \theta_2 & (-l_2 - \rho_3) \sin \theta_1 \sin \theta_2 \\ 0 & (-l_2 - \rho_3) \cos \theta_2 \\ (-l_2 - \rho_3) \sin \theta_1 \cos \theta_2 & (-l_2 - \rho_3) \sin \theta_2 \cos \theta_1 \end{bmatrix} \\
J_\omega(\boldsymbol{\theta}) &= J \left(\dot{\theta}_1 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \dot{\theta}_2 R_2(\theta_2)^r \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) = \begin{bmatrix} 0 & \cos \theta_1 \\ 1 & 0 \\ 0 & -\sin \theta_1 \end{bmatrix} \\
D(\boldsymbol{\theta}) &= m J_v^\top J_v + J_\omega^\top R_1(\theta_1)^r R_2(\theta_2)^r I (R_1(\theta_1)^r R_2(\theta_2)^r)^\top J_\omega \\
&= \text{diag} \left(\begin{bmatrix} I_{22} \cos^2 \theta_2 - I_{33} \cos^2 \theta_2 + I_{33} + l_2^2 m \cos^2 \theta_2 + 2l_2 m \rho_3 \cos^2 \theta_2 + m \rho_3^2 \cos^2 \theta_2 \\ I_{11} + l_2^2 m + 2l_2 m \rho_3 + m \rho_3^2 \end{bmatrix}^\top \right) \\
C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) &= C^{k \times j} \mid c_{k,j} = \sum_{i=1}^N \frac{1}{2} \left(\frac{\partial d_{kj}}{\partial \theta_i} + \frac{\partial d_{ki}}{\partial \theta_j} - \frac{\partial d_{ij}}{\partial \theta_k} \right) \dot{\theta}_i \\
&= \begin{bmatrix} \frac{(-I_{22} + I_{33} - l_2^2 m - 2l_2 m \rho_3 - m \rho_3^2) \sin(2\theta_2) \dot{\theta}_2^2}{2} & \frac{(-I_{22} + I_{33} - l_2^2 m - 2l_2 m \rho_3 - m \rho_3^2) \sin(2\theta_2) \dot{\theta}_1^2}{2} \\ \frac{(I_{22} - I_{33} + l_2^2 m + 2l_2 m \rho_3 + m \rho_3^2) \sin(2\theta_2) \dot{\theta}_1^2}{2} & 0 \end{bmatrix} \\
P(\boldsymbol{\theta}) &= mg T^\tau \\
G(\boldsymbol{\theta}) &= J(p_{34}) = [-gm(l_2 + \rho_3) \sin \theta_1 \cos \theta_2 \quad -gm(l_2 + \rho_3) \sin \theta_2 \cos \theta_1]
\end{aligned}$$

(3.5)

?? but set to zero for all other experiments. The torque is calculated as $\tau = D(\boldsymbol{\theta}) \mathbf{u} + C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \dot{\boldsymbol{\theta}}$ + with the functions from ??.

3.2.3 TSA Force Optimisation Algorithm

This uses a modified the *inverse force transformation* algorithm from **deissen1986**, which proposes an inverse force transformation algorithm to control an antagonistic triad using force controlled linear actuators, with the jacobian of equation ?? to select an optimal force vector from the desired joint torque. Here it is presented in an unexpanded and more general form.

$$\begin{aligned}
J_\Lambda &= \begin{bmatrix} \frac{\partial \lambda_1}{\partial \theta_1} & \frac{\partial \lambda_2}{\partial \theta_1} & \frac{\partial \lambda_3}{\partial \theta_1} \\ \frac{\partial \lambda_1}{\partial \theta_2} & \frac{\partial \lambda_2}{\partial \theta_2} & \frac{\partial \lambda_3}{\partial \theta_2} \end{bmatrix} \\
\boldsymbol{\gamma}(i) &= -J_{\Lambda_{-i,*}}^{-\top} \left(J_{\Lambda_{i,*}}^\top f_{\min} + \boldsymbol{\tau} \right) \\
\mathbf{F}(\boldsymbol{\tau}, \boldsymbol{\theta}) &= \begin{bmatrix} f_{\min} & \boldsymbol{\gamma}(2)_1 & \boldsymbol{\gamma}(3)_1 \\ \boldsymbol{\gamma}(1)_1 & f_{\min} & \boldsymbol{\gamma}(3)_2 \\ \boldsymbol{\gamma}(1)_2 & \boldsymbol{\gamma}(2)_2 & f_{\min} \end{bmatrix}.
\end{aligned}$$

(3.6)

A force matrix \mathbf{F} is created from the torque input $\boldsymbol{\tau}$, jacobian J_Λ from the vector function Λ as defined in ??, and minimum force constant f_{\min} . The diagonal components f_{ii} are equal to f_{\min} . f_{ii} is equal to f_{\min} , while the other elements in the column are based on a calculation using $J_{\Lambda_{-i,*}}$ where $-i$ is a row removed from the matrix.

$$J_{\Lambda} = \begin{bmatrix} \frac{\partial \lambda_1}{\partial \theta_1} & \frac{\partial \lambda_2}{\partial \theta_1} & \frac{\partial \lambda_3}{\partial \theta_1} \\ \frac{\partial \lambda_1}{\partial \theta_2} & \frac{\partial \lambda_2}{\partial \theta_2} & \frac{\partial \lambda_3}{\partial \theta_2} \end{bmatrix}$$

$$\gamma(i) = -J_{\Lambda_{-i,*}}^{-\top} \left(J_{\Lambda_{i,*}}^{\top} f_{\min} + \tau \right)$$

$$F(\tau, \theta) = \begin{bmatrix} f_{\min} & \gamma(2)_1 & \gamma(3)_1 \\ \gamma(1)_1 & f_{\min} & \gamma(3)_2 \\ \gamma(1)_2 & \gamma(2)_2 & f_{\min} \end{bmatrix}$$

The following algorithm then selects one column of F to be the output force vector f , where \top and \perp are boolean *true* and *false* respectively, and $f_{*,i}$ is the i th column of F . The solution to this algorithm minimises the net force on all twisted string actuator while producing the desired output torque on the universal joint.

```

1:  $s \leftarrow [\top \ \top \ \top]$ 
2: if  $f_{23} > f_{\min}$  then  $s_2 \leftarrow \perp$  else  $s_3 \leftarrow \perp$  end if
3: if  $f_{31} > f_{\min}$  then  $s_3 \leftarrow \perp$  else  $s_1 \leftarrow \perp$  end if
4: if  $f_{12} \geq f_{\min}$  then  $s_1 \leftarrow \perp$  else  $s_2 \leftarrow \perp$  end if
5: for  $i = 1$  to  $3$  do
6:   if  $s_i \rightarrow \top$  then  $f \leftarrow f_{*,i}$  end if
7: end for
```

3.2.4 TSA Force Proportional Controller

The selected forces are then used as an input to a P controller with gain k_{ps} using the measured load cell forces \hat{f} as feedback. The output from this can then be used to control the top, left and right twisted string actuator motors, corresponding to the actuators in figure ??.

3.2.4.1 Simulation Current Control

3.3 Simulation Design

To design and refine the parameters of the control system, a Simscape Multibody™ model of the antagonistic triad and control system was created in MATLAB®/Simulink™. This allowed for model design coefficients $l_{1,2}$ and controller gains k_p, k_i, k_d, k_{ps} to be modified in order to have the most stable control within design limits.

In the simulation, each twisted string actuator was modelled as a state-space system

$$\begin{aligned}
h(\theta_s) &= \frac{\theta r_s^2}{\sqrt{l_u^2 - \theta_s^2 r_s^2}} \\
k(\theta_s, \boldsymbol{\theta}) &= \lambda_n(\boldsymbol{\theta}) - \sqrt{l_u^2 - \theta_s^2 r_s^2} \\
\dot{\mathbf{x}} &= \begin{bmatrix} x_2 \\ -\frac{K_L}{J} h(x_1) k(x_1, \boldsymbol{\theta}) - \frac{C}{J} \operatorname{sgn}(x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_t}{J} \end{bmatrix} u \\
y &= K_L \operatorname{sat}_0^\infty k(x_1, \boldsymbol{\theta}),
\end{aligned} \tag{3.7}$$

modified from **wurtz2010**, which takes motor current u as an input and outputs y as the twisted string actuator tension force. **wurtz2010 defines it as such**, where J is the motor inertia, C is the motor coulomb friction (modified from viscous friction as the motor only has dry friction), K_t is the motor torque constant, and K_L is the load stiffness. As the original definition is for a fixed load l_u distance from the motor a modified model is required which takes into account the varying length between the motor and load defined by $\Lambda(\boldsymbol{\theta})$.

A saturation function, with the compact notation $\operatorname{sat}_x^y z = \max(x, \min(y, z))$ is used to prevent incorrect compression forces when the string is slack. All of the motor coefficients were taken from the Faulhaber 1724TSR datasheet **1724tsr** as this is the motor used in the experimental model. An estimated value is used for the load stiffness, this was chosen to be a high number as the model is expected to be very stiff.

$$\begin{aligned}
h(\theta_s) &= \frac{\theta r_s^2}{\sqrt{l_u^2 - \theta_s^2 r_s^2}} \\
k(\theta_s, \boldsymbol{\theta}) &= \lambda_n(\boldsymbol{\theta}) - \sqrt{l_u^2 - \theta_s^2 r_s^2} \\
\dot{\mathbf{x}} &= \begin{bmatrix} x_2 \\ -\frac{K_L}{J} h(x_1) k(x_1, \boldsymbol{\theta}) - \frac{C}{J} \operatorname{sgn}(x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_t}{J} \end{bmatrix} u \\
y &= K_L \operatorname{sat}_0^\infty k(x_1, \boldsymbol{\theta})
\end{aligned}$$

The state space model was then adapted to include constraints on motor velocity and acceleration set by the motor controller in order to keep the motor within design limits, **by replacing**

$$\dot{\mathbf{x}}' = \begin{bmatrix} \operatorname{sat}_{-\omega_s}^{\omega_s} \dot{x}_1 \\ \operatorname{sat}_{-\alpha_s}^{\alpha_s} \dot{x}_2 \end{bmatrix}, \tag{3.8}$$

which replaces $\dot{\mathbf{x}}$ with $\dot{\mathbf{x}}'$ ~~which contains~~, which contain saturation functions for maximum motor velocity v_s and acceleration α_s .

$$\dot{\mathbf{x}}' = \begin{bmatrix} \operatorname{sat}_{-\omega_s}^{\omega_s} \dot{x}_1 \\ \operatorname{sat}_{-\alpha_s}^{\alpha_s} \dot{x}_2 \end{bmatrix}$$

3.3.0.1 Experimental Velocity Control with Deadband Compensation

Due to a controller deadband within $\pm h$, an adjustable deadband compensator is used,

$$\phi_i(\omega_i) = \begin{cases} 10 & 10 \leq \omega_i < h \\ -10 & -h < \omega_i \leq -10 \\ 0 & h \leq \omega_i \leq -h \\ \omega_i & \text{otherwise} \end{cases}$$

where ϕ_i is the compensator for the controller i . An adjustment value $h \in [0, 10]$ changes the threshold at which the compensator switches on and off, allowing for a small deadband to remain.

The result from this is then a compressive force acting between each of the three and its corresponding endpoint on the Antagonistic Triad, imparting a torque on the axes of the universal joint. Based on a combination of simulation modelling, motor parameters, and physical design constraints, the model parameters were set to the values in table ???. l_2 was set to zero, as this produced the most stable control system in simulation.

3.4 Simulation & Experimental Results

Table 3.1: Model coefficients.

Coefficient	Value	Coefficient	Value
l_1	41.8 mm	J	$1 \times 10^{-6} \text{ kg m}^{-2}$
l_2	0 mm	K_L	1000 N m^{-1}
r	13 mm	f_{win}	3 N
l_u	41.8 mm	ω_s	441.9 rad s^{-1}
r_s	200 μm	I_s	0.19 A
m	72.619 13 g	K_t	$0.0263 \text{ N m A}^{-1}$
C	0.1315 N mm	τ_s	4.5 mN m
α_s	$1 \times 10^5 \text{ rad s}^{-2}$	ρ	[0 0 0] mm

Coefficient	Value		
I	3×10^{-5}	0	0
	0	3.2×10^{-5}	0
	0	0	1.4×10^{-5}

Table 3.2: PID gains in the simulation and experiment.

Gain	Value		
	Simulation	Simulation	Experiment*
k_p	800		3×10^4
k_i	3000		350
k_d	50		50
k_{ps}	19		100

* Tracking mode, see section ??.

3.4.1 Experimental Setup

For the experimental validation, a physical prototype of the mechanism was constructed with coefficients from table ?? as design parameters. This was mounted vertically, in order for the inertial measurement unit to measure the orientation of the universal joint. The twisted string actuator mechanisms consist of a compact high torque motor attached to the base segment, with a string clamp attached to the motor shaft. On the follower segment, a load cell is mounted on top of a universal joint to ensure a purely axial load, with a capstan bolt attached to the load cell. The string itself is attached to the clamp at both ends using two grub screws for extra security and easy adjustment, and looped through the hole in the capstan bolt. The total mass of the prototype, excluding the mount, is ≈ 176 g. Figure ?? details Figures ?? and ?? detail the construction of the experiment with all the constituent parts.

3.4.1 Windup & Tracking States

The motors were controlled by Fauhaber MCDC3002 motor controllers, which could interface with a National Instruments MyRIO via the USB port, using a USB to serial converter. The load cells were Futek LCM100 miniature load cells, selected for their small size. The signals from these were amplified using Flyde FE-359-TA instrumentation amplifiers and decoded using an external AD7606 ADC before being fed into the MyRIO using SPI. The orientation of the actuated universal joint was measured using a Bosch Sensortec BNO080 inertial measurement unit using the accelerometer data.

3.4.1 Force Proportional Controller & Motor Characterisation

Before verifying robust control of the actuated universal joint, the control of each individual twisted string actuator needed to be tested in order to characterise the performance of the motors and to ensure the inner loop control system was robust. This would involve selecting the control strategy that gave the best performance. A test trajectory consisting of a smooth ramp followed by a sine wave was fed into the inner loop of the cascade function as $[f(t) \dots f(t) \dots f(t)]$. Each control strategy was tested, and in the end the velocity control strategy proved most optimal, as is shown in figure ?? . The results of a sinusoidal force trajectory for the velocity control strategy is shown in figure ?? . As can be seen in the inset, there is a stick-slip

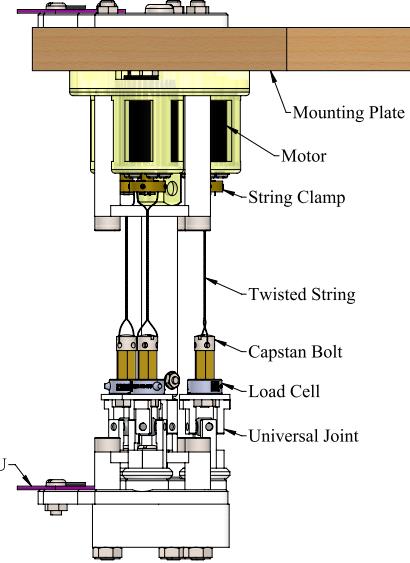


Figure 3.8: Schematic of the experimental system with labelled components.

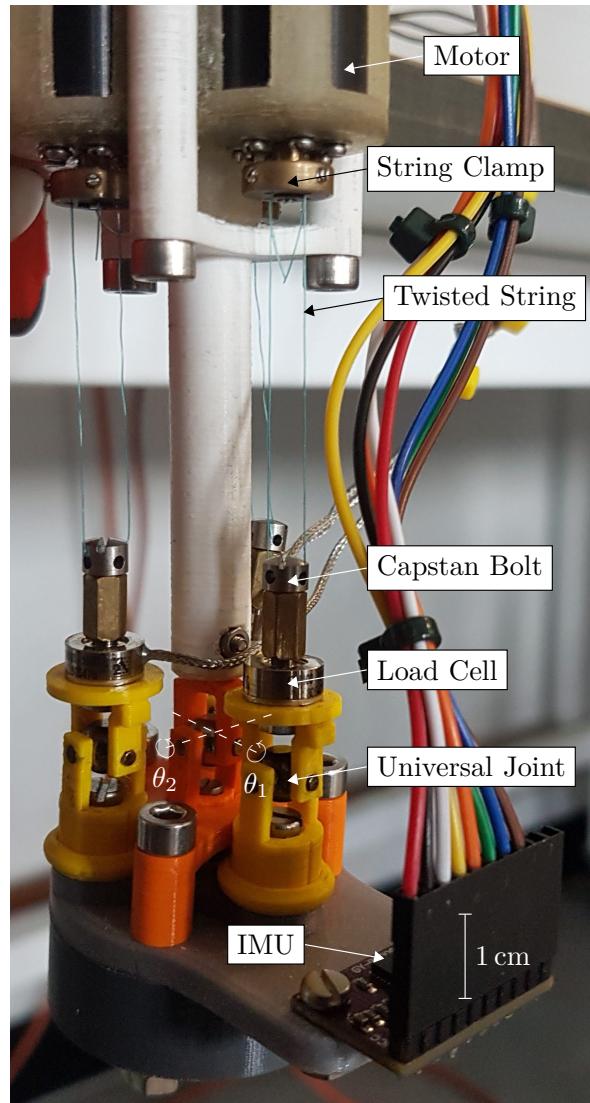


Figure 3.9: Annotated photograph of the experimental model, with the roll θ_1 and pitch θ_2 axes marked.

friction effect which causes a sawtooth like effect on the measured force. This means the motor angle has to be constantly adjusted even with a constant force input, in order to maintain

the setpoint force.

3.4.1.1 Direct current control (DCC)

This mode takes a preset velocity $\dot{\theta}_{set} \in [0, 4220]$ and uses a hardware PI controller with velocity feedback $\dot{\theta}_{act}$ to generate a control voltage. This voltage V is then multiplied by the signum of the result from the cascade function in order to ensure the motor spins in the right direction, and then passed to a current limiter with the current error (the result with the actual current I_{act} subtracted) as the limit, before being sent to the motor. This ensures the motor stops spinning when the target current is reached. V is calculated as

$$\epsilon_c = \dot{\theta}_{set} - \dot{\theta}_{act}$$

$$V = \left(K_{P_c} \epsilon_c + K_{D_c} \int_0^t \epsilon_c \right) \text{sgn}(C_4(\dots)) \omega(C_4(\dots) - I_{act}), \quad (3.9)$$

where $\omega(\dots) \in [0, 1]$ is an unknown hardware limiting function that controls the motor speed depending on the current error.

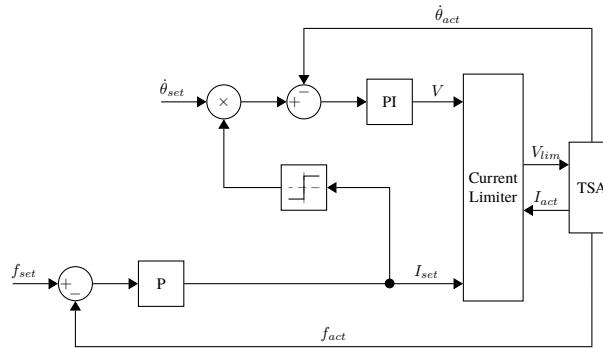


Figure 3.10: Block diagram of the DCC.

3.4.1.2 Proportional Current Controller

This strategy is a more direct method of current control, using a software P controller to directly set the voltage of the motor, using the MCDC 3002 as simply a passive amplifier. In this case the current error is passed directly to a P controller which has its output limited to prevent damage to the motors. V is calculated as

$$V = K_{P_c} C_4(\dots) - I_{act}. \quad (3.10)$$

3.4.1.3 Velocity Control

This strategy simply uses the result from the cascade function as a velocity setpoint using the hardware velocity PI controller. V is calculated as

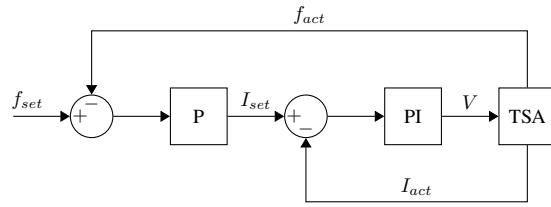


Figure 3.11: Block diagram of the proportional current controller.

$$\mathbf{V} = K_{P_c} \boldsymbol{\epsilon}_c + K_{D_c} \int_0^t \boldsymbol{\epsilon}_c. \quad (3.11)$$

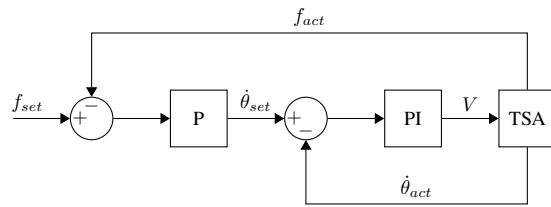


Figure 3.12: Block diagram of the velocity controller.

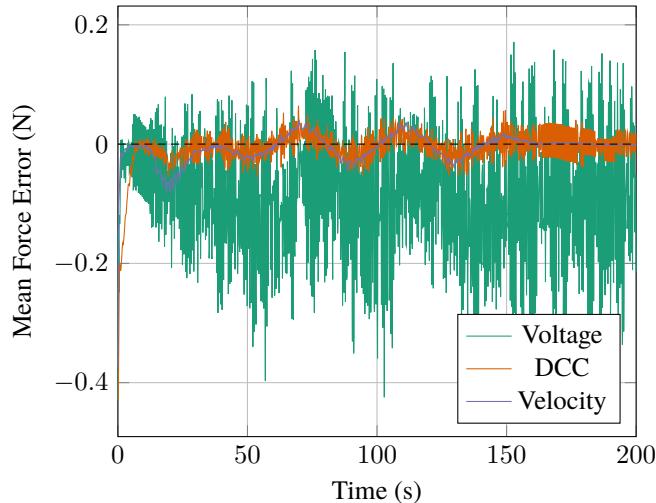


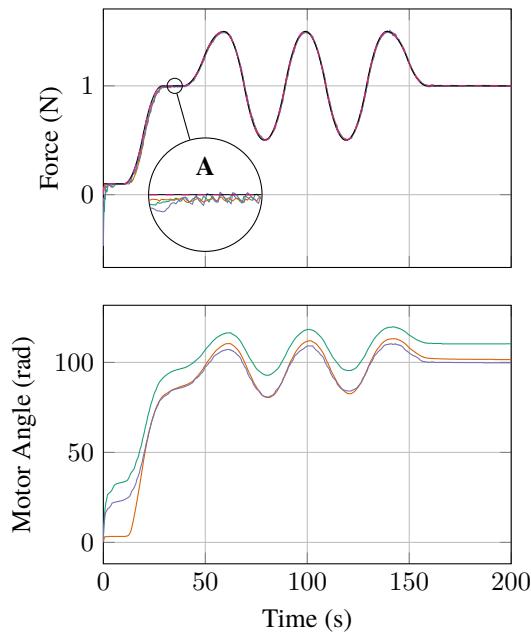
Figure 3.13: Mean tracking error for each control strategy.

3.4.2 Modifications to Control System

3.4.2.1 Experimental Velocity Control with Deadband Compensation

Due to a controller deadband within $\pm 10 \text{ min}^{-1}$, an adjustable deadband compensator is used,

$$\phi_i(\omega_i) = \begin{cases} 10 & 10 \leq \omega_i < h \\ -10 & -h < \omega_i \leq -10 \\ 0 & h \leq \omega_i \leq -h \\ \omega_i & \text{otherwise} \end{cases} \quad (3.12)$$



??

Figure 3.14: Load cell, motor angle, and motor velocity signal plot from each string using the velocity control strategy for an experimental trajectory. Inset A shows a zoomed in portion of the graph, illustrating the stick-slip phenomenon noticed in the load cell measurements.

where ϕ_i is the compensator for the controller i . An adjustment value $h \in [0, 10]$ changes the threshold at which the compensator switches on and off, allowing for a small deadband to remain.

3.4.2.2 Windup & Tracking States

The tracking controller is activated after a When the mechanism is started with the twisted string actuator in a completely unwound state, before it can begin tracking a motion trajectory, the twisted string actuator strings must “wind up” stage when the operating conditions are met. The controller parameters for the Windup and Tracking states are $k_p = 800$ to closely match the initial state of f . During this phase, the outer PID gains k_p, k_i are unsuitable and can result in damage to the mechanism. To mitigate this, two sets of PID gains are chosen, one for the windup state ($k_p = 800, k_i = 3000$, and $k_p = 3 \times 10^4$), and another for the tracking state ($k_p = 3 \times 10^4, k_i = 350$ respectively.), which the windup state transitions to once suitable stability is achieved. This transition trigger is defined on a per-axis basis, as the first zero crossing of the angle error (as $q = 0$ this is effectively θ). A graph showing the difference this state change makes to the actuated universal joint orientation is shown in figure ??.

Figure 3.15: The initial startup of the mechanism showing the transition between “windup” and “tracking” states.

Figure 3.16: Plots of the response for three different trajectories, one on only the roll axis θ_1 (column 1), one on only the pitch axis θ_2 (column 2), and one on both axes θ_1 and θ_2 (column 3). Plots include actuated universal joint orientation, forces at the top, left and right twisted string actuator, and the motor positions. Note the simulation error is very small, so the plot cannot be seen on the graph.

3.4.3 Results actuated universal joint Experiments

3.4.3.1 actuated universal joint Angle Tracking

Figure ?? plots the tracking response of both the simulation and experiment. Three trajectories were created to test the capabilities of the mechanism. Two were only in one axis of the Universal Joint, and the third was in both axes. ~~The deflection angle range was limited to \pm on a single axis,~~

3.4.3.2 The Effect of f_{\min} on AUJ Roll Range Limits

~~Ultimately the actuated universal joint range limits are determined by the angle at which one or more of the twisted string actuator are “fully unwound” where $l_s(\theta_s) = l_u$, or “fully wound” where $l_s(\theta_s) = l_{\min}$, unless there is a string failure before this limit. If f_{\min} is increased, the length of each twisted string actuator at $\theta = [0, 0]$ is reduced, as a greater difference between $\lambda_n(\theta)$ and \pm on both axes. This was partly due to the low value of l_u , but can easily be extended by increasing this value, a low l_u was chosen as it resulted in easier installation. $l_s(\theta_s)$ is needed for a larger tensile force, as described in section ???. Therefore, increasing f_{\min} should increase the actuated universal joint angle range as there is a greater difference between the length of the twisted string actuator at $\theta = [0, 0]$ and a fully unwound twisted string actuator, as long as there is sufficient length available so an antagonist twisted string actuator doesn’t become fully wound.~~

An experiment was set up where the setpoint angle of the actuated universal joint would continually linearly increase, after a short period of acceleration, on the roll (θ_1) axis. The control system monitored the motor angles of each twisted string actuator, and halted the experiment when any twisted string actuator motor angle was at or below 2π , as a safety margin to prevent damage to the mechanism. The actuated universal joint angle at the end of the experiment was recorded, as shown in figure ???. By repeating the experiment with increasing values of f_{\min} , it was possible to determine how it affected the maximum roll angle. ?? PythonTeX ?? Attempts to increase the upper bound of this interval resulted in string failure, possibly due to $l_s(\theta_s)$ exceeding l_{\min} but it was not possible to conclusively determine this. Similar experiments for the pitch angle resulted in string failure before any twisted string actuator fully unwound even at $f_{\min} = 3$.

A better solution for increasing actuated universal joint range limits is to reduce the value of r , as is discussed in section ??.

Plots of the setpoint and response for three different trajectories, one on only the roll axis (column-)

Figure 3.17: actuated universal joint roll angle (θ_1) when the smallest twisted string actuator motor angle is equal to 2π , and motor angle of the “opposing” twisted string actuator (the twisted string actuator with the largest motor angle) at the same position.

Table 3.3: Trapezoidal trajectory sequence parameters.

Cycle	Max./Min. Angle [rad]	Max. Velocity [rad s ⁻¹]	Acceleration [rad s ⁻²]
1), one on only the pitch axis (column-	0.2	0.01	0.00125
2), and one on both axes (column-	0.2	0.03	0.02125
3)	0.2	0.05	0.04125
4)	0.2	0.07	0.06125

3.4.3.3 The Effect of AUJ Angular Velocity on AUJ Angle Tracking

To verify the performance of the AUJ at higher angular velocities, the single axis experiments from section ?? were repeated with a trapezoidal “chirp” signal, with the maximum angular velocity increasing by $\omega_0 + (2(n - 1)\omega_0)$ each cycle, where n is the cycle number, and ω_0 is the initial maximum angular velocity, and the angular acceleration increasing by $\alpha_0 + (16(n - 1)\alpha_0)$ each cycle, where α_0 is the initial angular acceleration. A total of four cycles were performed, with the maximum velocity and acceleration values shown in table ?? . Figure ?? shows the results of these experiments, which shows the tracking error increasing marginally as maximum angular velocity and acceleration increases.

3.4.3.4 The Effect of Follower Mass on AUJ Angle Tracking

In order to demonstrate the feasibility of a multiple joint system, additional mass was added to the follower segment and the *Pitch & Roll* trajectory was performed to analyze the effect of the additional mass on the control system. This was performed by the modification of the end of the follower segment to allow multiple standard disc weights, each with a mass of 100 g, to be attached. Four experiments were performed, one with no disc weights attached, and three with one, two, and three disc weights attached, adding approximately 100 g each time. 3D printed custom length spacers were used to maintain an approximately equal center of mass for each of the experiments with disc weights. Figure ?? shows each configuration of the follower segment for the experiments. Table ?? lists the four follower mass configurations that were tested, with follower mass and inertia matrix m and I , and follower center of mass z component ρ_3 . The follower center of mass x and y components were set to 0 for these experiments, as they did not vary significantly between experiments, and had a negligible contribution to the dynamics. These figures are based on estimates from a computer aided design model developed in SolidWorks®, with known weights of 100 g. Any inertial terms less than 1×10^{-5} were rounded down to 0, which resulted in a diagonal inertia matrix. For each experiment, the inverse dynamics parameters m and ρ were updated so the correct setpoint torque would be calculated. Plots include orientation, forces at the top, left

Figure 3.18: Results of the trapezoidal velocity trajectory from table ?? for both actuated universal joint pitch and roll trajectories, including the maximum absolute angle error for each cycle.

Table 3.4: Table of all the follower mass configurations, with the parameters for follower mass m and follower center of mass z offset ρ_3 .

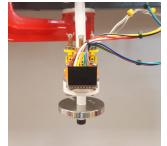
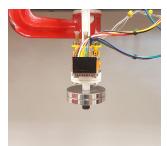
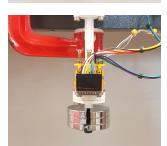
Configuration	m [g]	ρ_3 [mm]	I [kg m ⁻²]	Image
No Mass	67.07	6.65	$\begin{bmatrix} 3.3 \times 10^{-5} & 0 & 0 \\ 0 & 3.1 \times 10^{-5} & 0 \\ 0 & 0 & 1.4 \times 10^{-5} \end{bmatrix}$	
$\pm 100\text{ g}$	166.03	49.16	$\begin{bmatrix} 1.3 \times 10^{-4} & 0 & 0 \\ 0 & 1.3 \times 10^{-4} & 0 \\ 0 & 0 & 4.6 \times 10^{-5} \end{bmatrix}$	
$\pm 200\text{ g}$	215.61	50.01	$\begin{bmatrix} 1.7 \times 10^{-4} & 0 & 0 \\ 0 & 1.7 \times 10^{-4} & 0 \\ 0 & 0 & 7.7 \times 10^{-5} \end{bmatrix}$	
$\pm 300\text{ g}$	315.26	50.48	$\begin{bmatrix} 2.0 \times 10^{-4} & 0 & 0 \\ 0 & 2.0 \times 10^{-4} & 0 \\ 0 & 0 & 1.1 \times 10^{-4} \end{bmatrix}$	

Figure ?? shows the setpoint and response trajectory for each experiment. Initially, k_p was set to the same value as in table ??, however the system was unable to reach a steady state in any configuration other than “No Mass”. Reducing k_p to 1000 in the weighted configurations solved the steady state issue and resulted in an average maximum tracking error of 0° over all configurations with added mass, similar to the result from the initial experiments. However, for the “No Mass” configuration, $k_p = 1000$ resulted in a very poor tracking response, whereas k_p from table ?? resulted in a maximum tracking error of 0° , once again similar to the result from the initial experiments. In future implementations gain scheduling can be employed to select the most optimal k_p for a given follower mass, that allows for the smallest tracking error while being able to reach a steady state.

3.5 Discussion

3.6 Performance Comparison

3.5.1 TSA Comparison with Leadscrew

To compare the performance of a twisted string actuator actuated universal joint against alternatives, we can measure two metrics, the maximum tension force f_{\max} and maxi-

Figure 3.19: actuated universal joint [pitch](#) and [right](#)-roll tracking from figure ?? with increasing follower mass from table ??, and as well as two different values of k_p for the motor positions “No Mass” configuration.

mum stroke velocity \dot{p}_{max} . This corresponds to the equivalent of maximum torque and maximum velocity in a rotary motor, a larger f_{max} would be able to actuate a larger follower mass, and a larger \dot{p}_{max} would be able to rotate the actuated universal joint more quickly. The alternatives chosen for comparison are leadscrews of various rod diameters d_m and pitches λ , and a “direct drive” where the motor is rotating the universal joint directly without any reduction or motion transformation.

3.5.2 Twisted String Actuator

3.5.1.1 Twisted string actuator

For the twisted string actuator metrics, the equations from **wurtz2010**, in particular $h(\theta)$ and $k(\theta)$ which were as used for the state space, which can be used to determine f_{max} and \dot{p}_{max} . By extracting coefficient r_s as an input to make $f(p, r_s)$ and $\dot{p}(\dot{\theta}, p, r_s)$ the performance of different string thicknesses can be compared for a given unwound length l_u and τ_{max} , $\dot{\theta}_{max}$, $\dot{\tau}_{max}$, $\dot{\theta}_{max}$ over the range of the contraction length p . f_{max} is calculated as

$$\begin{aligned} k(\theta) &= l_u - \sqrt{l_u^2 - \theta_s^2 r_s^2} \\ k^{-1}(p) &= \pm \frac{\sqrt{p(2l_u - p)}}{r_s} \\ h^{-1}(\theta) &= \frac{\sqrt{l_u^2 - r_s^2 \theta^2}}{r_s^2 \theta} \\ f(p) &= h^{-1}(k^{-1}(p)) = \pm \frac{\sqrt{(l_u - p)^2}}{r_s \sqrt{p(2l_u - p)}} \\ f_{max} &= f(p)\tau_{max}, \end{aligned} \tag{3.13}$$

and \dot{p}_{max} is calculated as

$$\begin{aligned} \dot{k}(\dot{\theta}, \theta) &= \frac{\dot{\theta} r_s^2 \theta}{\sqrt{l_u^2 - r_s^2 \theta^2}} \\ \dot{p}(\dot{\theta}, p) &= \dot{k}(\dot{\theta}, k^{-1}(p)) = \pm \frac{\dot{\theta} r_s \sqrt{p(2l_u - p)}}{\sqrt{(l_u - p)^2}} \\ \dot{p}_{max} &= \dot{p}(\dot{\theta}_{max}, p). \end{aligned} \tag{3.14}$$

3.5.2 Leadscrew

3.5.1.1 Leadscrew

For the leadscrew metrics, the raising torque calculation shigley2004mechanical can be used as the absolute value of f_{max} , since the twisted string actuator only operates in tension, which can be used to determine the same metrics. The performance of different screw diameters d_m and leads λ can then be compared for a given τ_{max} and coefficient of friction μ . \dot{p}_{max} is then calculated by multiplying λ with $\dot{\theta}_{max}$. The performance of different λ can then be compared for a given $\dot{\theta}_{max}$. f_{max} is calculated as

$$\begin{aligned} |\tau(f)| &= \frac{d_m f (\lambda + \pi d_m \mu)}{2(\pi d_m - \lambda \mu)} \\ |f(\tau)| &= \frac{2\tau(\pi d_m - \lambda \mu)}{d_m (\lambda + \pi d_m \mu)} \\ f_{max} &= |f(\tau_{max})|, \end{aligned} \quad (3.15)$$

and \dot{p}_{max} is calculated as

$$\begin{aligned} \dot{p}(\dot{\theta}) &= \lambda \frac{\dot{\theta}}{2\pi} \\ \dot{p}_{max} &= \dot{p}(\dot{\theta}_{max}). \end{aligned} \quad (3.16)$$

3.5.2 Direct Drive

3.5.1.1 Direct Drive

The direct drive metrics are trivially calculated using the lever force and tangential velocity that would be generated at the endpoint of a linear actuator able to impart equivalent angular velocity and torque on the universal joint. f_{max} is calculated as

$$f_{max} = \frac{\tau_{max}}{\sqrt{l_2^2 + r^2}} \frac{\tau_{max}}{\sqrt{l_2^2 + r^2}}, \quad (3.17)$$

and \dot{p}_{max} is calculated as

$$\dot{p}_{max} = \dot{\theta}_{max} \sqrt{l_2^2 + r^2}. \quad (3.18)$$

3.5.2 Performance Comparison between Leadscrew and Twisted String Actuator

3.5.1.1 Comparison between TSA and Leadscrew

As the values for τ_{\max} and $\dot{\theta}_{\max}$ for the twisted string actuator depend on p , but remain constant for the leadscrew, the performance of the twisted string actuator is going to be better or worse than a given leadscrew depending on the value p . Figure ?? compares the twisted string actuator configuration using the coefficients from table ?? against a number of common leadscrew configurations that are practical for the dimensions of the actuated universal joint. The twisted string actuator outperforms or underperforms different leadscrew configurations depending on p . In simpler terms, the performance of the twisted string actuator is dependent on the contraction length. The maximum linear velocity increases with the contraction length, and the maximum tension force decreases with contraction length, both in a nonlinear fashion.

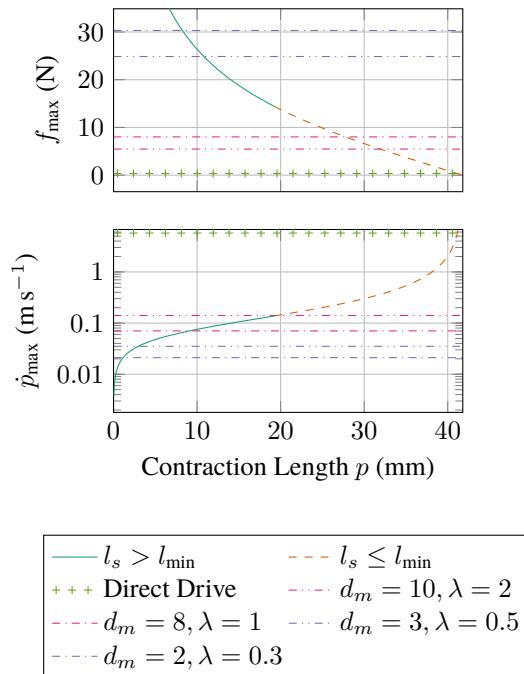


Figure 3.20: Performance comparison of the twisted string actuator configuration using coefficients from table ?? to various leadscrew configurations with different d_m and λ , and the direct drive, where $\mu = 0.1$ for the leadscrews.

3.5.2 Monofilament Torsion Fatigue & String Breakage

twisted string actuator string breakage was common during experiments, this was often due to a loss of control stability during tuning, or operation beyond previously explored motor angle or load limits. However, failure rates were observed to increase with the number of working cycles (winding and unwinding) of each twisted string actuator string, particularly when taken to high motor angles or loads. Nylon monofilament, as was used for the twisted string actuator string, is susceptible to torsion fatigue goswami1976comparative, toney1992bending which reduces tensile strength hennessey2012torsion. This decrease in strength could be a

potential explanation for the increase in failure rate. There was also the route and securing method of the string itself, which used four grub screws to clamp onto the string to hold both ends in place, and also passed close to six potentially sharp edges, as shown in figure ???. These locations could weaken the string, by either an edge biting into it when under high tension, or when being clamped by the grub screws.

The cause of the string break could be determined by the location of the break. If it was located at or near one of the grub screws or biting edges, then it was likely to be caused by those. If it was located on the “active” part of the string that twists, then torsion fatigue is likely to be the culprit. To somewhat mitigate the biting and clamping issue, polytetraflouoroethylene sleeving was added around the string near those locations, and this did result in a significant reduction in string breakages due to edge biting or clamping. This could be improved upon by redesigning the string clamp and capstan to round off sharp edges, and to consider alternatives to clamping such as end knotting or loop wrapping.

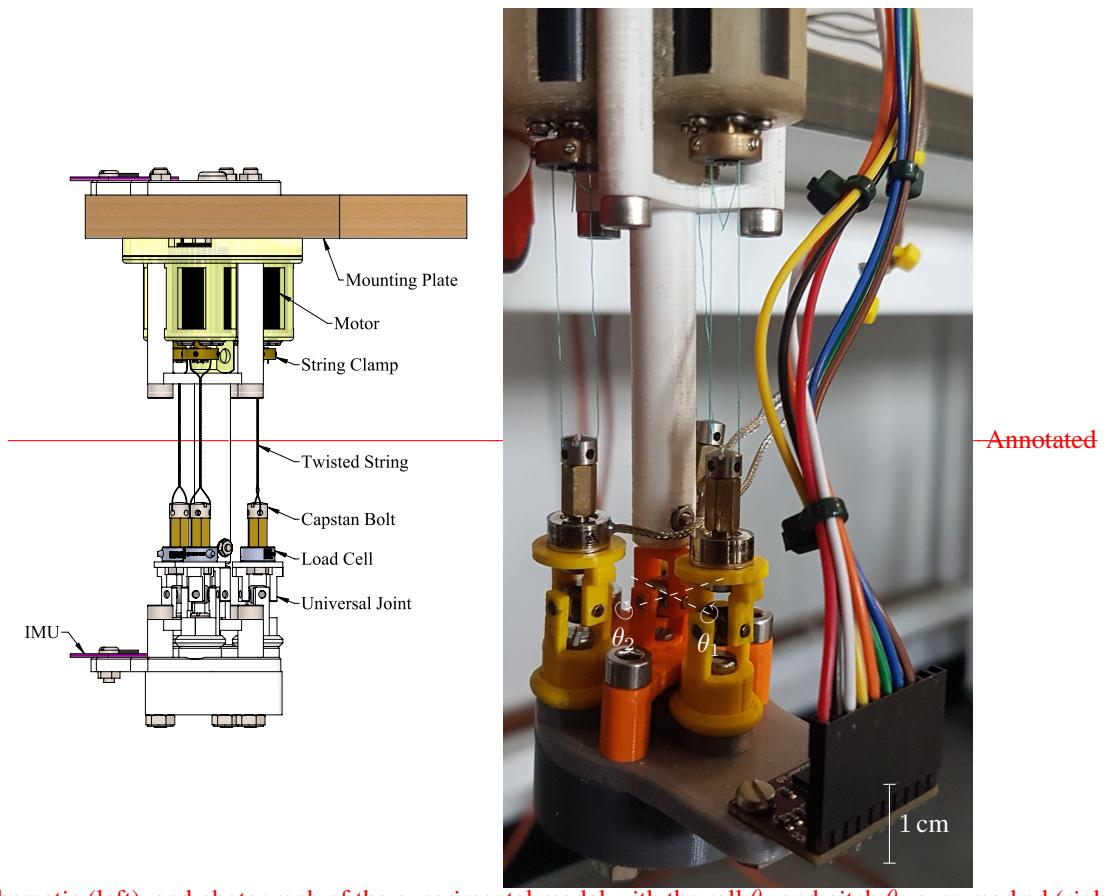
The experiments in **palli2012modeling** used 0.2 mm Dyneema® and Fast Flight® Plus polyethylene polyfilament, and were able to achieve over 1000 working cycles before failure at a load of ≤ 20 N at a motor angle of nearly 300 rad. Use of nylon monofilament was purely for convenience, assembly of polyfilament was difficult as slicing the ends usually resulted in individual fibres splaying, which made it impossible thread through the string clamp. A redesign of the string clamp could make it easier to use polyfilament, which based on the results from **palli2012modeling** could greatly improve string longevity.

3.6 Conclusion

3.5.1 Improving AUJ Angle Range & Enabling Active Transmission Adjustment by Reducing Triad Radius

This research has demonstrated the robust control of the orientation of a universal joint using an antagonistic triad configuration, with a low tracking error (\pm) at low speed (max. $\dot{\gamma}$) in 2 control with a vertical base orientation. It has also compared As can be seen in figure ???, decreasing the triad radius r decreases the stroke range of each twisted string actuator for a given actuated universal joint angle range. This increases the actuated universal joint angle range limits without having to increase f_{min} as was done in section ??.

If r or the actuated universal joint angle range is small enough, this allows for active transmission adjustment, where increasing or decreasing f_{min} shifts the active twisted string actuator stroke “window” to the right or left along the maximum force and maximum velocity curves from figure ??, as shown in figure ???. This would allow the performance of the system to alternative actuation methods. Future developments would include improvements to the orientation sensors, the data from the actuated universal joint to be adjusted during operation, which could be useful in some applications, such as a mobile snake robot which could increase actuated universal joint angular velocity when all segments are on the ground, and sacrifice



schematic (left), and photograph of the experimental model with the roll θ_1 and pitch θ_2 axes marked (right).

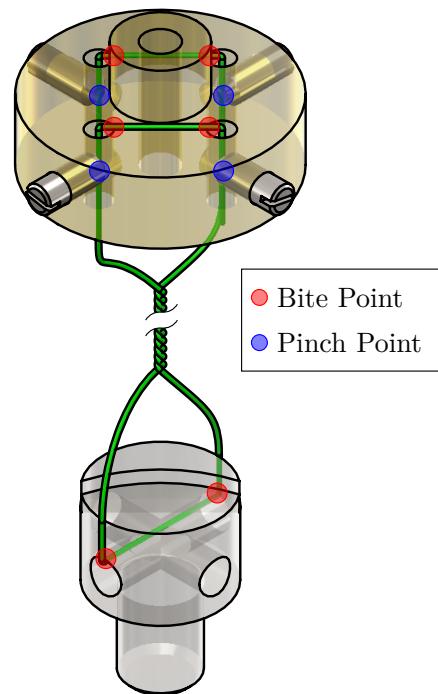


Figure 3.21: Cutaway of a single TSA string assembly, including the capstan and string mount, highlighting the “pinch” points and “bite” points that could cause weaknesses in the string material, which may lead to premature failure.

angular velocity in some segments for increased actuated universal joint torque when they are off the ground and therefore subject to gravitational forces.

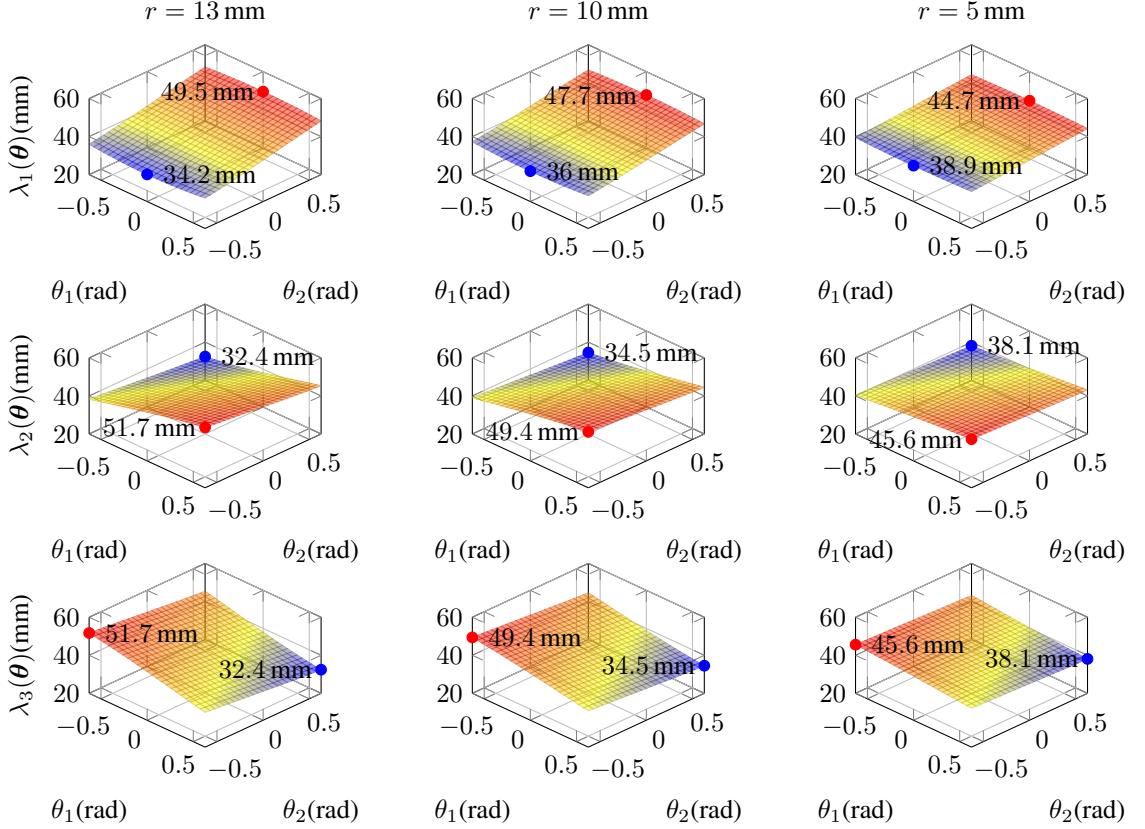


Figure 3.22: Surface plots, minima and maxima for the lengths of $\lambda_{1,2,3}$ at $r = 13 \text{ mm}$, $r = 10 \text{ mm}$ and $r = 5 \text{ mm}$ in the range $[-\frac{\pi}{5}, \frac{\pi}{5}]$. As r decreases, the difference between the minima and maxima also decreases. This means that a lower value of r requires a smaller stroke range of the twisted string actuator for a given actuated universal joint angle range.

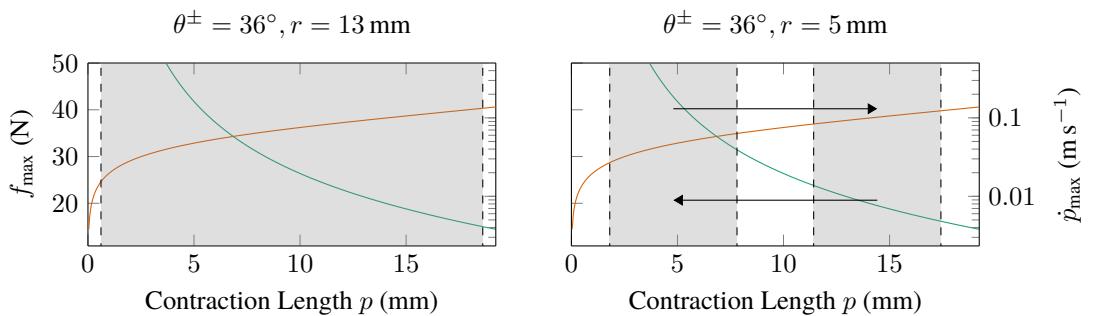


Figure 3.23: By decreasing r , not only can the AUJ angle range be increased by reducing the twisted string actuator stroke range, which is marked in grey, but by adjusting f_{\min} , the transmission ratio of the twisted string actuator can be altered. Reducing f_{\min} increases the maximum twisted string actuator force f_{\max} while reducing the maximum twisted string actuator stroke velocity \dot{p}_{\max} . Conversely, increasing f_{\min} reduces f_{\max} and increases \dot{p}_{\max} . This can be used to actively modify the dynamic properties of the actuated universal joint during operation.

An issue with a small triad radius is the difficulty of accommodating the central shaft with the universal joint. This can be solved by using a wide universal joint with a hollow spider, which allows the three twisted string actuator to pass through the middle of the joint and the central shaft is not required, as shown in figure ??.

Figure 3.24: An AUJ with a central universal joint, and one with a hollow spider. The hollow spider allows r to be decreased as space is no longer required for a central universal joint.

3.5.2 Hardware Improvements

Using the BNO080 as the actuated universal joint orientation sensor proved to be unreliable and of poor resolution, so either a superior will be used or an challenging. The magnetometer was unusable within the indoor experimental environment even after several calibration attempts, so the universal joint angle would not be able to be calculated when the gravity vector is not orthogonal to the universal joint degree of freedom, hence the experiments had the actuated universal joint in a vertical orientation. An alternative method for sensing the universal joint orientation will be investigated actuated universal joint orientation could be used, such as linear variable differential transformers, hall effect sensors or potentiometers. The system was tested in a vertical base orientation in order for the to If the hollow spider universal joint design in section ?? is used, the larger spider pins would be able to measure the joint orientation. With suitable modifications to the inverse dynamics function and a known base orientation, it will be possible to test the controller at non-vertical base orientations. The joint velocity, joint range and follower mass (m) accommodate potentiometers. The 1724TSR motors also had poor performance at low speeds, necessitating the deadband compensator. The use of similar size and torque brushless motors with sinusoidal commutation for stable low speed control could eliminate the need for the compensator, and allow for smoother actuated universal joint motion.

3.5.3 Alternative Control Algorithm

Alternative control algorithms that are less or not dependant on the dynamic properties of the follower segments (mass and inertia) will also be increased considered, in order to test the performance of the system under more strenuous conditions. The controller performance will also be quantified, examining the open and closed loop bandwidth. Eventually, improve the robustness of the control system to external forces.

3.5.4 Multiple Segments

Development of a system comprised of multiple segments will be developed is the eventual goal of the research, to demonstrate its suitability for applications such as mobile snake robots or continuum robots. snake-arm robots. This can use a embedded controller for each

segment, controlled by a primary controller for individual joint control or inverse kinematics, as in figure ??.

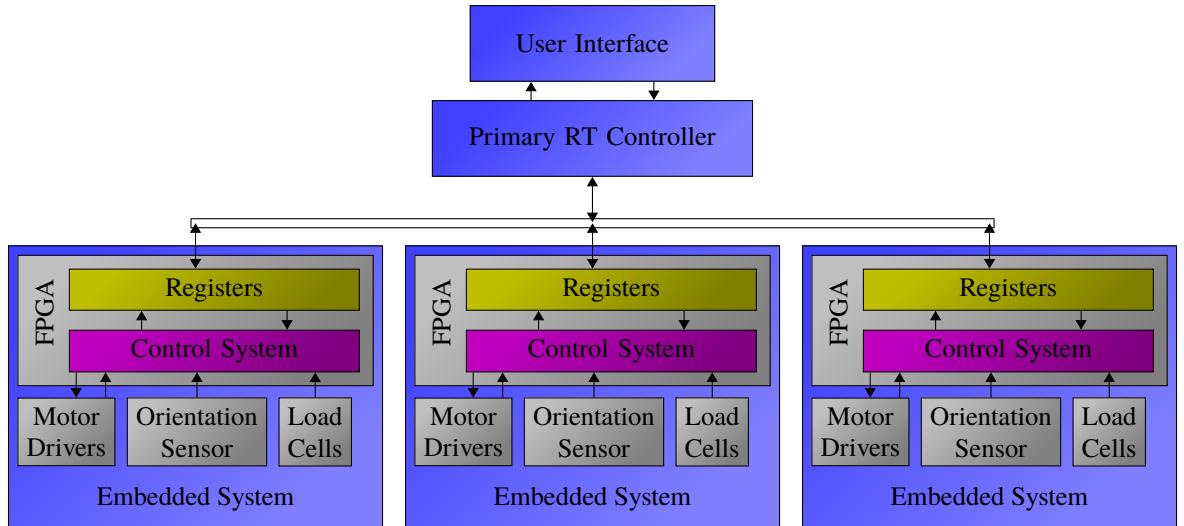


Figure 3.25: Proposed system architecture for a future multi-segment system. Each segment has an embedded FPGA controller programmed with the cascaded control loop in section ???. The FPGA interfaces with the load cells for each TSA, the orientation sensor for the actuated universal joint, and the drivers for the twisted string actuator motors (the FPGA may include some motor control for a more compact design). A primary real-time controller then uses a common control bus to interface with the embedded controllers, reading and writing FPGA registers to issue motion commands and get status updates.

3.6 Conclusion

This research has successfully demonstrated force based control of the orientation of a universal joint using twisted string actuator in an antagonistic triad configuration, at various velocities and follower loads. It has also compared the performance of the system to alternative actuation methods, and found that its non-linear nature makes it difficult to directly compare to similar linear actuators in the same application. Finally, it has examined the limitations of the current design, and has proposed design modifications to improve range and accuracy.

Chapter 4

Creating a Configurable Payload for Static Balance Experiments

In order to generate a diverse set of test data for the static balance experiments in chapter ??, where a robot arm will carry a payload with and without a robot tail to control static balance, a configurable payload was conceived, an object that could be configured to have different masses and center of mass relative to the geometric centre of the object. A series of test points were then generated which have a specific mass and center of mass, and a matching algorithm was used to find the configuration that mostly closely matched these parameters. The experiments in chapter ?? were then run with each of these test points to generate the test data. In this chapter, an abstract mathematical model of a configurable payload was conceived, which the mass and center of mass of a specific configuration of the payload could be derived from. Then three separate sets of “test points” were considered to cover a wide range of the available configuration space within the chosen robot arm’s performance limits, using a combined mass and center of mass vector as a target which finds the closest configuration using a search function, or extrema which maps onto a specific configuration. For target test points, two search methods were examined, one brute force method for small configuration spaces and one for larger configuration spaces where a brute force method would be computationally intractable. Then the chosen implementation was described, including the number of block materials, the number of blocks, the size and design of the container and the chosen search method. The results of this implementation for all test point sets were then tabulated and graphed. In the discussion, the limitations of the mass and center of mass range were considered, and potential designs to expand the range and scope of the experiments in chapter ?? were conceived, specifically improving the range of the center of mass and simulating dynamic loads where the center of mass changes over time.

4.1 Introduction

The payload consists of a matrix of cubes of various materials packed tightly into a three dimensional printed container. The cubes were designed to be changed after each experimental run to alter the mass and center of mass of the payload. A lid on the container prevents the cubes from falling out during the experiment, and the exterior design of the box may accommodate additional features to improve the handling of the payload by the robot arm. An abstract model of this is shown in figure ??.

The test points could then be generated by considering the *configuration space* of the payload design, i.e. how many configurations can be generated given a $n \times n \times n$ matrix of cubes, where each cube can be a number of different materials. The mass and center of mass could then be calculated for each configuration, taking into account the material density and mass and center of mass of the container. “Extrema” test points could then be found simply by finding the configuration with the maximum or minimum mass and center of mass, or combination thereof. Depending on the size of the configuration space, a search method could then be used that accepts an arbitrary mass and center of mass as a target, and finds the nearest configuration to that target for other test point sets.

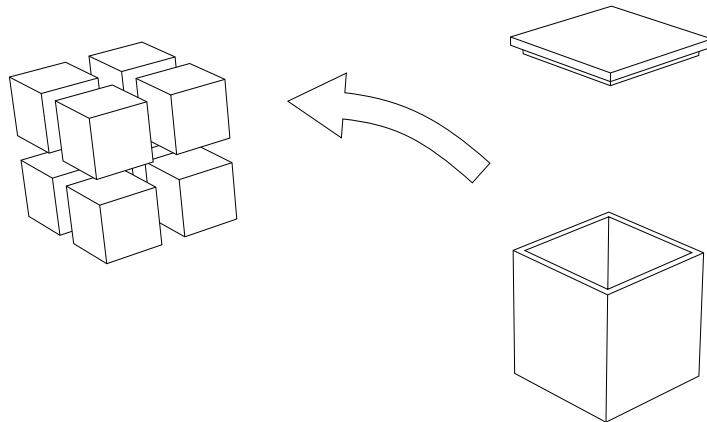


Figure 4.1: Concept drawing of the configurable payload, with swappable cubes that allow the COM to be varied relative to the geometric center.

4.2 Mathematical Design of the Configurable Payload

4.2.1 Mass and COM Configuration Space

Firstly, consider a positive real set of material densities $\mathcal{P} \in \mathbb{R}^+$, each element the density (in kg m^{-3}) of a material to be used \vdash

$$\mathcal{P} = \{\rho_1, \rho_2, \dots, \rho_m \mid \rho_i > 0\}. \quad (4.1)$$

Then consider an $n \times n \times n$ matrix \mathbf{C} , such that each element is an element of \mathcal{P} ,

$$\mathbf{C} \in \mathbb{R}^{n \times n \times n} \mid (c_{ijk}) \in \mathcal{P}, \quad (4.2)$$

where n^3 is the number of elements in the matrix:

$$\mathbf{C} \in \mathbb{R}^{n \times n \times n} \mid (c_{ijk}) \in \mathcal{P}$$

∴

The *configuration space* \mathcal{X} can then be defined as the set of all \mathbf{C} where the sequence of elements are all possible n^3 subsets of \mathcal{P} , which can be written as $[\mathcal{P}]^{n^3}$.

To calculate the mass of \mathbf{C} , take the sum of all the cube densities multiplied by their volume a^3 ,

$$M(\mathbf{C}) = \left(\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n c_{ijk} a^3 \right) + m_c, \quad (4.3)$$

where a is the cube edge length, plus the container mass m_c :

$$M(\mathbf{C}) = \left(\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n c_{ijk} a^3 \right) + m_c$$

∴

To calculate the center of mass, take the sum of each cube mass multiplied by its position relative to the geometric center,

$$R(\mathbf{C}) = \frac{\left(\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n c_{ijk} a^4 \begin{pmatrix} i - (n+1) \\ j - (n+1) \\ k - (n+1) \end{pmatrix} \right)}{M(\mathbf{C})} + \mathbf{r}_c, \quad (4.4)$$

which can be calculated from the cube indexes ijk , plus the container center of mass \mathbf{r}_c if non-zero:

$$R(\mathbf{C}) = \frac{\left(\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n c_{ijk} a^4 \begin{pmatrix} i - (n+1) \\ j - (n+1) \\ k - (n+1) \end{pmatrix} \right)}{M(\mathbf{C})} + \mathbf{r}_c$$

∴

The *solution space* \mathcal{Y} can then be considered as a set of \mathbb{R}^4 vectors containing the mass m_i and center of mass \mathbf{r}_i concatenated as $[m_i \quad \mathbf{r}_i]$ of each element. As such, \mathcal{Y} is a codomain

of \mathcal{X} , such that $H : \mathcal{X} \mapsto \mathcal{Y}$ where the map H is defined by $[M(\mathbf{C}) \ R(\mathbf{C})]$.

4.2.2 Mass Limitation

As \mathcal{X} may have mass values that are greater than what the robot arm can reliably lift, a subset \mathcal{X}_{lim} can be defined as

as

$$\mathcal{X}_{\text{lim}} = \{\boldsymbol{\chi} \in \mathcal{X} \mid M(\boldsymbol{\chi}) \leq m_{\max}\}, \quad (4.5)$$

here where m_{\max} is the maximum mass the robot arm can reliably lift. This can be mapped onto $H : \mathcal{X}_{\text{lim}} \mapsto \mathcal{Y}_{\text{lim}}$ in a similar fashion.

4.2.3 Test Point Sets

Test points can either be derived from subsets of \mathcal{X}_{lim} defined by logical expressions, or the nearest neighbours of \mathcal{Y}_{lim} from a target mass and center of mass concatenated into a vector as in H , found by a *search method*.

4.2.3.1 Extrema Set (\mathcal{E})

The extrema set is designed to test the extrema of the solution space \mathcal{Y}_{lim} in three dimensions, the mass m , and the x and y components of \mathbf{r} . The z component is ignored in this set, changes in this component should not have an effect on the payload stability static balance experiments in chapter ??, but this is verified in the *cube set* in section ???. This allows for test points to be generated at the limits of the capabilities of the payload or robot arm, i.e. the lightest and heaviest possible payloads, and the limits of the payload center of mass on the xy plane, both on the individual axes $\pm x, \pm y$, but also on combined axes (diagonals) $\pm x = \pm y$. The extrema set is defined from a set of logical constraints

$$\mathcal{E} = \left\{ \boldsymbol{\chi} \in \mathcal{X}_{\text{lim}} \mid \begin{array}{l} M(\boldsymbol{\chi}) = \max \{M(\mathcal{X}_{\text{lim}})\} \\ M(\boldsymbol{\chi}) = \min \{M(\mathcal{X}_{\text{lim}})\} \\ R(\boldsymbol{\chi})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \\ R(\boldsymbol{\chi})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \\ R(\boldsymbol{\chi})_y = \max \{R(\mathcal{X}_{\text{lim}})_y\} \\ R(\boldsymbol{\chi})_y = \min \{R(\mathcal{X}_{\text{lim}})_y\} \\ R(\boldsymbol{\chi})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\boldsymbol{\chi})_x = R(\boldsymbol{\chi})_y \\ R(\boldsymbol{\chi})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\boldsymbol{\chi})_x = R(\boldsymbol{\chi})_y \\ R(\boldsymbol{\chi})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\boldsymbol{\chi})_x = -R(\boldsymbol{\chi})_y \\ R(\boldsymbol{\chi})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\boldsymbol{\chi})_x = -R(\boldsymbol{\chi})_y \end{array} \right\}, \quad (4.6)$$

where $R(\mathcal{X})_{x,y}$ are the x and y components of the center of mass. The first two constraints of the set find the maximum and minimum values of the payload mass using $M(\mathcal{X}_{\text{lim}})$, and the next four constraints use the payload center of mass using $R(\mathcal{X}_{\text{lim}})$ to get the maximum and minimum values of the x and y component of the center of mass. Finally, and the last four constraints define the diagonal maximum and minimum values where the center of mass components match $x = y$ or $x = -y$.

$$\mathcal{E} = \left\{ \mathbf{x} \in \mathcal{X}_{\text{lim}} \mid \begin{array}{l} M(\mathbf{x}) = \max \{M(\mathcal{X}_{\text{lim}})\} \\ M(\mathbf{x}) = \min \{M(\mathcal{X}_{\text{lim}})\} \\ R(\mathbf{x})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \\ R(\mathbf{x})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \\ R(\mathbf{x})_y = \max \{R(\mathcal{X}_{\text{lim}})_y\} \\ R(\mathbf{x})_y = \min \{R(\mathcal{X}_{\text{lim}})_y\} \\ R(\mathbf{x})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\mathbf{x})_x = R(\mathbf{x})_y \\ R(\mathbf{x})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\mathbf{x})_x = R(\mathbf{x})_y \\ R(\mathbf{x})_x = \max \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\mathbf{x})_x = -R(\mathbf{x})_y \\ R(\mathbf{x})_x = \min \{R(\mathcal{X}_{\text{lim}})_x\} \wedge R(\mathbf{x})_x = -R(\mathbf{x})_y \end{array} \right\}$$

\mathcal{E}_1 Target Because it was possible for $\max \{M(\mathcal{X}_{\text{lim}})\}$ to have multiple solutions due to the mass constraint, \mathcal{E}_1 was changed to a target $S \left(\begin{bmatrix} m_{\text{max}} & 0 & 0 & * \end{bmatrix} \right)$ where S is the search method function with target structure $\begin{bmatrix} m & r \end{bmatrix}$ that returns an element in \mathcal{X}_{lim} , and $*$ is a “don’t care” notation, to indicate to the search method to not include that component as part of the target. One of the search methods described in section ?? can then be used for S to find the nearest point in configuration space.

4.2.3.2 Cube Set (\mathcal{C})

The cube set is defined by

$$\mathcal{C} = \left\{ \mathbf{x} \in \mathcal{X}_{\text{lim}} \mid \mathbf{x} = S \left(\begin{bmatrix} * & \pm \frac{b}{2} & \pm \frac{b}{2} & \pm \frac{b}{2} \end{bmatrix} \right) \right\}, \quad (4.7)$$

which describes the vertices of a cube of size b centred around the center of mass origin $\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$. This provides a suitable method to generate arbitrary test points within the configuration space, and also tests if the z component of the center of mass has any effect on the payload stability static balance experiment in chapter ??.

$$\mathcal{C} = \left\{ \mathbf{x} \in \mathcal{X}_{\text{lim}} \mid \mathbf{x} = S \left(\begin{bmatrix} * & \pm \frac{b}{2} & \pm \frac{b}{2} & \pm \frac{b}{2} \end{bmatrix} \right) \right\}$$

4.2.3.3 Balanced Set (\mathcal{B})

The balanced set is defined by

$$m_r = \frac{\max\{M(\mathcal{X}_{\text{lim}})\} - \min\{M(\mathcal{X}_{\text{lim}})\}}{q+1} \quad (4.8)$$

$$\mathcal{B} = \left\{ \boldsymbol{x} \in \mathcal{X}_{\text{lim}} \mid \boldsymbol{x} = S \left(\begin{bmatrix} [1..q] & m_r & 0 & 0 & * \end{bmatrix} \right) \right\}$$

which describes q points in \mathcal{X}_{lim} subject to the constraint $R(\boldsymbol{x})_x = 0 \wedge R(\boldsymbol{x})_y = 0 \wedge R(\boldsymbol{x})_z = 0 \wedge R(\boldsymbol{x})_{\text{mass}} = 0$. This can be defined as a “balanced” set as the center of mass x and y components are both zero. The points are evenly spaced between the maximum and minimum mass as defined in section ???. This provides a range of “balanced” test points with zero center of mass but varying mass, to isolate the mass influence of the payload in the payload stability static balance experiment in chapter ??.

$$m_r = \frac{\max\{M(\mathcal{X}_{\text{lim}})\} - \min\{M(\mathcal{X}_{\text{lim}})\}}{q+1}$$

$$\boldsymbol{\zeta} = \begin{bmatrix} m_r & 2m_r & \cdots & qm_r \end{bmatrix}$$

$$\mathcal{B} = \left\{ \boldsymbol{x} \in \mathcal{X}_{\text{lim}} \mid \boldsymbol{x} = S \left(\begin{bmatrix} \boldsymbol{\zeta}_i & 0 & 0 & * \end{bmatrix} \right) \right\}$$

4.2.4 Search Methods for Finding the Nearest Configuration to a Target

When considering a viable search method given a target, the cardinality of \mathcal{X} is important to consider. It is defined as $|\mathcal{X}| = |\mathcal{P}|^{n^3}$ which increases super exponentially with n . For example, when $|\mathcal{P}| = 4$, $n = 2$ results in a value of 65536 and $n = 3$ results in a value of approximately 1.8×10^{16} . It is very clear that when $n > 2$ for non-trivial cardinalities of \mathcal{P} , any kind of brute-force method is not computationally tractable. Therefore, a brute-force nearest neighbour method as in section ?? would be suitable for when $n = 2$, and a heuristic search method such as simulated annealing, as described in section ??, would be suitable for when $n > 2$. Simulated annealing was chosen over alternatives such as a simple gradient descent as it is less likely to get stuck in local minima.

4.2.4.1 Simulated Annealing Search Method

Simulated annealing **kirkpatrick1983optimization** is a modification to gradient descent optimisation **academie1857comptes** that allows the algorithm the chance to “jump out” of local minima early on, even though the approximation becomes temporarily worse. However, as the number of remaining steps decreases, that probability becomes smaller, becoming more and more like gradient descent. First, like any gradient descent algorithm, two things need to be generated, the initial configuration \boldsymbol{C}_0 , which can be random or manually selected, and the function $\mathcal{N}(\boldsymbol{C})$ which creates a set of all the “neighbours” of \boldsymbol{C} . In

this case, this can be defined as

$$\mathcal{N}(\mathbf{C}) = \{x \subset \mathcal{X}_{\text{lim}} \mid \exists! (i, j, k) \mid x_{ijk} \neq c_{ijk}\}, \quad (4.9)$$

the subset of \mathcal{X}_{lim} where the difference between \mathbf{C} and $\mathcal{N}(\mathbf{C})$ is one and only one element:

$$\mathcal{N}(\mathbf{C}) = \{x \subset \mathcal{X}_{\text{lim}} \mid \exists! (i, j, k) \mid x_{ijk} \neq c_{ijk}\}$$

\approx

Then the simulated annealing function can be described as follows:

1. Set \mathbf{C} to the initial configuration \mathbf{C}_0 .
2. For each of the optimisation steps:
 - (a) Set the temperature value t with function $T\left(\frac{s_{\max}}{s}\right)$ which takes into account the number of remaining steps s , where s_{\max} is the total number of steps.
 - (b) Set \mathbf{C}_{new} as a random element from the set of all neighbours of \mathbf{C} as defined by $\mathcal{N}(\mathbf{C})$.
 - (c) Use acceptance probability function $P(E(\mathbf{C}), E(\mathbf{C}_{\text{new}}), t)$ where $E(\mathbf{C})$ is the energy function.
 - (d) Compare that value with a random uniformly distributed real number between 0 and 1. If greater than or equal to, then replace \mathbf{C} with \mathbf{C}_{new} . Otherwise, keep it the same.
 - (e) Repeat with \mathbf{C} until there are no remaining steps.
3. Return the approximated configuration \mathbf{C} .

```

 $C = \mathbf{C}_0$ 
for  $s \leftarrow 1, s_{\max}$  do
   $t = T\left(\frac{s_{\max}}{s}\right)$ 
   $\mathbf{C}_{\text{new}} = \mathcal{N}(\mathbf{C}) \xleftarrow{R} x$ 
  if  $P(E(\mathbf{C}), E(\mathbf{C}_{\text{new}}), t) \geq x \sim U([0, 1])$  then
     $\mathbf{C} = \mathbf{C}_{\text{new}}$ 
  end if
end for
return  $\mathbf{C}$ 

```

Simulated Annealing Energy Function Simulated annealing can also be adapted for multi-objective optimisation **serafini1994simulated**, so it is possible to generate test points that approximate a desired mass and center of mass simultaneously, using a weighting equation such as ??.

Simulated Annealing Cooling Function The function which controls the probability of exiting local minima (known as the *temperature*) is known as the *cooling function*. This function can be any function which monotonically decreases (except in adaptive simulated annealing where it is dependant on the accuracy of the current approximation). Different functions will result in a different cooling profile, generally decreasing quickly in the first few steps, and then slowing down after that. A graph of different cooling functions over 1000 steps is shown in figure ??.

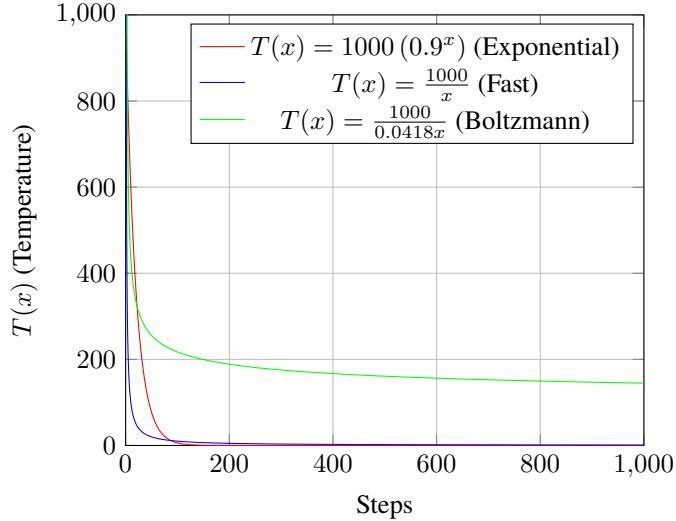


Figure 4.2: Various temperature cooling profiles for simulated annealing, assuming 1000 steps.

4.2.4.2 Nearest Neighbour Search Method

If \mathcal{X}_{lim} is suitably small, then a brute-force method

$$\underline{\underline{NN(t, \mathcal{Y}_{\text{lim}}) = H^{-1}(\min \{\|t - x\|_2 \mid \forall x \in \mathcal{Y}_{\text{lim}}\})}} \quad (4.10)$$

can be used which is guaranteed to find the nearest element to the target within a finite time. This ~~can-be-is~~ done by calculating the 2-norm between the target vector t and all the elements of \mathcal{Y}_{lim} and finding the minimum, then finding the set of \mathcal{X}_{lim} that maps to the minimum \mathcal{Y}_{lim} with H^{-1} . If there are several elements in the domain of \mathcal{X}_{lim} or \mathcal{Y}_{lim} , then one element is chosen at random from the set.

$$\underline{\underline{NN(t, \mathcal{Y}_{\text{lim}}) = H^{-1}(\min \{\|t - x\|_2 \mid \forall x \in \mathcal{Y}_{\text{lim}}\})}}$$

4.3 Implementation of the Configurable Payload

4.3.1 Container Design

The internal width of the container was chosen to be 71 mm to fit comfortably between the robot arm grippers, which combined with a wall width of 5 mm gives an overall width of 81 mm. Therefore, each cube would be $\frac{70}{n}$ mm in size, to allow for fitting clearance. For additional resiliency, the base would be an additional 4 mm, giving the container an overall height of 95 mm with the lid (not including gripper guides). A dimensioned schematic is shown in figure ??.

Two sides of the container were made of a textured pattern designed to enhance grip when picked up by the robot arm. The other two sides had small viewports centred around each cube, for configuration verification during the experiments in chapter ???. The lid of the container had a small notch to allow for it to be levered off in case of a tight fit, and guides to ensure the robot arm grippers would pick up the container in the same location each time, ensuring experiment repeatability. A labelled photograph is shown in figure ??.

The cube container was 3D printed in polylactic acid with an 80% fill density. This gave it a total mass of 0.221 kg which provides the value for m_c . By design, the center of mass of the container would be 0 mm in both the x and y direction, to within a reasonable manufacturing tolerance. Due to the minimal contribution of the container to the total mass, this was a suitable assumption.

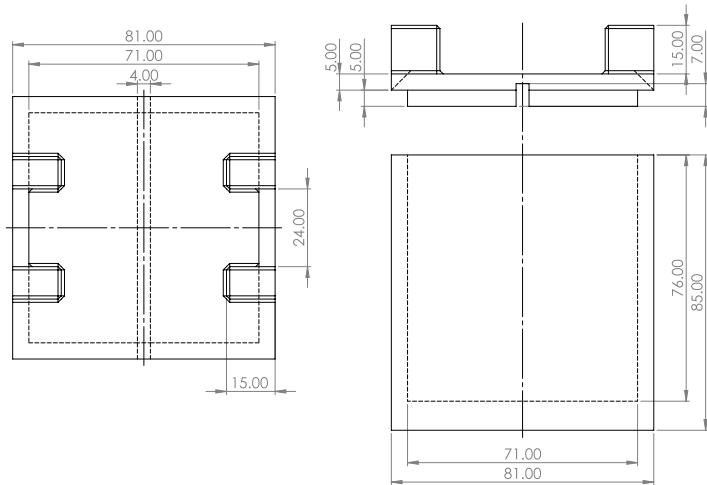


Figure 4.3: Schematic of the container, including body and lid.

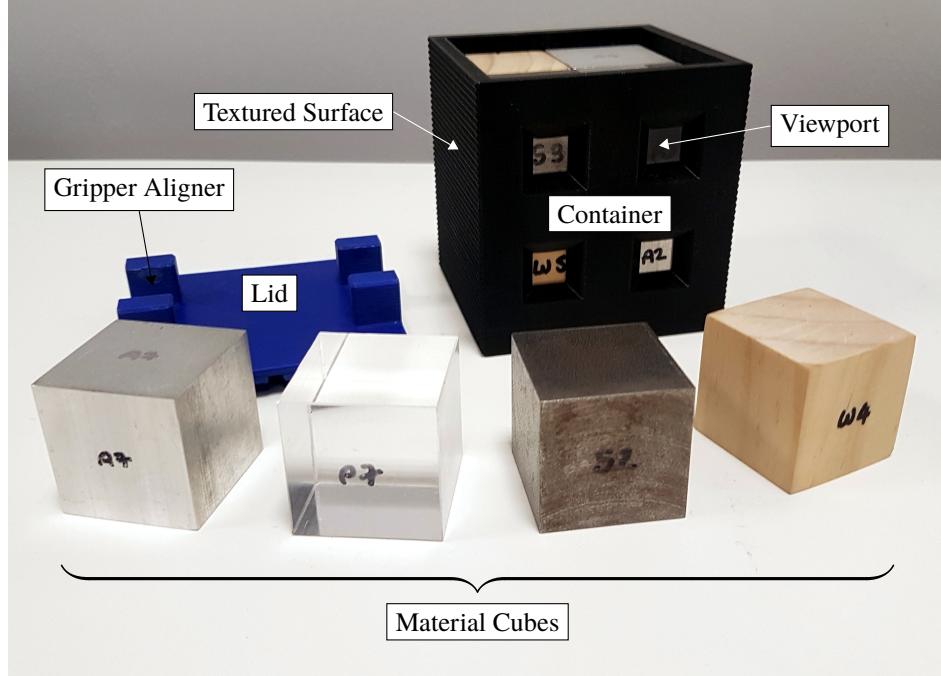


Figure 4.4: Labelled photograph of the final container design, along with the material cubes.

4.3.2 Selected Search Method on the Configuration Space for the Cube Set and Balanced Set

Initially an $n = 3$ configuration was used, using the simulated annealing search method with the acceptance probability function

$$P(\chi, \mathbf{y}, \mathbf{w}, t) = \alpha \underbrace{\prod_{i=1}^m \min \left\{ 1, e^{\frac{w_i(x_i - y_i)}{t}} \right\}}_{\text{Rule P}} + (1 - \alpha) \underbrace{\min \left\{ 1, \max_{i=1, \dots, m} \left\{ 1, e^{\frac{w_i(x_i - y_i)}{t}} \right\} \right\}}_{\text{Rule W}} \quad (4.11)$$

which is Rule M from [serafini1994simulated](#) to allow for a multiobjective search. This is a weighted blend of two other algorithms defined in the paper, Rule P and Rule W with a weighting coefficient $\alpha \in (0, 1) \subset \mathbb{R}$. There is also a weighting vector for each element of the test point $\mathbf{w} \in \mathbb{R}^4 \mid w_i \in (0, 1)$.

$$P(\mathbf{x}, \mathbf{y}, \mathbf{w}, t) = \alpha \underbrace{\prod_{i=1}^m \min \left\{ 1, e^{\frac{w_i(x_i - y_i)}{t}} \right\}}_{\text{Rule P}} + (1 - \alpha) \underbrace{\min \left\{ 1, \max_{i=1, \dots, m} \left\{ 1, e^{\frac{w_i(x_i - y_i)}{t}} \right\} \right\}}_{\text{Rule W}} \quad (4.11)$$

Unfortunately it was difficult to find a stable and consistent result even after 4.3×10^4 steps, as shown in figure ???. Therefore as an alternative to using simulated annealing, the $n = 2$ configuration was used, with larger cubes to compensate. As this only had 65536 configurations, a brute force search would be tractable using the nearest neighbour search algorithm.

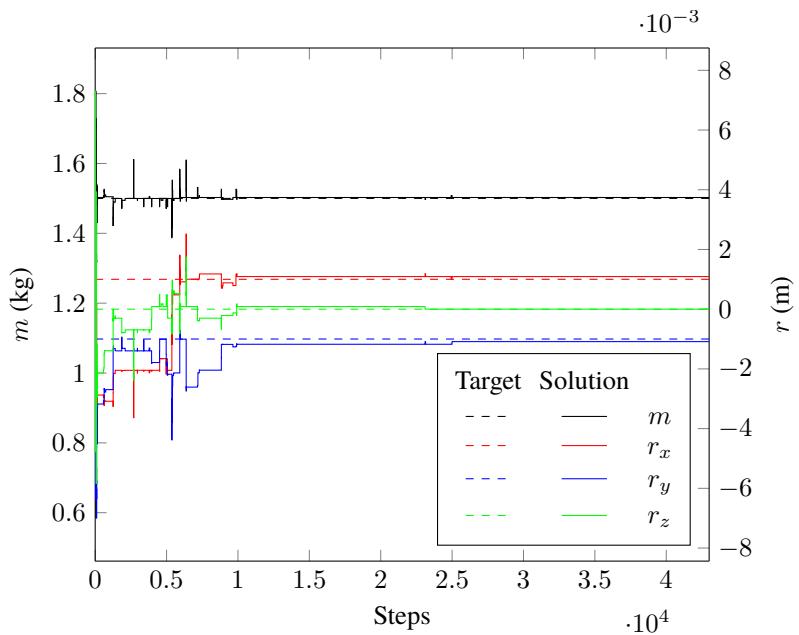


Figure 4.5: Simulated annealing output for the target $[1.5 \quad 0.001 \quad 0.001 \quad 0.001]$ with $\alpha = 0.997$ and even weighting $w_m, w_r = 0.25$ for 4.3×10^4 steps.

4.3.3 Material Selection

In order to produce a reasonably wide and dense configuration space, four materials: *wood*, *plastic*, *aluminium* and *steel*, were chosen. More dense materials, such as nickel and lead, were rejected due to difficulty sourcing stock of the correct size, or potential issues with machining. Initially estimated densities were used in order to test the search methods, but after the cubes were manufactured, it was possible to get an average density based on the measured mass of each cube as seen in figure ??, given a cube size of 35 mm. Table ?? lists the exact kind of material used for each of the four materials, and its average density based on the measured mass. These differ from many stated values available from other sources such as online material databases, likely due to small discrepancies in cube size due to manufacturing tolerances and variability in material composition, particularly for wood since the blocks required sanding in order to fit in the container, and due to the less precise properties of natural materials.

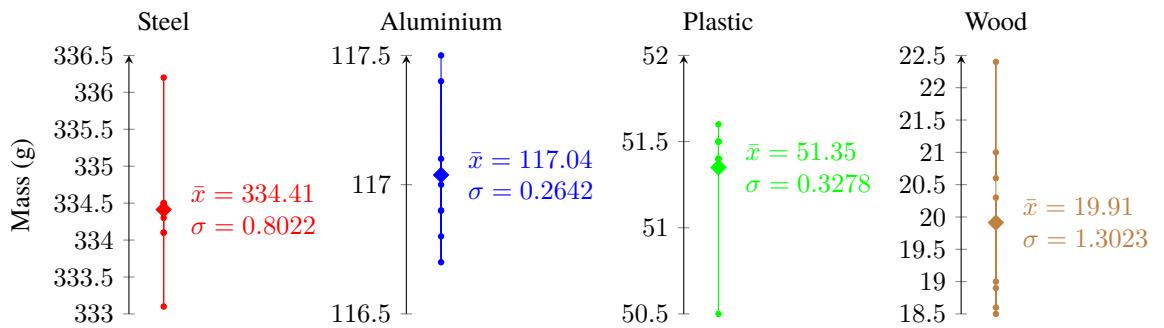


Figure 4.6: Masses for each set of eight 35 mm cubes of the configurable payload for each of the four materials, with the mean mass \bar{x} and standard deviation σ of each set as measured by an electronic scale.

Material	Variant	Density (kg m^{-3})
Wood	Pine	464.37
Plastic	Acrylic	1201.9
Aluminium	6082	2740
Steel	EN3B	7800

Table 4.1: The materials chosen for the cubes and their calculated densities based on the measurements from figure ??.

4.4 Resulting Mass and COM Configurations of the Physical Payload

4.4.1 Solution Space

Figure ?? shows a scatter graph of all the solutions of mass and center of mass of the constructed payload.

Given a $2 \times 2 \times 2$ matrix of cubes with the materials in table ??, there were a total of 65536 configurations, with a total mass range of $[0.38, 2.90]\text{kg}$, and a total center of mass range of $[-13.44, 13.44]\text{mm}$ on all axes. As some configurations mapped to the same point in solution space, there were 62969 unique mass and center of mass vectors.

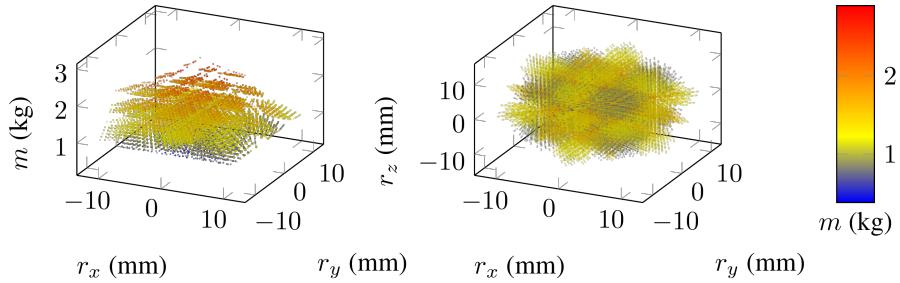


Figure 4.7: Scatter graph of all 62969 unique mass and COM vectors in the configuration space, each point represents a unique configuration.

4.4.2 Test Point Results

Figure ?? shows scatter plots of the mass and center of mass for each test point.

Figure 4.8: Mass and COM coordinates for each test point set.

4.4.2.1 Test Point Mass and center of mass

Tables ?? and ?? show the target positions defined by the notations in section ???. Table ?? also shows the configuration found in the configuration space nearest neighbour search method. The mass limit m_{\max} for the extrema set was set at 1.25 kg, and the cube set length parameter b was set at 7 mm.

Table 4.2: Table of the mass and COM vectors of \mathcal{E} , excluding \mathcal{E}_1 .

Table 4.3: Table of the target and actual mass and COM vectors for \mathcal{C} , \mathcal{B} and \mathcal{E}_1 with the 2-norm error. * notation indicates “don’t care” and is excluded from the search algorithm.

4.4.2.2 Test Point Material Configuration Tables

Tables ??, ?? and ?? show the material configurations for each test point, i.e. the material of each cube in the container, for each point in the test point sets.

Table 4.4: Table of all test point configurations of the *Extrema Set*, showing the mass and COM and material configuration in matrix and 3D form.

Table 4.5: Table of all test point configurations of the *Cube Set*, showing the mass and COM and material configuration in matrix and 3D form.

Table 4.6: Table of all test point configurations of the *Balanced Set*, showing the mass and COM and material configuration in matrix and 3D form.

4.5 Discussion and Conclusion

Due to the discrete nature of a composition of cubes of various densities, exact mass and center of mass configurations were often not possible, so a search method algorithm was required to find the configuration closest to the target. With a more granular ability to vary mass and center of mass, it would be possible to dispense with the need for a search method, as long as a suitably limited level of precision is considered. But this may result in more time required for reconfiguration for each test point, which would make the **payload stability static balance** experiments in chapter ?? more difficult.

The results show it is possible to generate a very large number of configurations (65536) with only 8 cubes of 4 different densities, and increasing to 12 cubes generates such an enormous number of configurations (1.8×10^{16}) it is not tractable to use a brute force search method given a target mass and center of mass. However, the center of mass extrema was small, only

$[-13.44, 13.44]$ mm on all axes, due to the container and material cubes needing to fit inside the robot arm grippers. A different design that allowed the payload to be larger than the gripper width would allow for a significantly greater center of mass range. A potential design that achieves this is discussed in section ??.

4.5.1 Future Work

4.5.1.1 Payload with Improved COM Range

As mentioned in section ??, the center of mass range of the configurable payload is only $[-13.44, 13.44]$ mm on all axes. Similarly to the mass range, the center of mass range could be increased by the use of denser and lighter materials. However, with the current design, it is not possible to increase the center of mass range by a significant amount, even the inclusion of densest and lightest elements, osmium at $22\,590\text{ kg m}^{-3}$ **james1992macmillan**, and hydrogen at $0.089\,88\text{ kg m}^{-3}$ **crc1978crc**, which would be completely impractical or impossible to use, the center of mass range only increased by 2.7 mm. This is due to the small offset distance of the center of mass of each material cube, only 17.5 mm from the geometric centroid of the payload. A design that could increase the maximum offset distance would therefore increase the center of mass range significantly. One proposal would be to use a rod with a mass on one end and a handle for the grippers on the other. The mass material, thickness, and length of the rod could then be varied to adjust the mass and center of mass of the payload. Since the gripper can rotate 360° with the chosen robot arm, any center of mass can be simulated within an annulus shaped solution space around $z = 0$ for a given mass, but could be extruded to an annular cylinder if some mechanism was used to vary the z element of the center of mass (asymmetric masses, rod sliders on the handle, etc.). If two rods either side of the handle are used, the annuli become disks and cylinders respectively, allowing for “balanced” payloads, as can be seen in figure ??.

This is an advantage over the

original design, and eliminates the need for any kind of search algorithm for the center of mass, since it could be found analytically for a given mass using polar coordinates, where the *radius* is the rod length and the *azimuth* is the gripper rotation angle, though it must be noted the mass and maximum rod length will determine the available range of values for the center of mass. A visualization of a potential design is shown in figure ??.

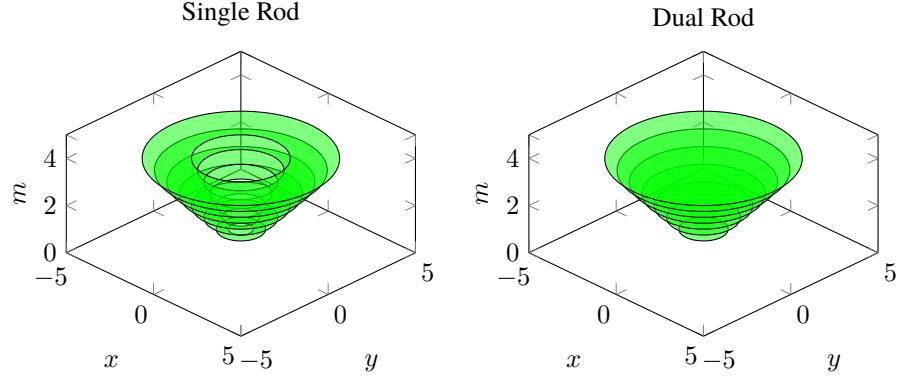


Figure 4.9: Example solution space for the new configurable payload design, where $z = 0$.

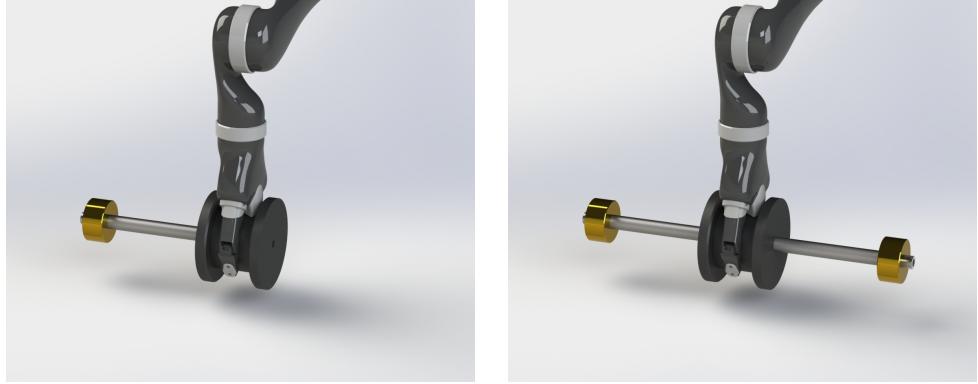


Figure 4.10: Renders of a concept design for the new configurable payload, with a single rod and dual rod configuration.

4.5.1.2 Dynamic COM Implementation: Fluid Filled Container

Not all payloads that a robot may pick up are solid objects, some may have center of mass that shift during transit due to external disturbances. Generally any container partially filled with either loose material or fluid, such as a bottle or sack, could have a so-called “dynamic” center of mass. Therefore, a container partially filled with fluid, with a specific viscosity and mass would be a suitable simulacrum for these kinds of payloads, as shown in figure ???. The robot arm could then shake the container from side to side briefly to simulate a disturbance, creating a standing wave inside the container that would have an effect on the center of mass.

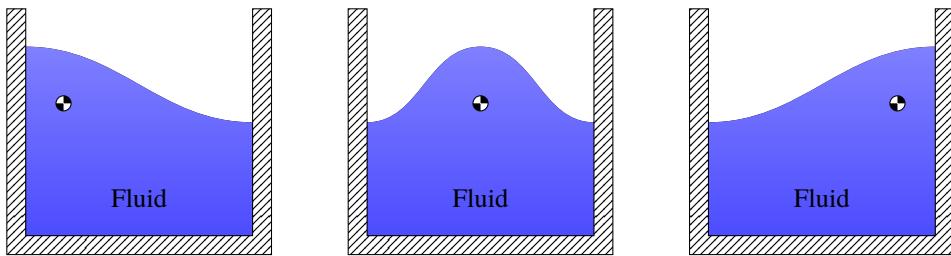


Figure 4.11: Diagram on the COM dynamics of a fluid filled container. If the container is disturbed, a standing wave forms inside the container which causes the COM to oscillate.

4.5.2 Conclusion

This work has outlined the design and implementation of a configurable payload that is suitable for the [payload stability static balance](#) experiments in chapter ???. Using this payload, a range of objects with different mass and center of mass was emulated in order to generate a significant range of experimental data for different simulated payloads.

Chapter 5

Investigating the use of a 2 Degree of Freedom Pendulum Tail for Compensating for Instability when Carrying a Payload

In this chapter, the design of a “static rig”, a stationary simulacrum of a mobile robot with sensors to measure stability, was outlined, made up of a base, a robotic arm, the configurable payload from chapter ??, a 2 degree of freedom “inverted pendulum” tail driven by two brushless motors using a bevel gear arrangement, and four load cells used to calculate the center of pressure. Two cost functions for stability were then derived from the center of pressure data when the payload is being lifted by the arm. An experimental trajectory was designed which replicated the motions of a mobile robot moving an object from one location to another. A proportional derivative controller was then developed to control the robot tail in order to minimise the center of pressure in simulation, and a derived proportional controller was used for the same purpose for experiments using the static rig, for each test point set in chapter ??. Each test point was then executed, with the robot arm picking up, holding, and setting down the payload, with and without the robot tail, in order to compare the stability between the two. A significant improvement in stability of up to 87% was found when the payload was being held with the tail, as opposed to without the tail. The discussion then examines the efficacy of the chosen design and actuation method for the robotic tail, and its limitations and potential improvements, as well as examining the ability of the configurable payload to generate a suitably wide range of test points, given the variation in the motion of the tail for each point.

5.1 Introduction

The primary aim of this research is to quantify the efficacy of a robotic tail for maintaining the stability of a mobile robot when carrying a payload. While the ultimate verification for this question would be to attach a tail to a mobile robot and engage in field trials, carrying objects in a close simulacrum to a real operating environment, a simpler and more flexible approach is to use a static rig, a stationary simulacrum of a mobile robot which can measure the static stability using the center of pressure. By comparing the measurement to a robot's known stability region and initial center of mass, it is possible to determine if a specific mobile robot would remain statically stable without having to topple a mobile robot, and risk equipment damage.

However, demonstrating an improvement in a discrete set of pass/fail stability tests based on specific mobile robots is not necessary to answer the research question. Instead, a continuous approach can be used, which considers stability as a cost function rather than a binary result. This has the advantage of being able to quantify *how* stable the experiment is, and the magnitude of dynamic forces a given robot could withstand. The smaller the cost and the larger the support polygon, the greater the magnitude, depending on the distance to the polygon boundary in a specific direction.

5.2 Measuring the COP using Four Load Cells

The standard method of measuring the center of pressure (see definition in chapter 1.1.1.1) has been to use a *force plate* **robertson2013research**, which uses 4 3-axis load cells in a square or rectangular arrangement as shown in figure ??, which measure the linear force in all 3 axes for a total of 12 degree of freedom.

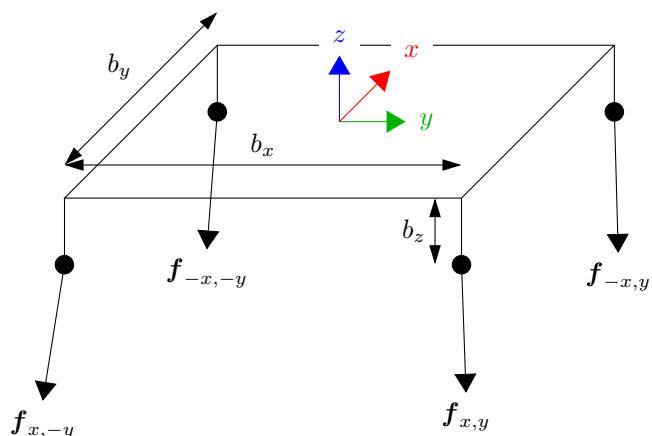


Figure 5.1: Diagram of a typical force plate setup.

To calculate the location The position of the center of pressure ,the following equation can be used:

is then given by

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}_{x,y} & \mathbf{f}_{-x,y} \\ \mathbf{f}_{x,-y} & \mathbf{f}_{-x,-y} \end{bmatrix}$$

$$CP(\mathbf{F}) = \begin{bmatrix} \frac{b_x((f_{11z} + f_{21z}) - (f_{12z} + f_{22z})) + b_z \sum_{i=1}^2 \sum_{j=1}^2 f_{ijx}}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \\ \frac{b_y((f_{11z} + f_{12z}) - (f_{21z} + f_{22z})) + b_z \sum_{i=1}^2 \sum_{j=1}^2 f_{ijy}}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \end{bmatrix}. \quad (5.1)$$

This could be easily adapted to a mobile robot with an arbitrary number of legs or wheels.
~~The following equation can be used for Given n ground contact points with associated load cells:-~~

, equation ?? is redefined as

$$CP(\mathbf{F}) = \frac{\sum_{i=1}^n [b_{x_i} \ b_{y_i}] f_{iz} + b_z [f_{ix} \ f_{iy}]}{\sum_{i=1}^n f_{iz}}, \quad (5.2)$$

~~Where where~~ $[b_{x_i} \ b_{y_i}]$ is the x and y position of the load cell.

It is possible to further simplify the design if only static stability is measured and the load cells Z axis are always in parallel with the gravity vector. Because $\sum_{i=1}^2 \sum_{j=1}^2 f_{ij} = [0 \ 0 \ mg]$ and therefore has no x or y component, 3-axis load cells can be substituted for single axis load cells, which are considerably cheaper and easier to use. By letting $b_z = 0$, ~~the equation equation ?? is further redefined as~~

$$CP(\mathbf{F}) = \begin{bmatrix} \frac{b_x((f_{11z} + f_{21z}) - (f_{12z} + f_{22z}))}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \\ \frac{b_y((f_{11z} + f_{12z}) - (f_{21z} + f_{22z}))}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \end{bmatrix}, \quad (5.3)$$

which now only relies on the scalar z component of the load cells:-

$$CP(\mathbf{F}) = \begin{bmatrix} \frac{b_x((f_{11z} + f_{21z}) - (f_{12z} + f_{22z}))}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \\ \frac{b_y((f_{11z} + f_{12z}) - (f_{21z} + f_{22z}))}{\sum_{i=1}^2 \sum_{j=1}^2 f_{ijz}} \end{bmatrix}$$

~

This is similar to the design of the Wii Fit™ balance board, which is able to measure the center of pressure fairly accurately, as demonstrated in **bartlett2014accuracy**, **leach2014validating** after calibration. The experimental setup used significantly higher quality load cells, each one notably more expensive than an entire balance board, therefore a greater measurement accuracy was expected.

Based on this simplified force plate concept, a “static rig” can be designed that can measure the center of pressure of a robot arm carrying a payload and a robot tail. This can act as a suitable simulacrum for a generic mobile robot, with any number of legs or wheels, that is picking up, carrying, and setting down a payload. The robot arm and tail are positioned along the origin X axis at a_y and $-t_y$ on the Y axis, and a_z and t_z on the Z axis. The tail is a rod attached to a 2 degree of freedom joint, with the orientation controlled by input angles θ . The robot arm is a 4 degree of freedom design with a pair of grippers to pick up a payload, with the input angles q . The following masses and center of mass then need to be considered, the base m_b, \mathbf{r}_b , the arm $m_a, \mathbf{r}_a(q)$, the payload $m_p, \mathbf{r}_p(q)$ the tail drive $m_d, \mathbf{r}_d(q)$ ¹ and the tail, which is calculated from m_t, l_t and θ . A free body diagram is shown in figure ??.

5.2.1 Theoretical COM Derivation

If the mass and center of mass of every component of the system is known, it is possible to calculate the total mass and overall center of mass theoretically. Using the terms from the free body diagram in figure ??, the center of mass can be **found as follows:**

defined as

$$M = m_b + m_d + m_t + m_a + m_p$$

$$R(\theta, q) = \frac{\mathbf{r}_b + \left(\begin{bmatrix} 0 & a_y & a_z \end{bmatrix} + \mathbf{r}_a(q) + \mathbf{r}_p(q) \right) + \left(\begin{bmatrix} 0 & -t_y & t_z \end{bmatrix} + (\mathbf{r}_d + l_t \cos \theta) \right)}{M}. \quad (5.4)$$

If the tail is removed, then the calculations for the tail center of mass and tail drive center of mass are removed from **the equation :**

equation ??, which then becomes

$$\bar{M} = m_b + m_a + m_p$$

$$\bar{R}(q) = \frac{\mathbf{r}_b + \left(\begin{bmatrix} 0 & a_y & a_z \end{bmatrix} + \mathbf{r}_a(q) + \mathbf{r}_p(q) \right)}{\bar{M}}. \quad (5.5)$$

5.3 Robot Arm Trajectory

The experiment robot arm trajectory is designed to replicate the actions of a mobile robot that is moving an object from one place to another. The robot arm grasps the payload and lifts it off the ground plane, then holds it for a predetermined period of time, before placing it back down on the ground plane and releasing it. This sequence is illustrated in figure ??.

This sequence is repeated twice for each test point in the three “sets” designed in chapter

¹separated from the base so it can be removed from the equation when the tail is removed.

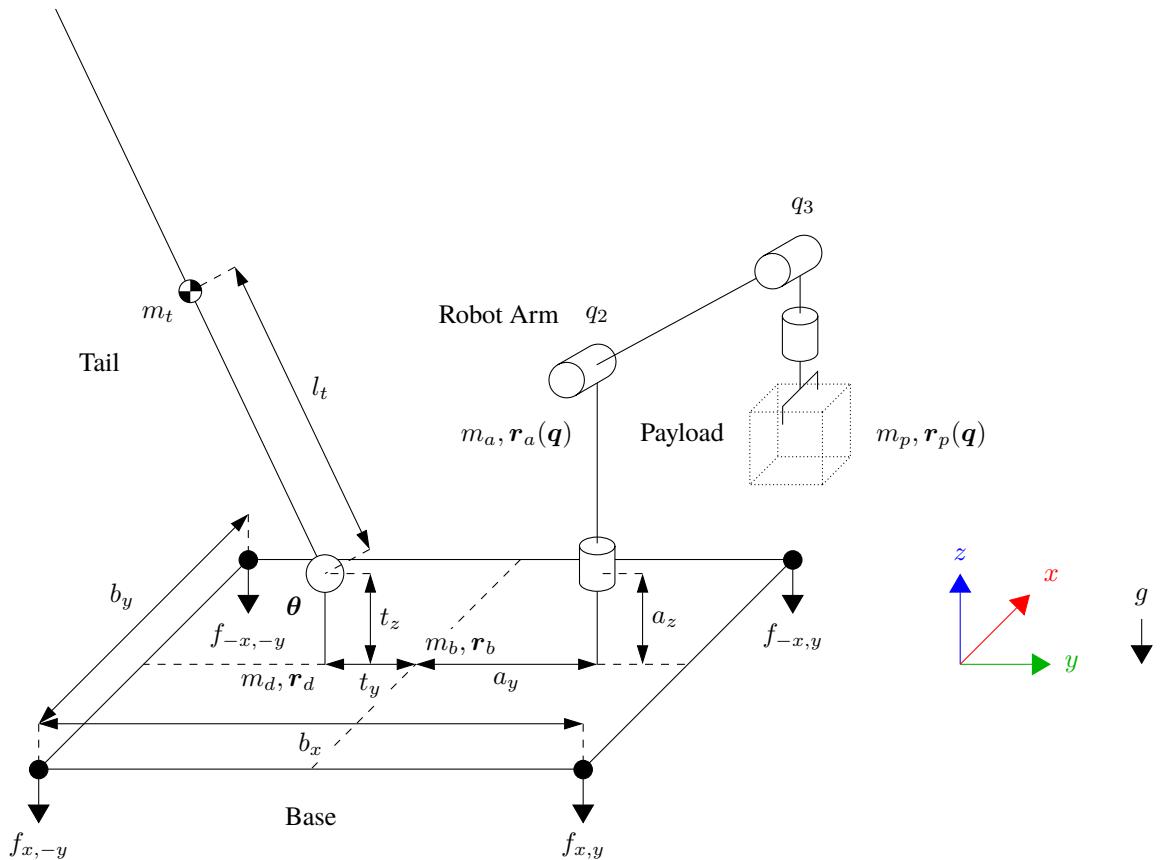


Figure 5.2: Free body diagram of the static test setup.

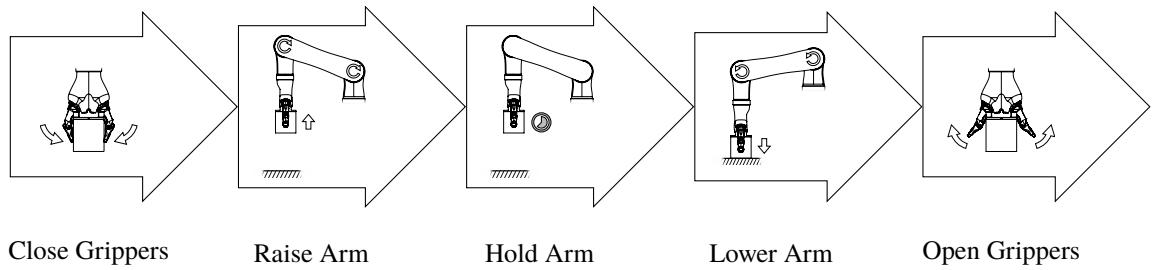


Figure 5.3: Trajectory sequence of the robot arm and payload, The Payload is picked up after *Close Grippers* and set down after *Open Grippers*.

??, once with no tail attached to the rig as a control, and once with the active tail attached to the rig. The area of interest for the research is the “carry” interval of the experiment as shown in figure ??, between the *Raise Arm* and *Lower Arm* stages, specifically when the payload is not in contact with the ground plane.

Figure 5.4: Graph of joint trajectories q_2 and q_3 during an example test sequence (all other joints remain static), and the position of the grippers relative to the fully open position and the maximum closed position to grip the payload. Hatched area indicates the “carry” interval, where the tail will be compensating for the payload mass. Timings vary slightly between each experiment.

Coefficient	Value
b_x, b_y	400 mm
a_y, t_y	170 mm
a_z	132 mm
t_z	116 mm
l_t	439 mm ^a
m_t	1.66 kg ^a
m_a	5 kg ^b
m_p	Variable

^a Composite value calculated from the rod and tip masses and lengths and the position of the tip mass at the end of the rod.

^b Robot arm mass sourced from **micospec** plus the mass of the robot mount.

Table 5.1: Table of the values of the coefficients from figure ?? used in the simulation and experimental apparatus.

5.4 Simscape Multibody™ Simulation of the Experiment

Based on the diagram in figure ?? a Simscape Multibody™ simulation was built. Simscape Multibody™ is a Simulink library that allows the construction of dynamic models from solid bodies, rigid transform and joint blocks. The coefficients in table ?? were used for the simulation construction, which were consistent with the experimental model. As r_b is assumed to be $\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$, the value of m_b could be discounted. A three dimensional preview of this model is shown in figure ?? . These models can then be simulated, and inputs and output from the joints can be used to control the model.

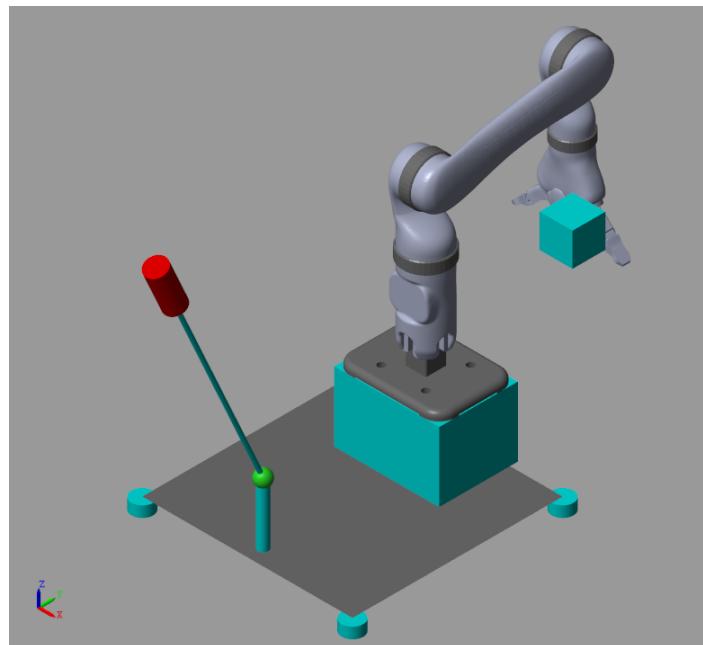


Figure 5.5: 3D preview of the dynamic model of the robot using the Simscape Multibody™ library.

5.4.1 Simulation Construction

5.4.1.1 Simplified Force Plate

For the force plate, a rectangular solid body of dimensions $[b_x \ b_y \ 1 \times 10^{-5}]^2$ ² is connected to the ground (or “world frame” as used in the software) via four weld joints on the lower corners. These joints have no degree of freedom, but act as sensors for the force imparted on them, which is measured and used to calculate the center of pressure using the method described in section ??.

5.4.1.2 Robot Arm

For the robot arm, a unified robotics description format model of the arm to be used in the experiment, the Kinova™ MICO² 4degree of freedom model with a KG-2 gripper end effector, was imported and automatically converted into a chain of revolute joints and solid bodies. The gripper action was not modelled as their low mass had a negligible contribution to the dynamics of the system. To simulate picking up and setting down a payload, a variable mass and center of mass solid was attached to the end effector frame of the arm, initially set to the values for the test point being simulated. A boolean switch would then set the mass to 0 outside a defined interval, ~~as in the equation below:~~

defined as

$$\bar{m}_p(t) = \begin{cases} m_p & \text{if } t_r \leq t \leq t_l \\ 0 & \text{otherwise} \end{cases}, \quad (5.6)$$

where the interval is defined as $[t_r, t_l]$ ³.

5.4.1.3 Robot Tail

For the robot tail, a simple rod and mass model was constructed using cylindrical solids and an actuated universal joint.

5.4.2 Simulation Control System

The control system is a proportional derivative controller that is deigned to minimise the value of the center of pressure relative to the origin at the center of the base, where a mobile robot would typically have the most stability.

The proportional derivative values were determined through the following tuning method:

1. initial values were set to $k_p = 1, k_d = 0$.

²The non-zero z value is required due to simulation limitations. As explained in section ?? this will have no effect on the results.

³In the simulation, the center of mass was also controlled in a similar fashion, again due to simulation limitations.

2. k_p was increased until there was an optimum balance between rise time and % overshoot.
3. k_d was increased until there was no overshoot.

The aim of this tuning method was to maximise the rise time with no overshoot, as this could increase oscillatory behaviour in the robot tail, as shown in section ??.

After this method was complete, the resulting proportional derivative coefficients were $k_p = 180$, $k_d = 60$ with a derivative filter coefficient N of 5 as used in Simulink, which was reduced from 100 in order to filter out simulation noise. A low pass filter is also applied to the center of pressure input to mitigate noise from the simulation solver. An offset of 0.01 was also applied to the y component of the filtered center of pressure input, in order to have the center of pressure at zero when the tail is approximately at zero. For a practical implementation on a mobile robot, mass could be added or moved on the robot body to replicate this offset. A block diagram of the simulation control system is shown in figure ??.

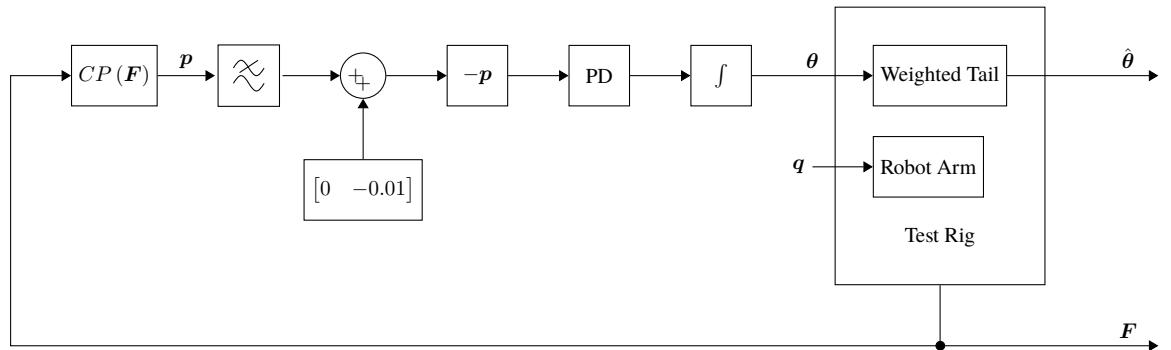


Figure 5.6: Simulation block diagram.

5.5 Experimental Apparatus

As with the simulation, the coefficients from table ?? were used for the experimental apparatus. The control hardware consists of a national instruments myRIO® as a real-time controller connected to a personal computer. This provides an interface for the closed loop control of the robot tail, from the load cell inputs to the motor outputs. The tail orientation is also recorded via the myRIO®. The robot arm is controlled via the personal computer since it has no real-time control driver. A diagram of the control system is shown in figure ??, and an annotated photograph of the apparatus is shown in figure ??.

5.5.1 Simplified Force Plate (Base)

The base consists of two pairs of steel plates orthogonal to each other, separated by load cells and rubber vibration dampers. The lower plates are clamped to a sturdy table at the four positions indicated in figure ??, and a pair of shallow Unistrut® channels are bolted between the upper plates. The plates for the robot arm and tail are then secured to the rig

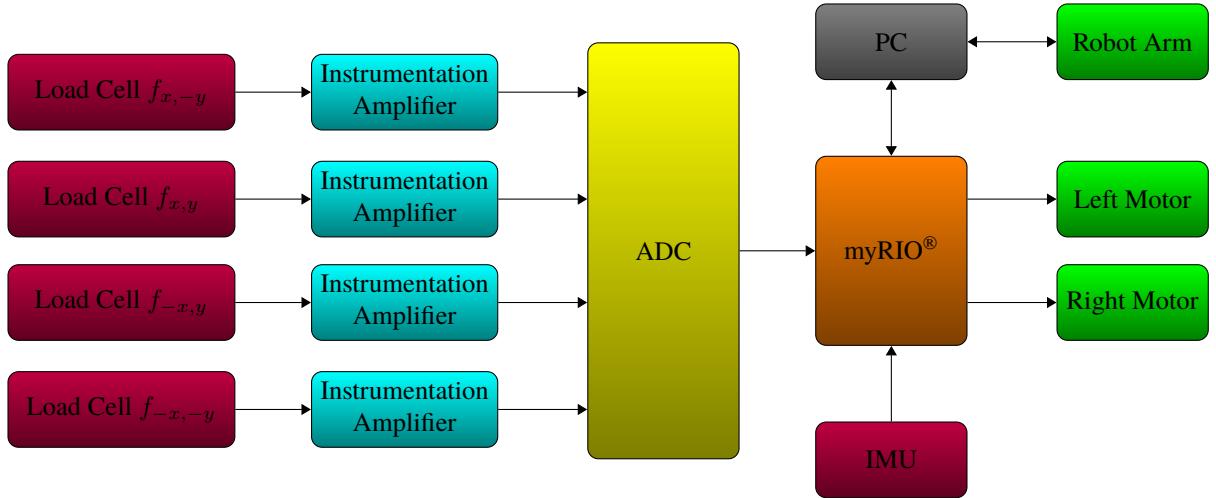


Figure 5.7: Hardware diagram of the experimental apparatus.

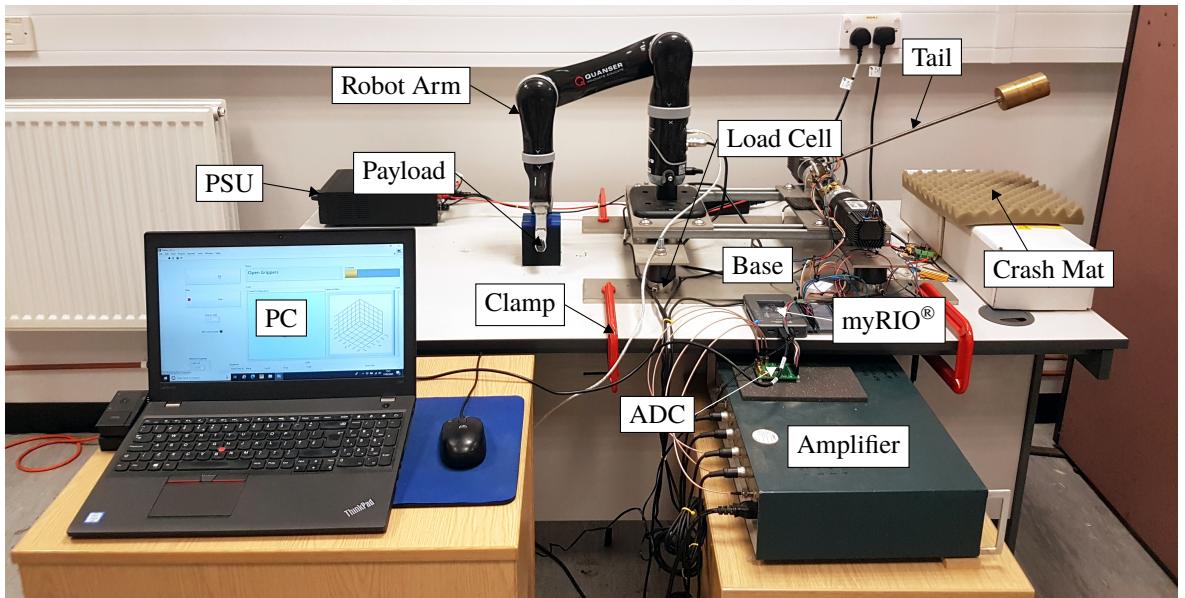


Figure 5.8: Labelled photograph of the experimental apparatus.

with nuts that slot into the channels, allowing them to be shifted along the Y axis so a_y and t_y can be adjusted.

The load cells are Omega™ LCM204 with a range of $\pm 200\text{N}$, connected to Fylde FE-359-TA instrumentation amplifiers that fed the amplified signal to an Analog Devices AD7606 analog to digital converter, which was connected to the national instruments myRIO®. A separate analog to digital converter was chosen due the limited number of differential analog inputs on the myRIO®, with only two on the myRIO® **myrio**. The chosen analog to digital converter also had a greater resolution of 16 bits **ad7606** than the built-in myRIO® analog to digital converter of 12 bits **myrio**.

5.5.2 Robot Arm

The robot arm is a Kinova™ MICO² 4degree of freedom model with a KG-2 gripper end effector. It can carry a maximum payload of 2 kg when fully extended, though in exper-

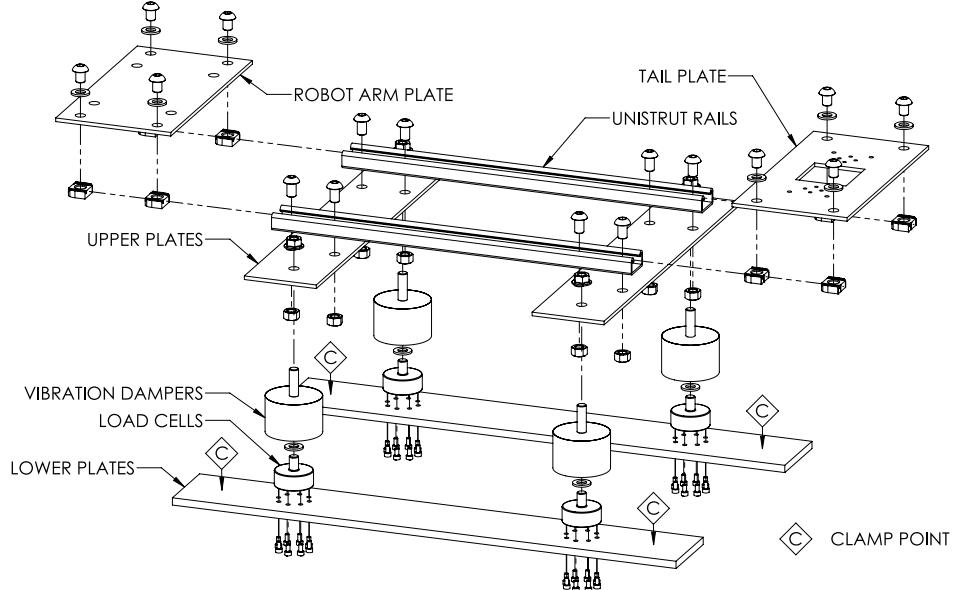


Figure 5.9: Exploded schematic of the base of the static rig, with important components labelled and clamping points marked.

imental conditions for the configurable payload designed in chapter ?? the friction force between the grippers and textured surface of the container was insufficient for loads much lower than this limit. It is attached to the base using a mount which is bolted to the robot arm plate.

5.5.3 Robot Tail

While the actuator designed in chapter 3 proved that it could be a successful actuation system for the robot tail, it was not considered mature enough to be applied to this experiment. Instead, a simpler design was used, based on the common “inverted pendulum” designs used in many of the publications found in chapter 2, with a 2 degree of freedom joint attached to a shaft with a “tip mass” on the end. A crash mat was added at the rear of the setup, in order to prevent a falling tail from damaging the table when power was cut to the actuators. An exploded diagram of the robot tail is shown in figure ??.

5.5.3.1 Bevel Gear Mechanism for Robot Tail Actuation

The mechanical design of the tail joint uses the design from **patel2015conical**, which uses a 3 way bevelled gear system to create a 2 degree of freedom joint, which allows both motors to be mounted to the base, increasing the available tail torque. This design works as shown in figure ??, where rotating both motors in the same direction rotates the tail joint around the axis of the *driving* gears connected to the motor shafts, and rotating the motors in opposite directions rotates the tail in the axis of the *driven* gear attached to the tail.

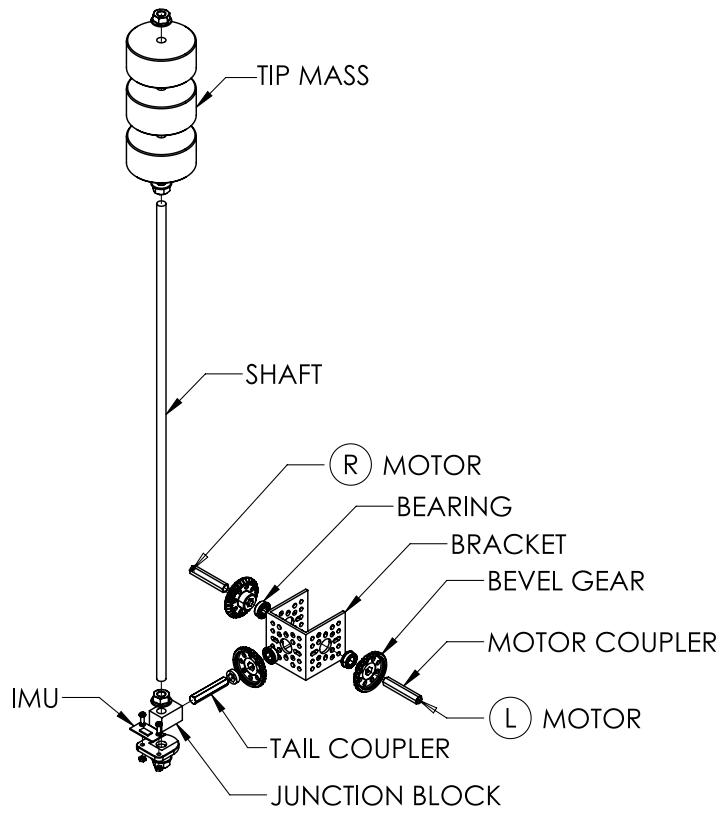


Figure 5.10: Exploded schematic of the robot tail, with important components labelled and attachments for motor shafts marked.

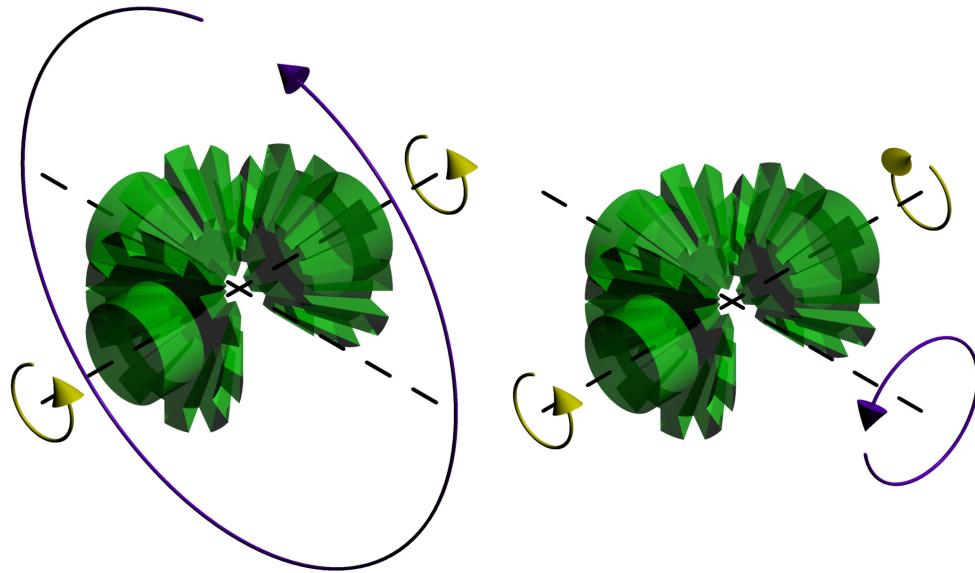


Figure 5.11: Visualisation of the bevel gear system. When both driving gears turn in the same direction, the driven gear rotates around the driving gears axis. When the driving gears turn in opposite directions, it rotates around its own axis.

In order to translate motor rotations to tail motion, [the following equation](#)

$$\dot{\phi} = \begin{bmatrix} 1 & -1 \\ -1 & -1 \end{bmatrix} \dot{\theta} \quad (5.7)$$

can be used [as defined in patel2015conical](#), where $\phi = [\phi_1 \ \phi_2]^T$ are the motor angles:

$$\dot{\phi} = \begin{bmatrix} 1 & -1 \\ -1 & -1 \end{bmatrix} \dot{\theta}$$

⋮

5.5.3.2 Motors for Robot Tail Actuation

The motors chosen for the tail mechanism are Crouzet DCmind brushless gearmotors, with a maximum output speed of 208 min^{-1} at 12 V(DC) and a maximum torque of 6.8 N m **crouzet**. These consisted of a brushless motor and a 19:1 ratio gearbox. 6.8 N m is comfortably in excess of half the maximum torque required to move the tail, which can be calculated as $\tau_{\max} = l_t m_t g$ which corresponds to a horizontal tail orientation. Using the coefficients in table ??, this is 7.1 N m . Lower cost brushed motors (goBILDA® 5204 Series Yellow Jacket Planetary Gear Motor) were initially used, but were unable to move reliably at low velocity, making smooth control of the tail difficult. Using brushless motors allowed for better control at low velocities, using field oriented control **john2011space**, **godbbole2006field** implemented on the motor's internal controller hardware. The motors were attached to shafts A and B as seen in figure ?? with couplings that would transfer the rotation of the motor shaft to the shaft on the bevel gear assembly on the robot tail.

The motors were controlled from the myRIO® using analog and digital outputs which could be used as inputs to a variety of preprogrammed internal control systems that could be selected by the user before operation. The control program chosen, called *V202*, used a “moving target” velocity control system that controlled the motor using a position proportional integral derivative controller, with the setpoint being changed at the rate specified by the velocity input. The proportional integral derivative parameters were set at $k_{p_m} = 1.75$, $k_{i_m} = 0.006$, $k_{d_m} = 3$, except for the balanced set where $k_{d_m} = 2$, due to unknown instability issues caused by a gap in experiments, discussed in section ??.

5.5.3.3 Inertial Measurement Unit

The tail also includes a Bosch Sensortec BNO080 inertial measurement unit as used in chapter 3, mounted to the base of the tail to sense the angular orientation of the tail. This is used to limit the tail range and for data acquisition purposes.

5.5.4 Control System

While the experimental control system was very similar the control system used for the simulation in section ??, there were some differences.

The initial proportional integral derivative were set to the same values as in the simulation, $k_p = 1$, $k_i = 0$, $k_d = 0$. However, due to the fact that tuning was occurring on a physical system, where overshoot or oscillation could cause permanent damage to equipment,

k_p was increased much more slowly and carefully, so as to keep the response smooth and stable, overshoot was to be minimised or avoided entirely during tuning. Using the proportional term alone, it was possible to achieve an acceptable rise time within the carry interval, with the y component of the center of pressure reduced to an absolute value of less than 2.887 mm (see section ??). With the potential of causing permanent damage to the system from overshoot or oscillation taken into consideration, adding a k_d term would be an unnecessary risk, simply guessing what value would be needed to avoid any overshoot for a given value of k_p may result in damage to the system. Therefore only a proportional controller was used for the experimental control system, instead of a proportional derivative controller, with $k_p = 40$.

The control loop was set at a frequency of 1 kHz. ~~There, and there~~ was also a limit function $\lim(\dot{\theta}, \hat{\theta})$ to replicate the joint limits used in the simulation, using the current tail angle $\hat{\theta}$ as an input. This function multiplies each element of the input with a linearly interpolated look up table based on the current tail angle:

$$\lim(\dot{\theta}, \hat{\theta}) = \mathbf{x} \mid x_i = \dot{\theta}_i \times \begin{cases} 0 & \text{if } \hat{\theta}_i \geq \theta_i^+ \vee \hat{\theta}_i \leq \theta_i^- \\ \frac{\theta_i^+ - \hat{\theta}_i}{d} & \text{if } \theta_i^+ > \hat{\theta}_i > \theta_i^+ - d \\ \frac{\hat{\theta}_i - \theta_i^-}{d} & \text{if } \theta_i^- + d > \hat{\theta}_i > \theta_i^- \\ 1 & \text{otherwise} \end{cases} \quad (5.8)$$

where the joint limits are defined as θ^+ and θ^- , and d is a “falloff” constant that allows for a smooth slowdown of the tail near the limits. ~~This function multiplies each element of the input with a linearly interpolated look up table based on the current tail angle, and was used to replicate the joint limits used in the simulation, using the current tail angle $\hat{\theta}$ as an input.~~

Following that, there was a function to convert the joint angles to motor angles as defined in section ??, a saturation function to limit motor velocity for safety reasons set at 0.2 rad s⁻¹ for the experiments, a conversion from rad s⁻¹ to min⁻¹ (RPM), and a gain to compensate for the gear reduction. The center of pressure y component offset was also changed to -0.03 , as r_b had a non-zero y component, but as it would have been difficult to measure the value of that precisely, changing the offset using a trial and error approach until the robot tail angle was approximately zero was deemed sufficient for the experiment.

A block diagram of the system is shown in figure ??.

5.6 Results

For each test point, 20 experiments were conducted, 10 *control* experiments without the tail attached, and 10 *tail* experiments with the feedback controlled tail. The mean values for the 10 experiments were then taken as the measurements, with the minimum and maxi-

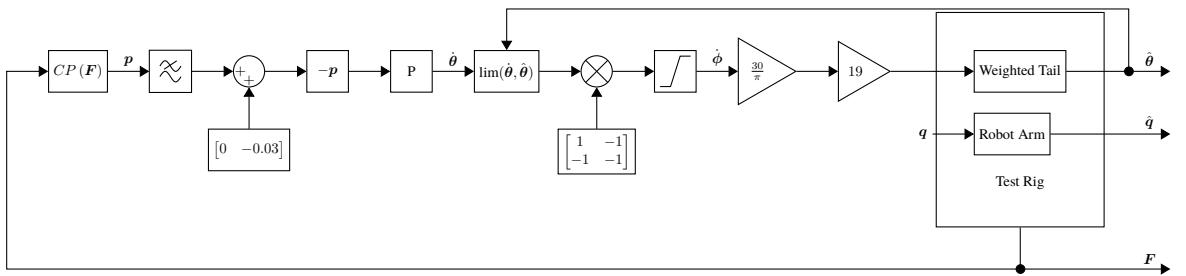


Figure 5.12: Block diagram of the experimental control system.

mum values considered as the measurement error. Data was collected approximately every 100 ms by the PC, with a millisecond timestamp included for accurate analysis.

In order to provide a fair comparison between the control and tail experiments, a perfectly balanced initial condition, $CP(\mathbf{F}) = \begin{bmatrix} 0 & 0 \end{bmatrix}$, is assumed for the control experiments. This is because it would be possible to balance the robot in a stationary state without using the tail, by adding mass at specific locations on the body. This was done by offsetting the center of pressure data by the value of the initial condition. An example of the output from one of the experiments is shown in figure ??.

$$1.291 [0.000 \quad -0.000 \quad -0.008]$$

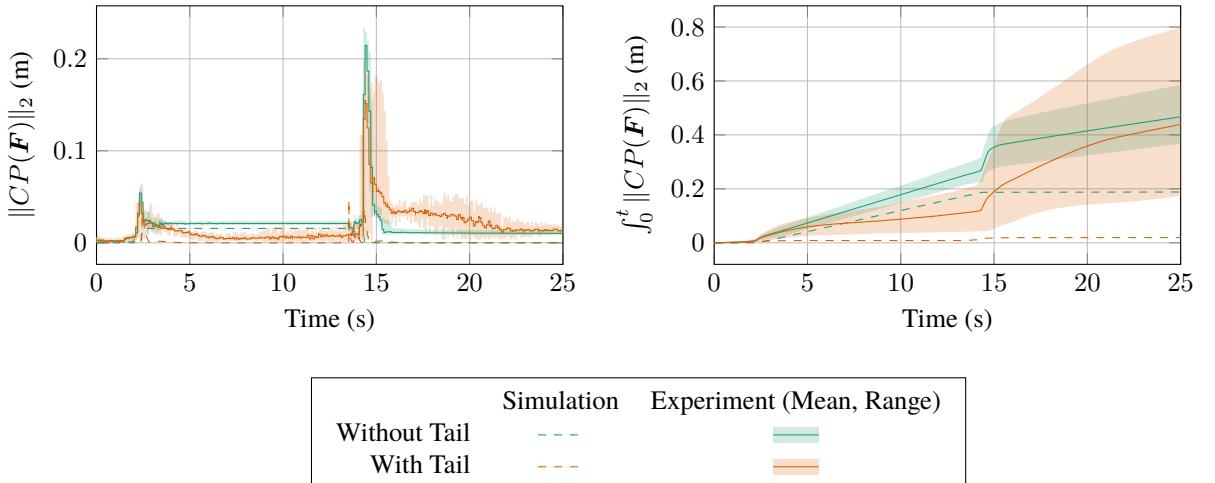


Figure 5.13: Example output of the simulation and experimental results from \mathcal{E}_1 , showing the COP 2-norm $\|CP(\mathbf{F})\|_2$, and the cumulative integrated COP 2-norm $\sum_0^t \|CP(\mathbf{F})\|_2 \int_0^t \|CP(\mathbf{F})\|_2$ using the trapezoidal rule. Note the large spikes seen on the experimental results are the result of the robot arm pushing the payload into the ground to acquire a secure grip and keep the payload in the same position for each experiment, which creates an opposing force on the load cells but is not important for the experiment. Results for each test point can be found in appendix ??.

In order to compare the performance with and without the robot tail, the stability measurement given by the center of pressure can be quantified into a “cost function” that gives a single scalar or two-vector (for both components of the center of pressure) value for each test point. These values can then be used to score the performance of the experiment, and provide a means to compare the control and tail experiments.

Since the area of interest for the experiments is the carry interval as defined in section ??,

these cost functions were only applied to a similar interval, $t \in [5, 13]$, with 5 and 13 chosen as the lower and upper bound respectively as they were always well within the carry interval for each experiment. This would also ensure a consistent measurement interval, which was not dependant on the exact timing of test sequence stage transitions.

5.6.1 Minimum COP Deviation

This cost function measures the smallest (or minimum) value of the center of pressure. This effectively measures the most stable state of the system during the experiment. This can be defined by [the following equation](#),

$$\min \{CP(\mathbf{F}(t)) \mid t \in T, 5 < t < 13\} \quad (5.9)$$

which takes the minimum center of pressure sample from the experiment time series T within the defined carry interval defined in section ??.

$$\min \{CP(\mathbf{F}(t)) \mid t \in T, 5 < t < 13\}$$

^

Figure ?? shows the mean minimum COP deviation position for the control and tail experiments. As can be clearly seen, the y component of the center of pressure is reduced to an absolute value of less than 2.701 mm in every tail experiment. Though the x component does increase, this is by at least an order of magnitude less than the reduction in the y component. This is due to control instabilities caused by the dynamic properties of the motor and tail as explained in section ??.

From the data in figure ??, a clear reduction in the y component of the mean minimum center of pressure can be seen, a maximum of 88% for test point \mathcal{E}_5 .

The confidence in the change in value can be measured using a paired t-test, which measures the probability of two hypothesis, that a set of value pairs *is* zero and a set of value pairs *is not* zero. It does this by comparing the mean difference between each pair of values. The result from all sets shows a statistically significant mean difference between the y component of the cost function between the control and tail experiments, with $p = 2.057 \times 10^{-15}$. The mean reduction is not statistically useful, since it would be expected to very greatly depending on the mass and center of mass of the payload.

5.6.2 Mean [Cumulative Integrated](#) COP Deviation 2-Norm

This cost function measures the [cumulative integrated](#) deviation of the euclidean norm of the center of pressure. Whereas the previous cost function measured the most stable state the experiment achieved, this measures how well the stability was maintained over the entire interval. This can be defined by [the following equation](#):

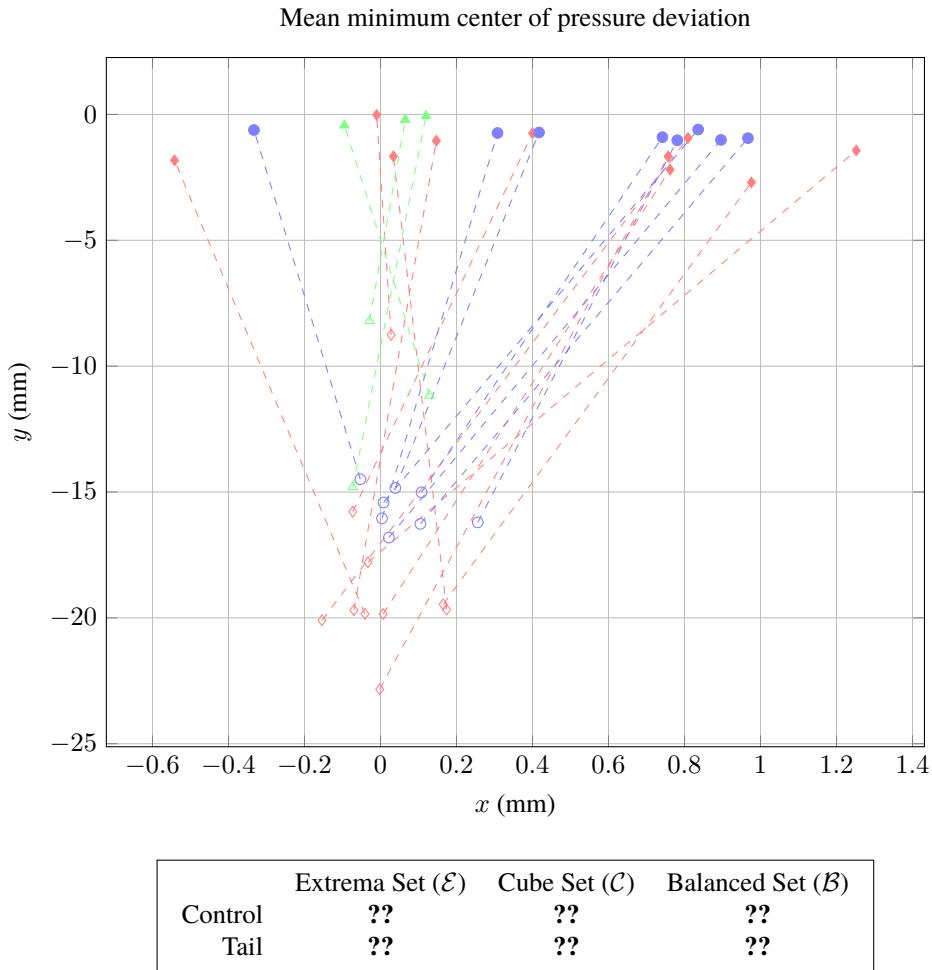


Figure 5.14: Graph of the mean minimum COP during the *carry* interval of the experiment, with and without the robot tail.

$$\sum_{\substack{t \in \{T, 5 < t < 13\} \\ \sim}} \int \|CP(\mathbf{F}(t))\|_2. \quad (5.10)$$

For discrete integration of non-uniform data, this function uses the trapezoidal rule.

Figure ?? shows the mean **eumulative-integrated** COP deviation 2-norm for the control and tail experiments. Every tail experiment achieves a significant reduction of this cost function.

From the data in figure ??, another clear reduction in the mean **eumulative-integrated** center of pressure 2-norm can be seen, a maximum of 76% for test point \mathcal{E}_5 .

Using the same paired t-test used in the previous cost function, the reduction is again shown to be statistically significant, with $p = 1.100 \times 10^{-14}$.

Integrated 2-Norm Deviation in t interval [5, 13]

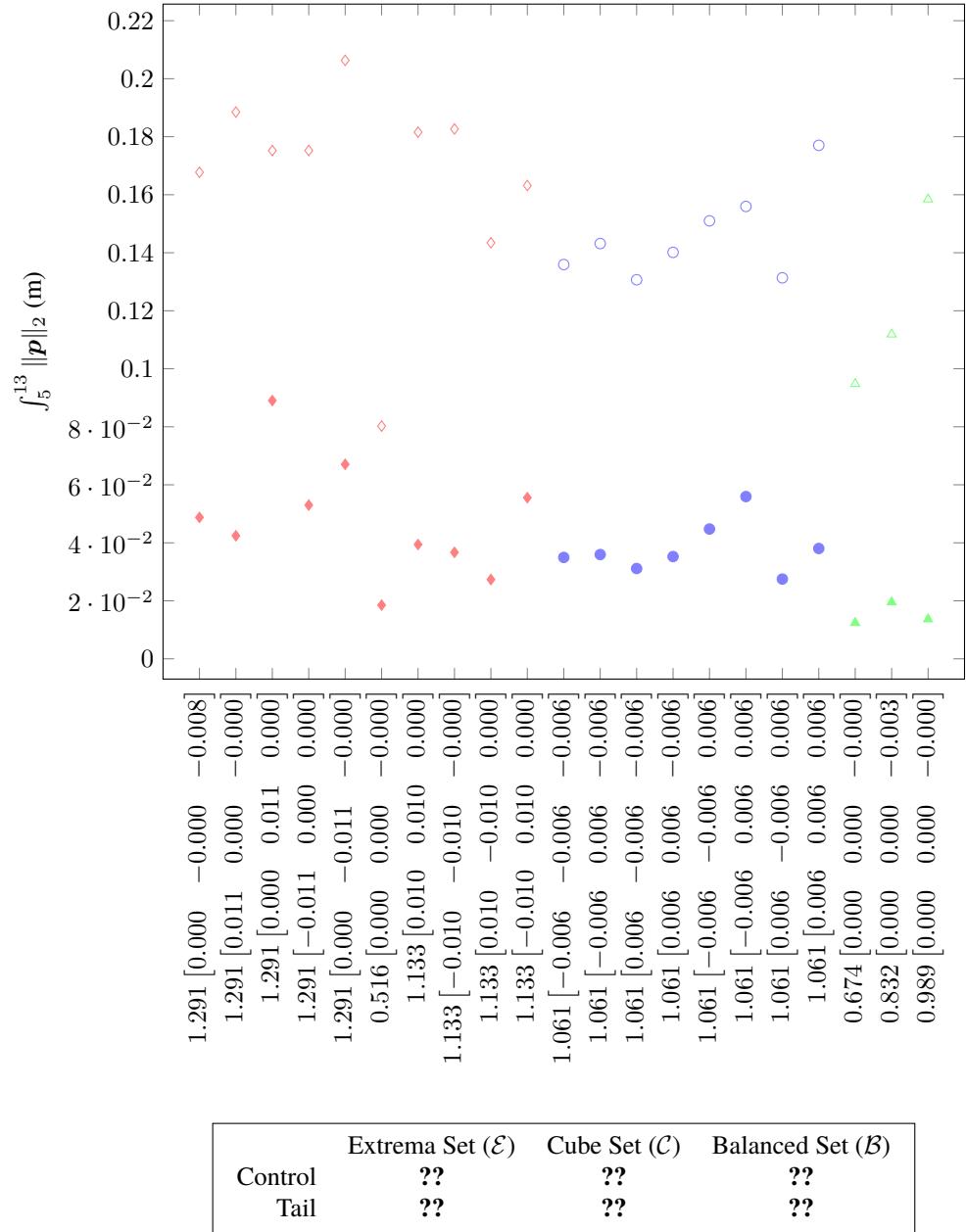


Figure 5.15: Graph of the mean cumulative integrated COP 2-norm over the *carry interval* of the experiment, with and without the robot tail.

5.7 Discussion

5.7.1 Payload and Tail Angle Regression Analysis

To examine the effects of varying the payload mass and center of mass on the tail trajectory, a regression study was conducted comparing mass and center of mass *inputs* to tail angle *outputs*, specifically the peak angle output measured by the maximum magnitude function \wedge ,

$$\wedge(\theta) = \begin{cases} \max \theta & |\max \theta| \leq |\min \theta| \\ \min \theta & |\max \theta| > |\min \theta| \end{cases}, \quad (5.11)$$

where θ represents the vector of tail angle measurements from the experiment:-

$$\wedge(\theta) = \begin{cases} \max \theta & |\max \theta| \leq |\min \theta| \\ \min \theta & |\max \theta| > |\min \theta| \end{cases}$$

\wedge

Figure ?? shows the correlations between the maximum magnitude of the tail on both pitch and roll axes, and the mass and corresponding center of mass of the payload. A change in the x component of the center of mass would only affect the roll of the tail, and the y component would only affect the pitch, but a change in mass could effect both, though the y component to a much greater degree.

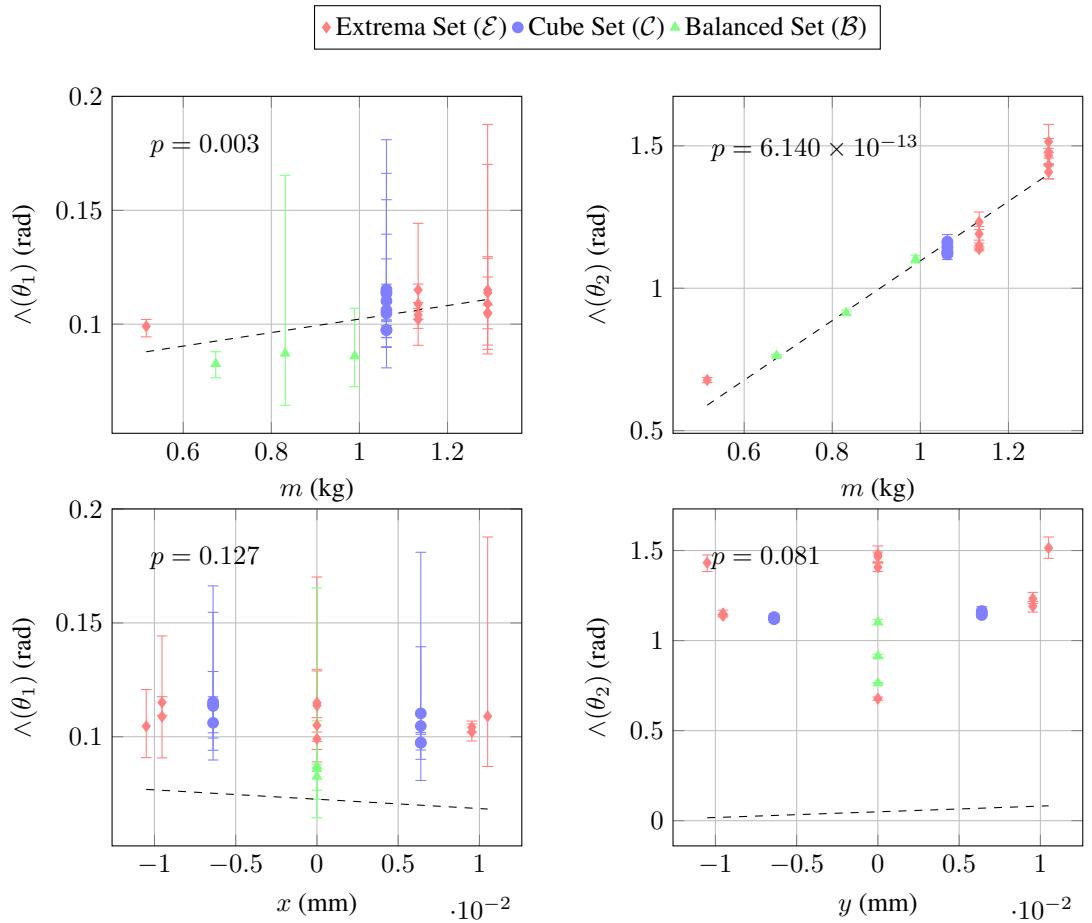


Figure 5.16: Linear regression plots of the tail angles θ against the mass and x and y COM components of the payload, with the center of mass axis plotted against the orthogonal tail axis.

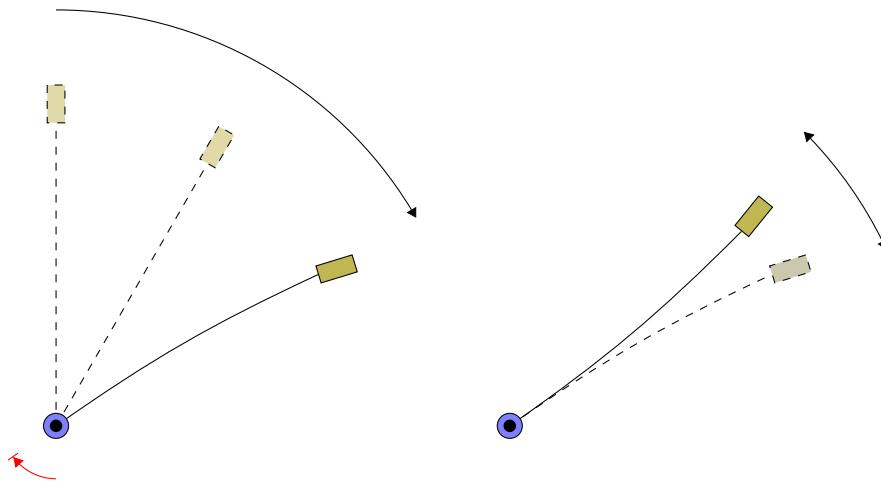
The results show that there is a strong correlation $p = 6.140 \times 10^{-13}$ between the payload mass and the pitch θ_1 maximum magnitude of the tail of $1.047 \text{ rad kg}^{-1}$, and the payload mass and roll maximum magnitude θ_2 of the tail of $0.030 \text{ rad kg}^{-1}$ with $p = 3.223 \times 10^{-3}$. This shows that changing the payload mass had a strong and significant effect on the behaviour of the control system, demonstrating the robustness of the control system over a range of different payload masses.

For the payload center of mass, the correlations were much weaker, $p = 0.127$ for θ_2 and

the center of mass x component, and $p = 0.081$ for θ_2 and the center of mass y component. While the p value from the latter is relatively close to statistical significance compared to the former, it is not enough to rule out the null hypothesis. Therefore it cannot be said with any certainty that adjusting the payload center of mass had any effect on the behaviour of the control system. This is likely due to the small center of mass range, as discussed in section ??, combined with sensor noise masking small changes in load on the load cells and the vibration and control issues discussed in section ?? and. Increasing the payload center of mass range, reducing vibration and improving the tail control system stability are all likely to improve these correlations.

5.7.2 Vibration Analysis

A major issue with the experiments that likely prevented the collection of more accurate data and more stable control of the tail was the vibration of the tail shaft, as shown in figure ???. This imposed a semi-regular oscillating signal on the center of pressure data, which could be reduced by averaging several experimental runs as was done for these experiments, but was impossible to filter out completely. This was due to greater than expected elastic deformation of the shaft, caused by stress induced when the tail accelerates and decelerates, as was noticed from visual observations of the experiments. A diagram illustrating this effect is shown in figure ??.



1. The tail decelerates approaching a position. This creates stress on the tail shaft and causes elastic deformation, causing the shaft to bend slightly.
2. The restoring force then bends the shaft in the opposite direction, resulting in a slowly decaying oscillation

Figure 5.17: Diagram of the cause of the tail oscillation during a deceleration. Effect has been exaggerated to make it visible.

In order to characterise this vibration, the tail was removed from the system and subjected to an impact test in order to find the damped natural frequency. This was achieved by fixing the base of the tail and striking the tip mass with a small steel hammer. The BNO080 inertial measurement unit was fixed above the mass and the accelerometer data was recorded.

Figure ?? shows the resulting y component of the accelerometer data, and a discrete fourier transform of the data shows the damped natural frequency.

Figure 5.18: Damped natural frequency of the tail calculated from accelerometer data captured during an impact with a small steel hammer. The left graph is the accelerometer data, and the right graph is the real part from a DFT on the data.

This differs from the vibration frequency measured during the tail experiments from the load cells. By taking a similar discrete fourier transform from the y component of the center of pressure measurement

$\Re(\mathcal{F}((CP(\mathbf{F})_y)))$, which had the strongest measure of vibration, a noticeable harmonic was observed around 0.85 Hz for each experiment, as shown in figure ??.

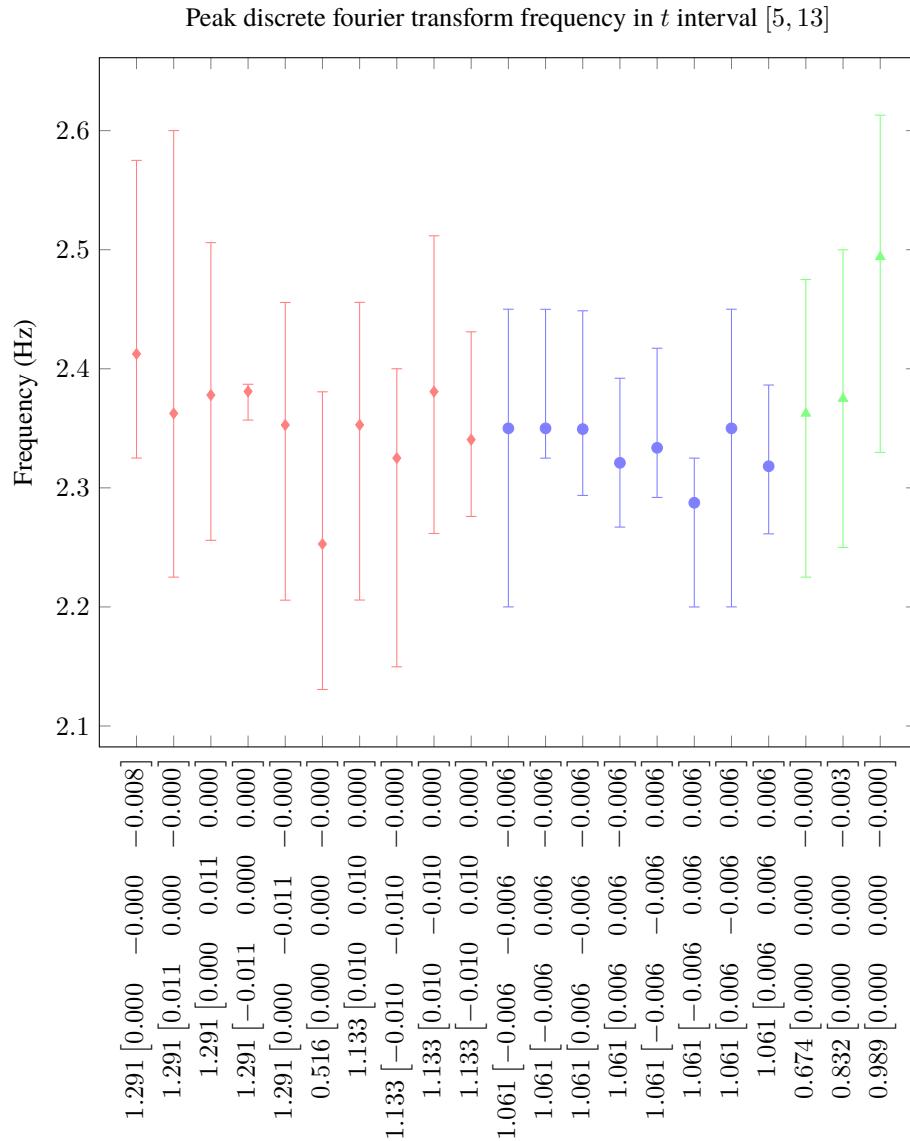


Figure 5.19: Mean frequency of the maximum absolute real value discrete fourier transform of each tail experiment in the t interval [5, 13].

The presence of a consistent harmonic oscillation within the load cell data is likely to be at least partially the result of the tail vibration. To reduce the tail vibration, two approaches can be taken, either generate smaller stresses on the tail shaft, by lowering the maximum angular acceleration of the tail, or improve the ability for the system to absorb the energy

from these forces. The former approach would reduce the responsiveness of the control system, and therefore reduce the ability for the tail to restore stability in an adequate timeframe in a real world application, i.e. the mobile robot may tip over before the tail could prevent it from doing so. The latter approach is more technically challenging, and could require major modifications to the tail design, actuation, and control system, but would maintain or even improve the performance.

Increasing the stiffness of the tail shaft would reduce the amplitude of the oscillation by providing greater resistance to deformation, but would also transfer more load torque to the motors, so this would need to be compensated for. Magnetic brakes for electric rotary motors is one potential solution, or using hydraulic or pneumatic motors where more fine control over the torque output is easier. An advanced torque controller may also be able act as a ‘virtual’ magnetic brake by providing a suitable opposing torque.

There are also linear actuator designs, such as the pneumatic muscle actuator that can have damping properties due to the viscoelasticity of the cell material **sarosi2014study**, **okui2018fundam**
okui2017variable, as well as the muscles that are found in biological systems, and could possibly explain how animal tails can manage smooth, vibration free motion even under sizable dynamic loads **taylor1990viscoelastic**, **van2008viscoelastic**, **van2009viscoelastic**. Viscoelasticity is a material property where there is a hysteretic stress-strain response, see figure ???. This allows it to dissipate energy as heat, which would have a damping effect on any vibration.

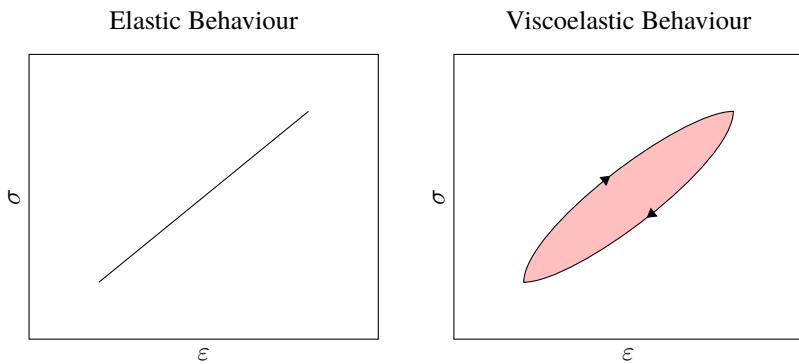


Figure 5.20: Stress-strain response for an elastic and viscoelastic material. The shaded area is equivalent to the amount of energy lost in one cycle.

5.7.3 Control System Stability

While the mechanical vibration of the tail shaft was the most significant contribution to the overall vibration of the system, there was some oscillation within the motor proportional integral derivative controllers which created some additional vibration. Figure ?? shows a strong oscillation with a maximum range of approximately 100 pulses, or 8.8° , which corresponds to an output shaft oscillation of $\approx 0.46^\circ$. This was small, but potentially enough to significantly contribute to the vibration issue.

After a time gap in experiments between the extrema and cube set, and the balanced set, the

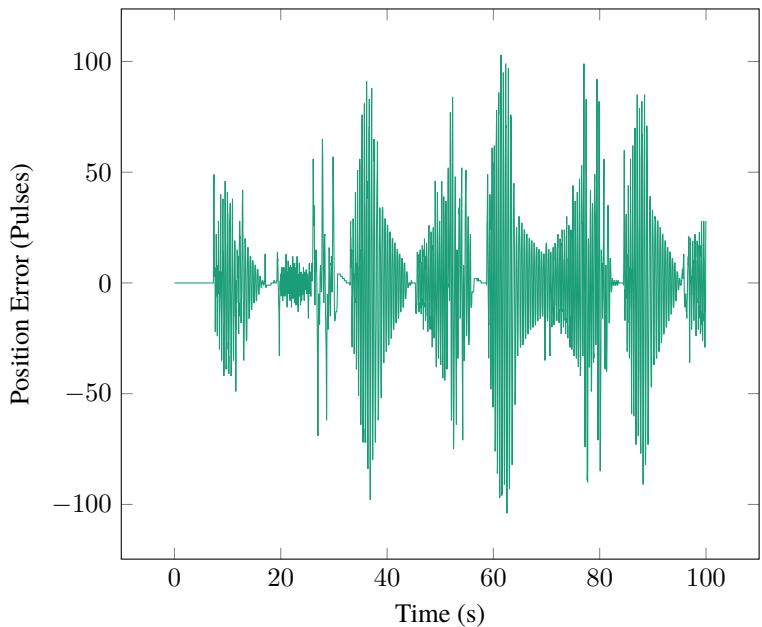


Figure 5.21: Data captured from one of the tail motor controllers during the \mathcal{E}_3 experiment run with a tail, showing the motor position error (setpoint - response) in motor pulses (where 1 pulse = $\approx 0.088^\circ$) for a 100 second interval, capturing several experimental runs.

oscillation got significantly worse, to a point where the system was uncontrollable, and despite a mechanical inspection of the system to tighten loose connections, the problem persisted. At this time the cause is still unknown. However, reducing the value of k_{d_m} from 3 to 2 improved the oscillation enough to resume the experiments, and did not appear to have a major effect on the results.

5.7.4 Incorporation of Tail into a Mobile Robot

The evolution of these static experiments, once robust and effective tail control is achieved over a wide range of payload mass and center of mass, will be to incorporate the arm and tail on a mobile robot platform, in order to conduct a test of the tail's capabilities in a more realistic environment. The load cells will be fitted to each individual leg or wheel to calculate the center of pressure, in a manner that passes all of the ground reaction force from that appendage to the robot body through the load cell. If there are more than four legs or wheels, the function to calculate the center of pressure becomes more complex, but is still fairly trivial. Depending on the environment, 3-axis load cells may have to be used, as if the robot is on a slope the Z axis of the load cell will not line up with the gravity vector. Figure ?? illustrates how the arm and tail may be mounted to different mobile robots, in a simplified fashion.

Mounting to a mobile robot will also allow the exploration of other potential issues with the system. For example, if the tail response is too fast, it may cause the robot to tip over when the tail compensates for the payload the moment it is picked up.



Figure 5.22: Illustrative examples of the robot tail fitted to a legged or wheeled mobile robot.

5.8 Conclusion

This work outlined the design and method of measuring stability using four load cells, then designed a system where a robot arm carries a payload with and without a 2 degree of freedom robot tail to minimise the instability. The results with and without a tail were then compared using cost functions, and a significant improvement was demonstrated using these functions. Issues encountered during the experiments were then discussed, and potential solutions were explored in order to acquire even more definitive results.

Chapter 6

Discussion and Conclusion

In this chapter, the achievements and contributions to the field of this thesis are summarised. Future work is then proposed based on both limitations of the existing research in order to improve the quality of the results, and new research that builds on from what has been completed in this thesis.

6.1 Discussion and Conclusion

6.1.1 Novel Contributions to the Research Field

This thesis has contributed, and will contribute pending limited further research, the following novel original research to the field:

- A systematic and comprehensive review of robot tails used in terrestrial mobile robots, with a broad categorisation of their functions. While there is a review paper in this field already **saab_rone_ben-tzvi_2018**, it is much more selective and narrower in scope than the research conducted in this thesis.
- The development and demonstration of robust control of a twisted string actuator based actuated universal joint that demonstrates robust 2 degree of freedom control, investigating the benefits, technical challenges, and making performance comparisons with alternative actuation systems.
- The results of a simulation study investigating if multi-segment tails can track a center of mass trajectory more efficiently than a single segment, either through reduced joint velocity, joint torque, or both.
- The design and control system of a robotic tail that is able to improve the **stability static balance** of a mobile robot while carrying a payload with a robot arm.

6.1.2 Discussion

As discussed in chapter 1, **enhancing stability controlling locomotion** using a tail is found in both the animal kingdom and in mobile robots. After a literature review on the applications of robotic tails for mobile robots in chapter 2, it was found that the vast majority of experimental implementations used single segment tails. There was also a gap identified in the research for using a tail to maintain **robot stability static balance for a robot** when carrying a payload, which validated the novelty of the research in this thesis.

The following research questions were considered:

1. Which robot tails are utilised for mobile robot **stability problems locomotion control**?
2. What is a good actuation method to use for the tail?
3. **Should the tail have a single segment or have multiple segments?**
4. Can a payload be developed to allow the investigation of **stabilising the static balance of** a mobile robot when carrying a payload against a range of payload mass and center of mass?

5. How to measure the **stability static balance** of a mobile robot?
6. Can a robot tail be used to **stabilise improve the static balance of** a mobile robot when carrying a payload?

These were developed into the aims and objectives shown in section 1.2. Table ?? lists these objectives, with an indicator of the “level of achievement” that was made in this PhD, and any publications that were submitted from the work conducted for this objective. *Not Achieved* indicates that results or findings were inconclusive, and that further research is required to draw conclusions with any reasonable level of confidence. *Partially Achieved* indicates that results or findings allowed conclusions to be drawn with a moderate level of confidence, and that further research would provide a stronger foundation for any conclusions. *Fully Achieved* indicates results or findings led to conclusions with a high level of confidence, and that while further research is still beneficial for the field, it is not necessary.

No	Objective	Question	Achieved?	Publication
1	Investigate the current uses of robotic tails in mobile robots in order to gain insight into further research.	1	●	A Scoping Review of Robotic Tails for Land-based Mobile Robot Locomotion, <i>Submission Pending</i>
2	Investigate potential actuation methods for the robotic tail and select a suitable method.	2	●	icra , <i>Evolved Publication to Follow</i>
3	Investigate if there is any advantage to using a multi-segmented tail. 3-Minimisation of Velocity and Torque 2-Norm for a Multi-Segment Robot Tail Tracking a Center of Mass Trajectory (Provisional Title), Submission Pending, Requires Additional Research	4.3	○	
4	4-Design a configurable payload that can simulate a range of mass and center of mass.			
5.4	5.4-Design a system that can measure stability static balance in a way that could be easily used on a mobile robot platform.	5.4	●	
6.5	6.5-Compare the stability static balance performance of a system that is able to pick up, carry and set down the payload with and without a robot tail that uses the measured stability static balance as a closed loop control signal.	6.5	○	Use of a Robot Tail for Stability Improving the Static Balance of a Mobile Robot when Carrying a Payload (Provisional Title), <i>Submission Pending, Requires Additional Research</i>

○ Not Achieved ○ Partially Achieved ● Fully Achieved

Table 6.1: Table of all the objectives in section 1.2, their achievement progress, and any publications that resulted from the work.

6.1.2.1 Summary of Research

Chapter 2 achieved objective **1** by conducting a literature review of robotic tails used in terrestrial robots, with a focus on tails that influenced robot dynamics. The research found that there was a very diverse set of tail functions for terrestrial robots, such as maintaining the robots heading while walking **zhang2016effects**, allowing the robot to turn more quickly **pullin2012dynamic**, **kohut2013precise**, and reorienting the robot in mid-air **chang2011lizard**. These functions were categorised into a tree-like system as shown in figure 2.3, based on the robot's environment, the action it was attempting to perform, and what the tail was doing for that action. Observations were also made about the performance comparisons made in some publications, as the level of bio-inspiration. The main observations that would influence the later research were that multi-segment tails were very rare on mobile robots, with only two examples found in the literature **sato2016development**, **simon2018development** which used their tails to increase their jumping height. Other multi segment tails were either only simulation studies, or were static prototypes, sometimes for potential future mobile robots. While the research field is clear of the potential of robotic tails for improving performance, to date there are few systems which have experimentally implemented multi-segment systems, due to the complexity and robustness required for their design and control.

?? **PythonTeX** ?? It also compared the twisted string actuator to alternative actuation methods for the tail joint actuator, using linear leadscrews and a “direct drive” with motors directly rotating the joint, and discovered the performance of the twisted string actuator compared to these alternatives depended on the stroke position, or number of twists in the twisted string actuator. However, the design was not resilient enough for the **payload stability static balance** experiments in chapter ??, and such a compact design was unnecessary for a single segment tail. The range would also need to be significantly increased, as $\pm 11^\circ$ on a single axis, and $\pm 6^\circ$ on both axes would not be sufficient for the experiments. This would have required additional work, as this research did not directly contribute to the primary research aim, and so an alternative existing actuator design was used instead based on the design in **patel2015conical**. A single segment design was investigated as a precursor to a multi segment system, unfortunately the impact of COVID-19 limited further investigations. Nevertheless, it still provided valuable novel research that contributes to the field, even if its best applications may be outside the focus of this thesis. This research resulted in a conference proceeding accepted for the *IEEE Conference on Robotics and Automation (ICRA) 2022*, see appendix ?? for the accepted manuscript.

Chapter ?? then attempted to achieve objective **3**, and was unable to find a practical advantage for multi-segment tails based on a simulation study attempting to minimise the joint velocity 2-norm, as while reductions in both peak and mean were measured, they were significantly negated by the increase in joint torque 2-norm. However, the results from the modelling and simulation were still inconclusive due to issues with the quadratic programming optimisation algorithm, so an initial investigation with a single segment tail was used

for the experiments in chapter ??, based on a bevel gear design found in **patel2015conical** that was able to mount the motors to the robot body, giving the tail joint access to the full torque limit of each motor. Unfortunately the impact of COVID-19 limited further investigations into additional segments.

Chapter ?? partially achieved objective 4, designing and fabricating a payload consisting of a container filled with cubes of various materials of different densities, allowing the payload to change its mass and center of mass based on the location and density of the material of the cubes inside it. Test points were then generated, some by taking the extreme points of the configuration space generated by the payload, others using a brute force search method to find the nearest point in the space given a target mass and center of mass. While the resulting payload had a significant mass range of [0.38, 2.90]kg, the center of mass range was only [-13.44, 13.44]mm on all axes. This was insufficient for creating notable variations in the results for the **payload stability static balance** experiments in chapter ??.

Chapter ?? then completed objective 2 by selecting the actuation system for the tail. A “Static Rig” was designed and fabricated that could measure the **stability static balance** of the system using four load cells, which achieved objective 5. A robot arm, a Kinova™ MICO² 4degree of freedom, was then mounted to the setup, and picked up, held, and put down the payload for each test point multiple times to ensure repeatability. The experiments were then repeated with the final tail design, that used the load cell data as a closed loop control system to **minimise the system instability maximise static balance**. The results were compared using two cost functions, and a clear improvement in **stability static balance** was noticed for all test points, despite control and vibration issues, which partially achieved objective 6. For the heaviest payload with a mass of 1.25 kg, there was a 88.17% reduction in the y component of the mean minimum COP deviation, and a 76.37% reduction in the mean cumulative COP deviation 2-norm, both cost functions used to quantify the stability of the system. This proved the hypothesis proposed at the beginning of the thesis, and represents the main scientific contribution of this work.

6.1.3 Conclusion

In conclusion, this thesis has demonstrated the efficacy of using a robotic tail for **maintaining robot stability improving the static balance of a robot** while carrying a payload at the fundamental level with limited but definitive experimental data. The data from section ?? clearly shows a consistent, statistically significant improvement based on the cost function criteria defined therein, over a range of different payload mass and center of mass. This can be used as foundational research to move towards simulated field experiments as discussed in section ??, and hopefully to be incorporated into various mobile robots with payload carrying applications and in various environments. This thesis has also developed, as part of the investigation into potential tail actuation methods, a compact, lightweight actuated universal joint based on the twisted string actuator, and while it was not selected for use in the **payload stability static balance** experiments in chapter ??, it can still be used to create com-

pact multi-segment robotic tails, as well as other robotic mechanisms such as mobile snake robots.

6.2 Future Work

6.2.1 Additional Experiments for the twisted string actuator

Additional experiments will be carried out to increase the joint range and angular velocity of the mechanism. There will also be work to further characterise the control system, including open and closed loop bandwidth, and to test the load limits of the mechanism by adding mass to the follower body. It is planned to commence this work in April 2022, the results of which will be added to an evolved publication from the conference proceeding, to be submitted to IEEE Transactions on Robotics (T-RO) sometime in Q2 2022.

6.2.2 Development of a Multi-Segment Tail based on the twisted string actuator Mechanism

The eventual goal of the twisted string actuator research would be to chain multiple segments together, to create a lightweight, compact, powerful robotic tail that could have a wide range of potential applications, not just the application in this thesis. This would be coupled with improvements to the design, sensing and control hardware of the system to create a more advanced prototype that would be significantly closer to a product with practical applications.

6.2.3 Further Research into Multi-Segment Optimisation

The results obtained from the multi-segment optimisation simulation study in chapter ?? were unsatisfactory, since it was not possible to optimise for the torque 2-norm. Further investigation into the optimisation algorithms is clearly needed in order to rectify this issue, then it can be proven if multiple segments can have any performance advantage for the application in this thesis.

6.2.4 Further Experiments on the Static Rig with an Improved Payload with Greater center of mass Range

Using an improved payload as discussed in section ??, further experiments could be carried out on the static rig, with improvements made to the actuators and controllers the limits of the system could be better understood. Experiments at various values of a_y and t_y as shown in figure ?? in section ?? could also be carried out, to simulate mobile robots of different lengths.

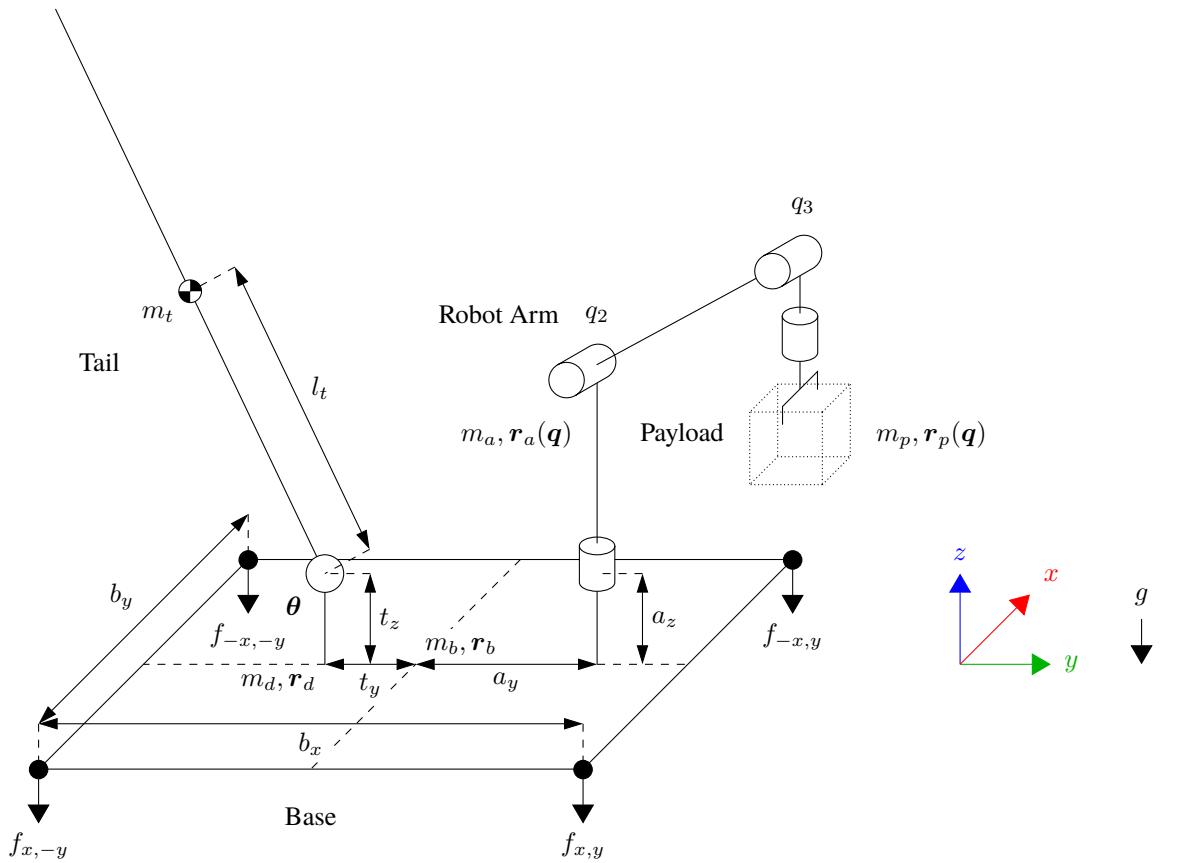


Figure 6.1: Free body diagram from chapter ??.

6.2.5 Evolution of the Static Rig into Simulated Field Trials on a Mobile Robot

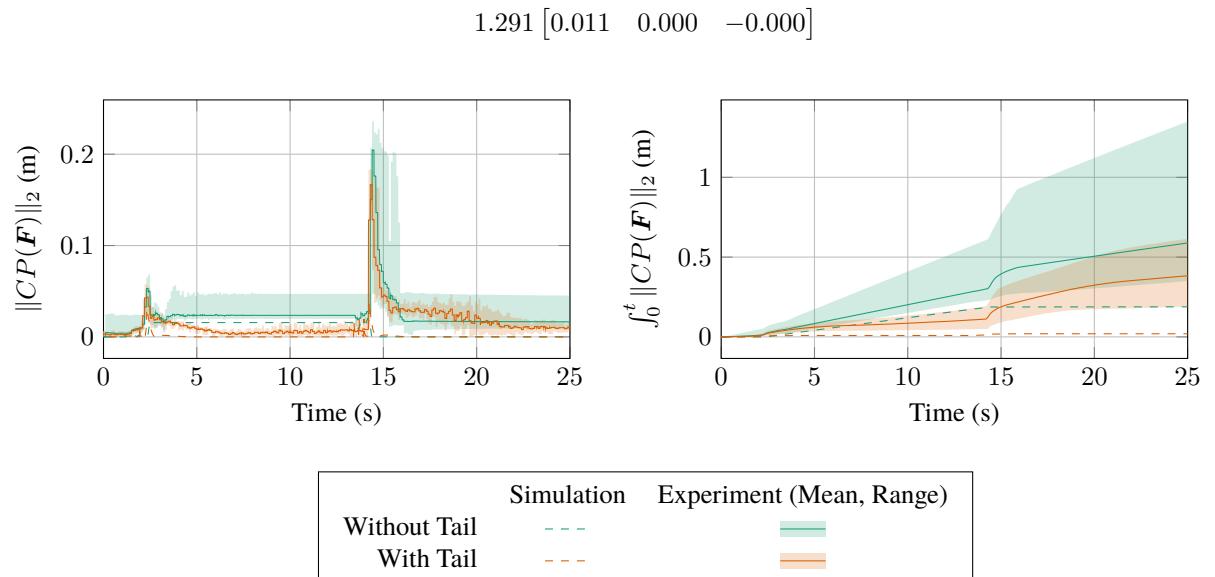
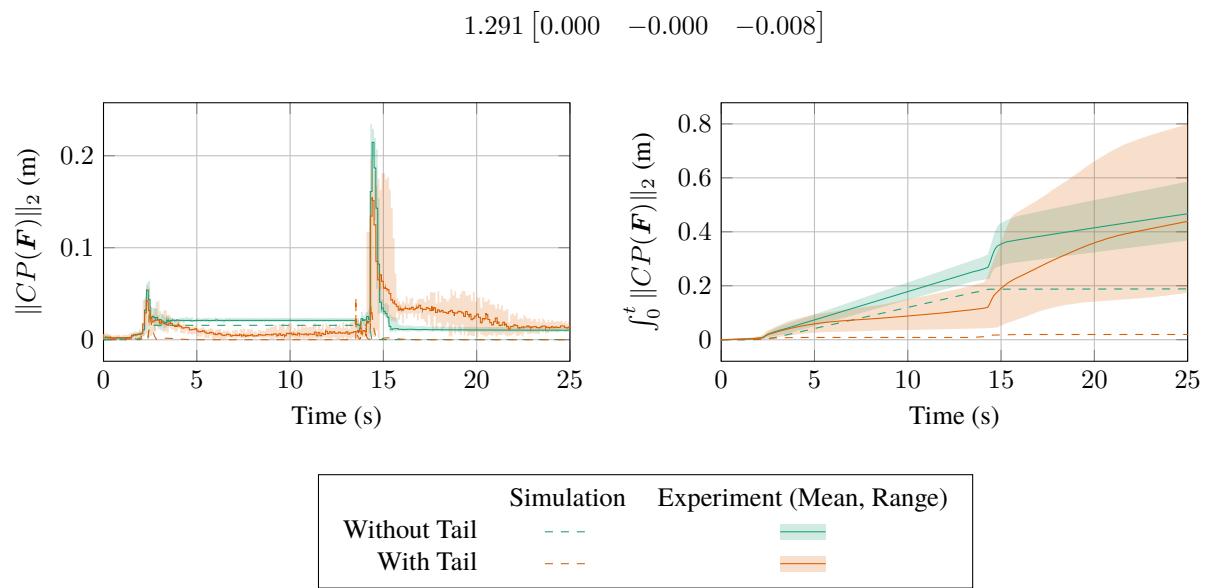
The eventual goal for the line of research in this thesis would be to prove the efficacy of a robotic tail for stabilisation when carrying a payload under near “real world” conditions. This would entail placing the robot arm and tail mechanism on a mobile robot base, either legged or wheeled, and then constructing a simulated environment where such a robot may be carrying out its tasks. This could have hazards such as uneven terrain, loose rubble, or confined spaces. Operators could then be challenged to transport various payloads of different mass and center of mass from one location to another, possibly impaired by having no line of sight and relying on onboard cameras. Robot and operator performance could then be recorded and scored, based on time to complete the task and number of “faults”, especially if the robot tips over due to **instability****loss of balance**. By repeating the experiments numerous times, with and without a robotic tail, average scores collated with data collected during the tests from sensors such as load cells and IMU which are able to measure **stability directly****static balance**, could be used to determine the true efficacy of the robotic tail in a realistic environment.

Appendices

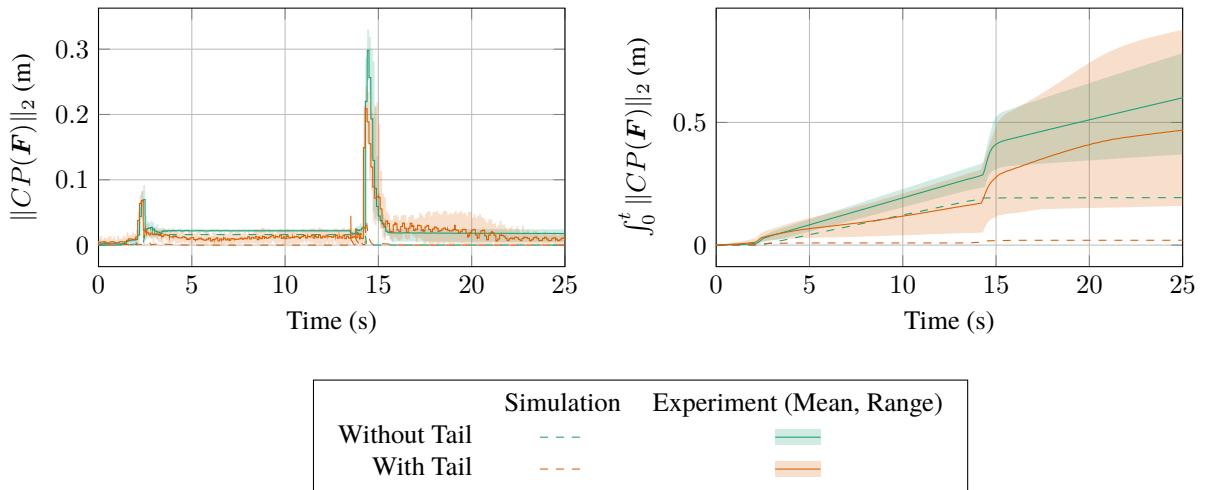
Appendix A

Graphs of Each Test Point Experiment from Chapter ??

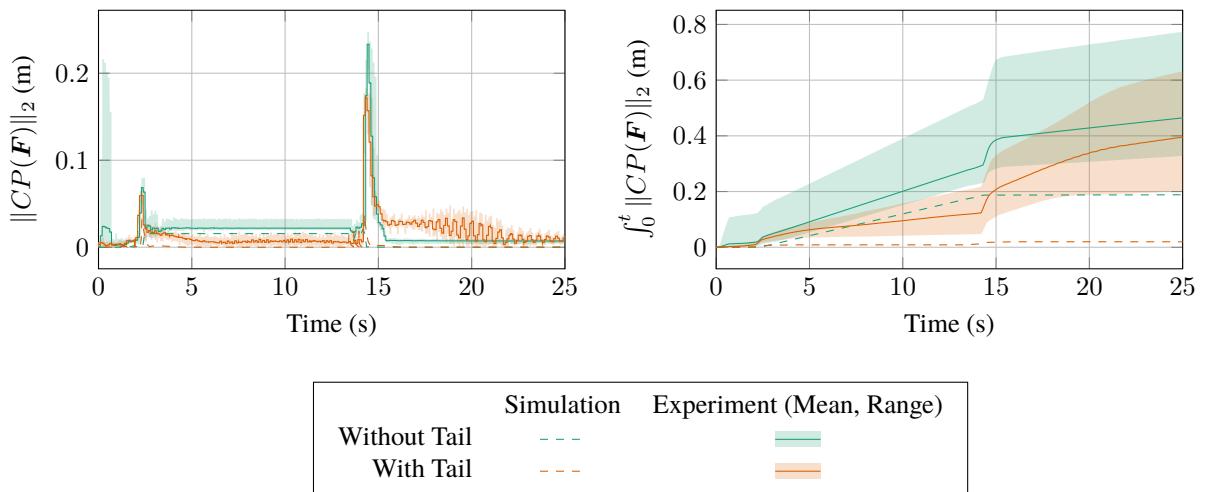
A.1 Extrema Set (\mathcal{E})



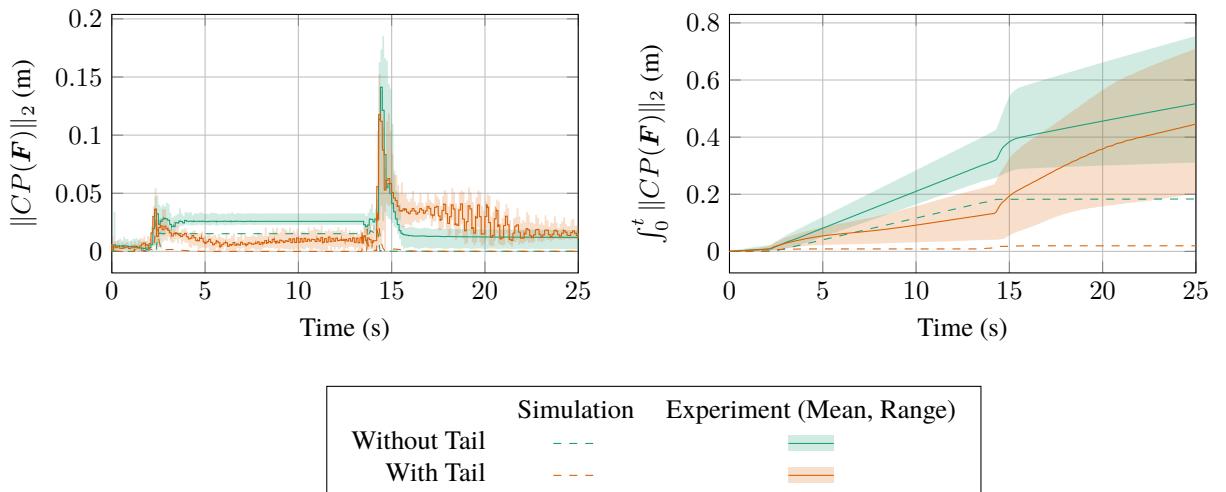
$$1.291 [0.000 \quad 0.011 \quad 0.000]$$



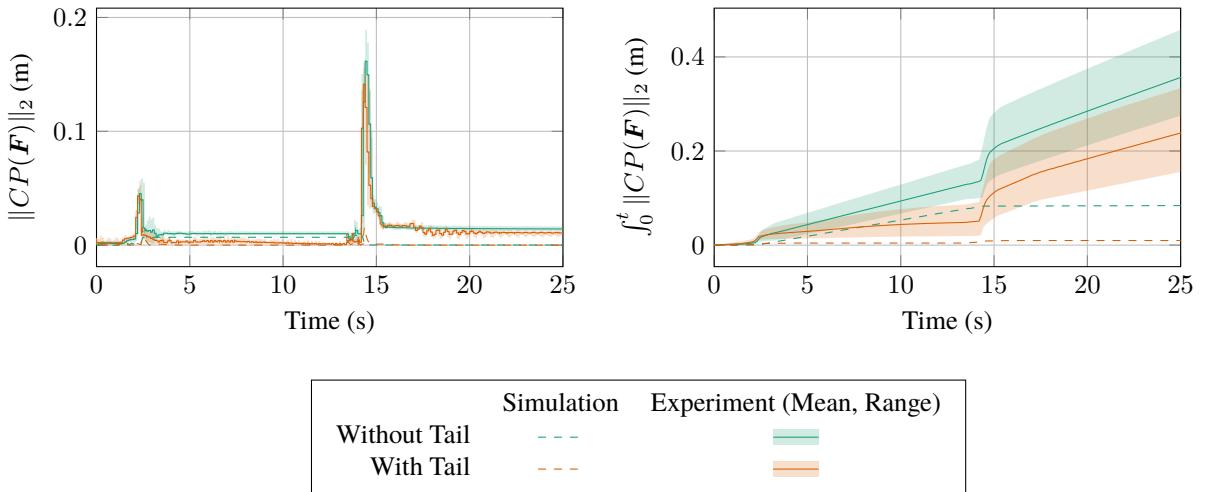
$$1.291 [-0.011 \quad 0.000 \quad 0.000]$$



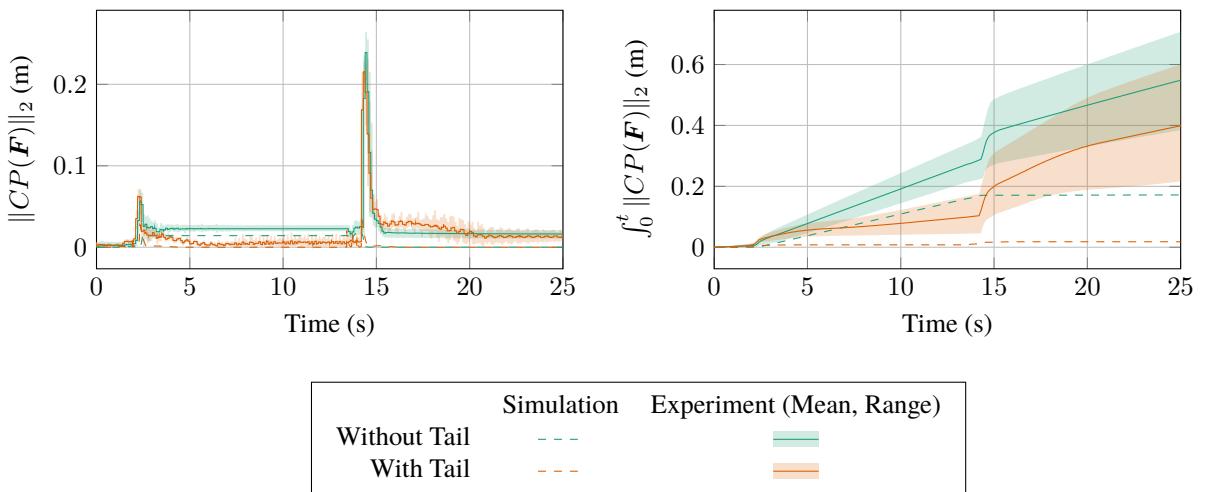
$$1.291 [0.000 \quad -0.011 \quad -0.000]$$



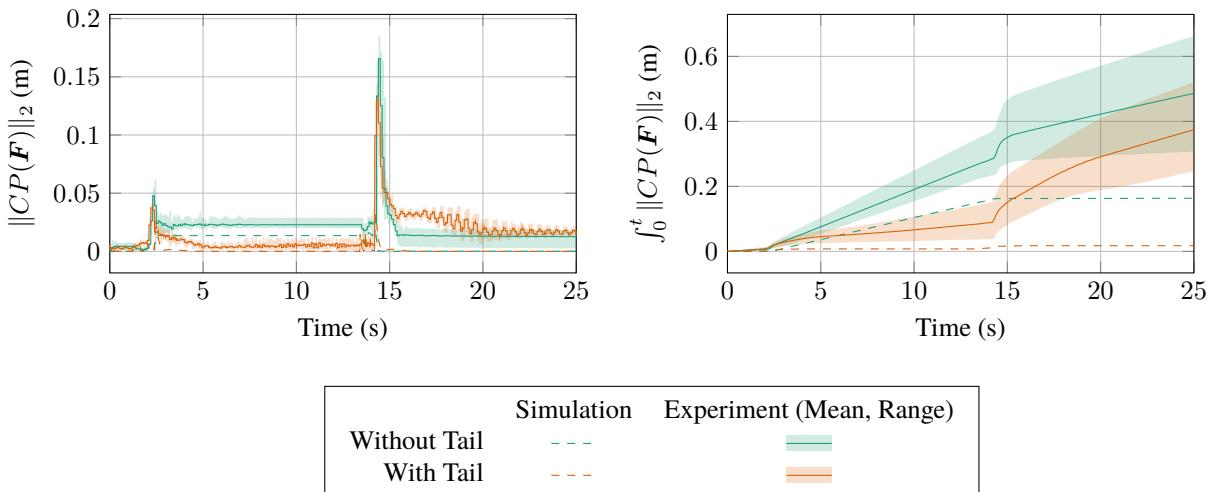
$$0.516 [0.000 \quad 0.000 \quad -0.000]$$



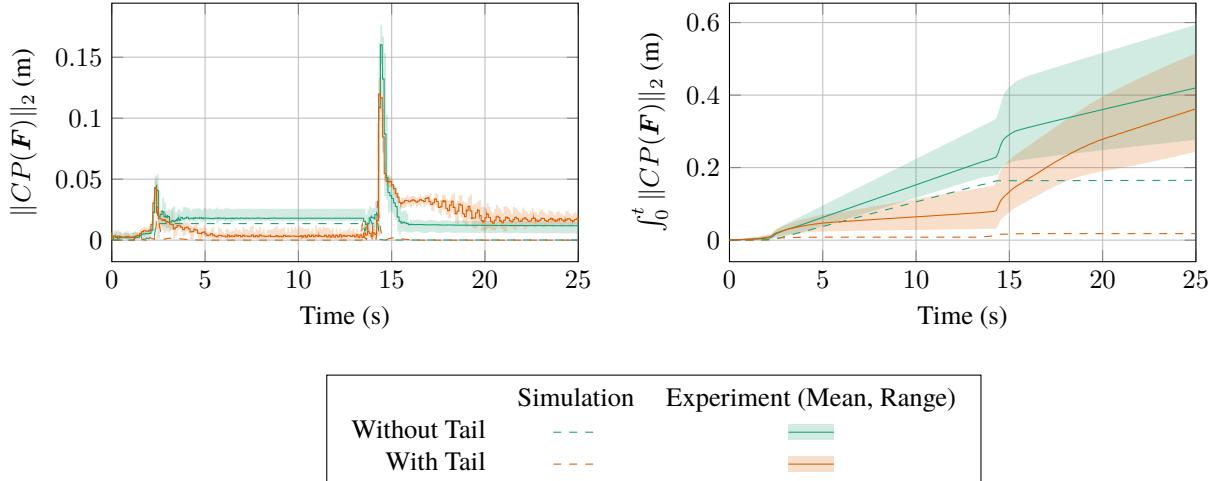
$$1.133 [0.010 \quad 0.010 \quad 0.000]$$



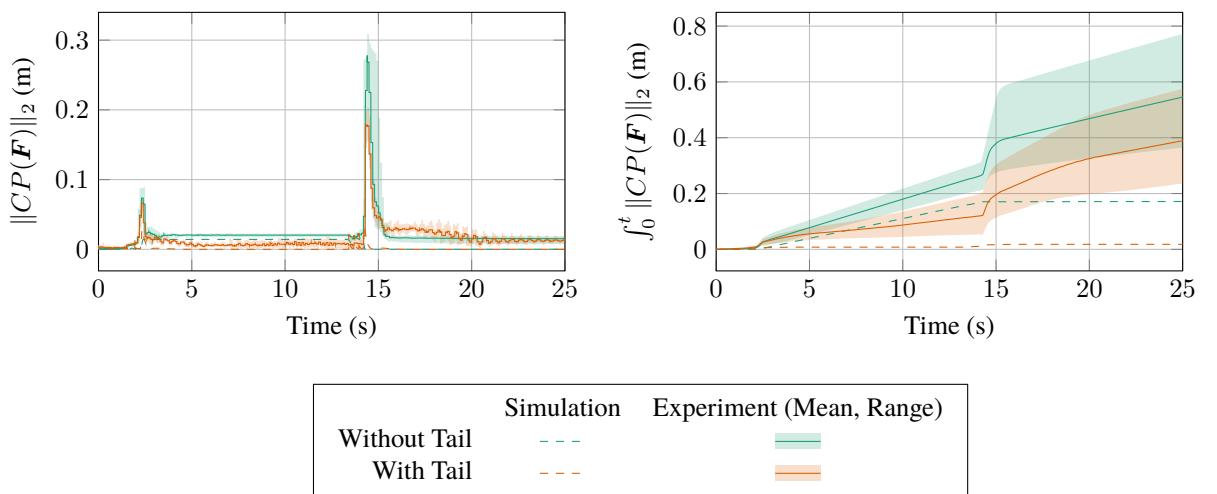
$$1.133 [-0.010 \quad -0.010 \quad -0.000]$$



$$1.133 [0.010 \quad -0.010 \quad 0.000]$$

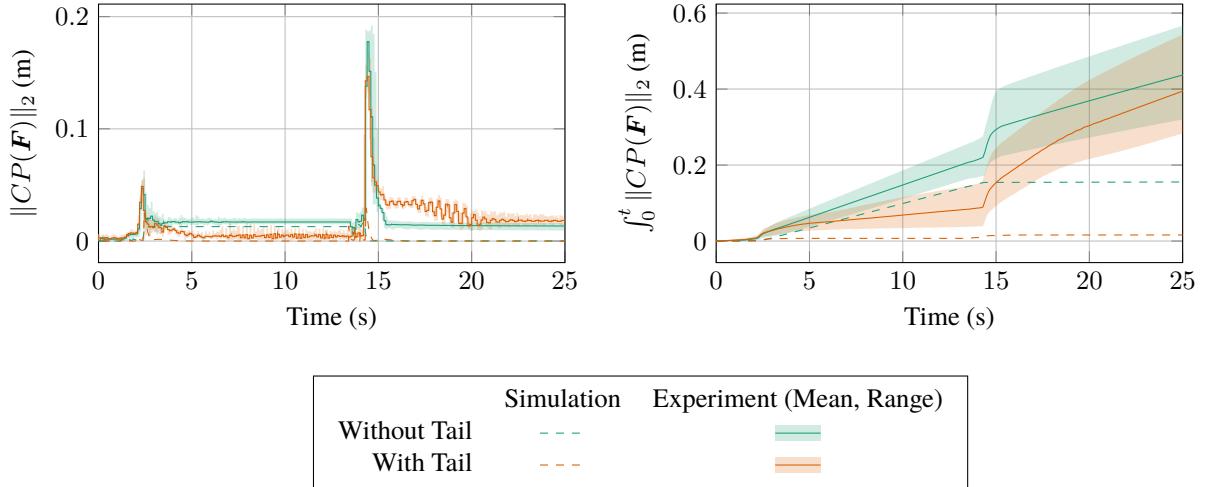


$$1.133 [-0.010 \quad 0.010 \quad 0.000]$$

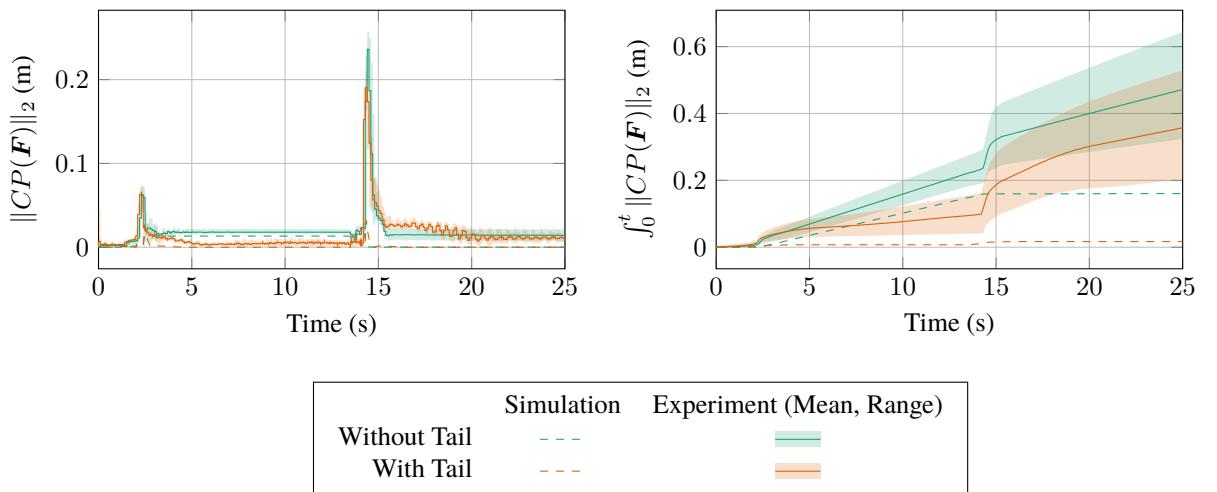


A.2 Cube Set (\mathcal{C})

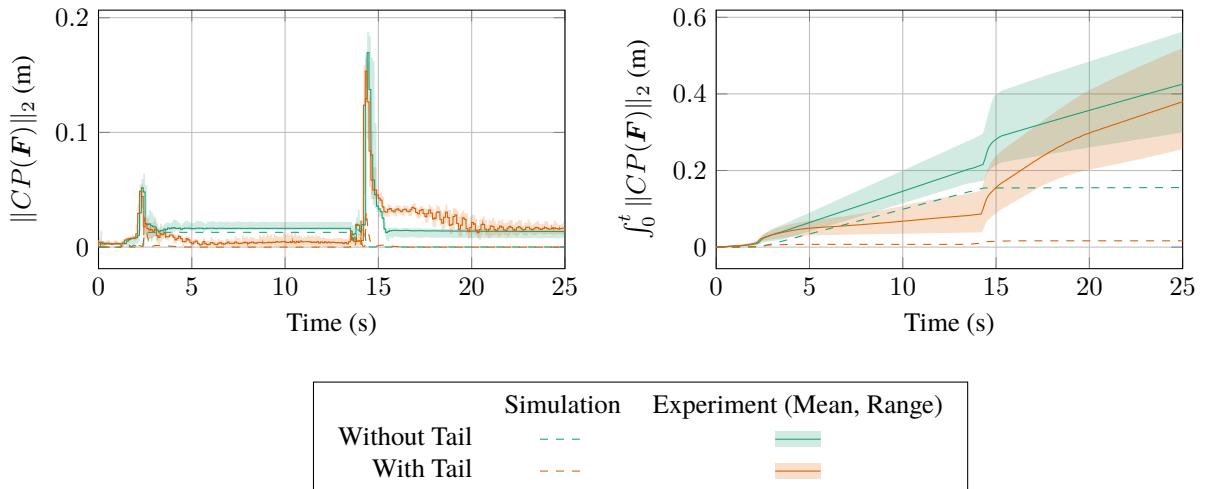
$$1.061 \begin{bmatrix} -0.006 & -0.006 & -0.006 \end{bmatrix}$$



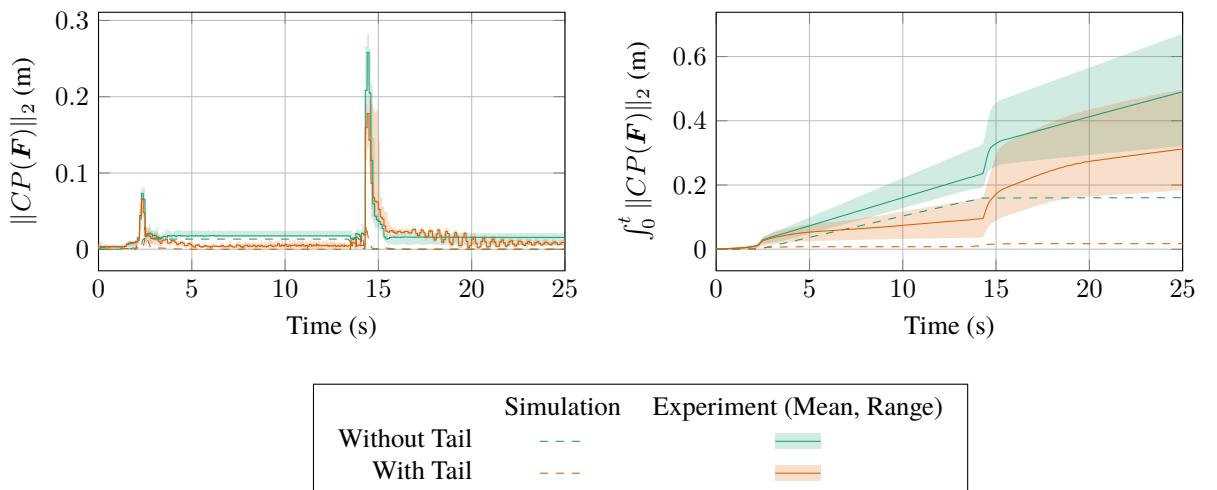
$$1.061 \begin{bmatrix} -0.006 & 0.006 & -0.006 \end{bmatrix}$$



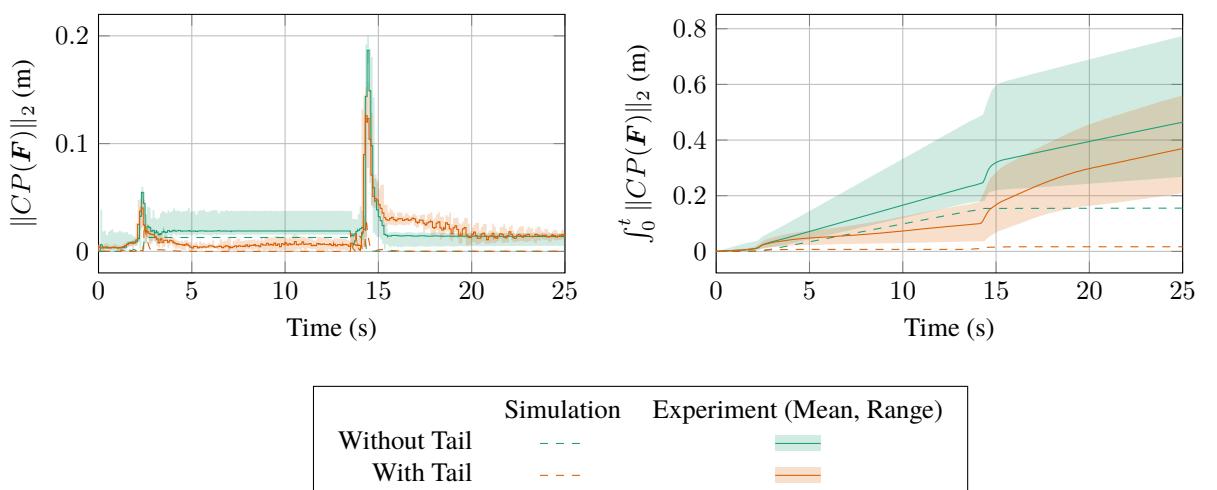
$$1.061 [0.006 \quad -0.006 \quad -0.006]$$



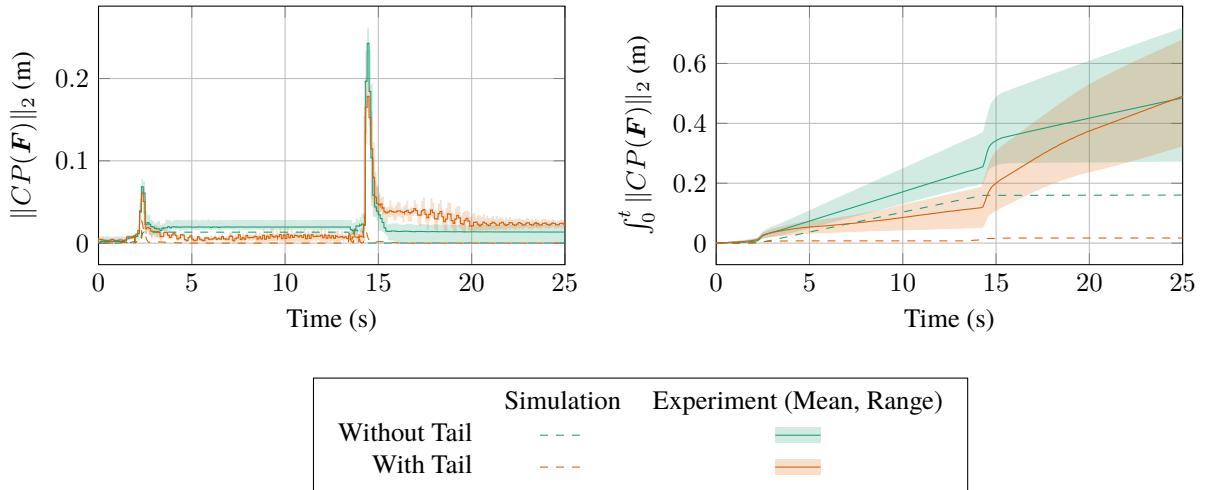
$$1.061 [0.006 \quad 0.006 \quad -0.006]$$



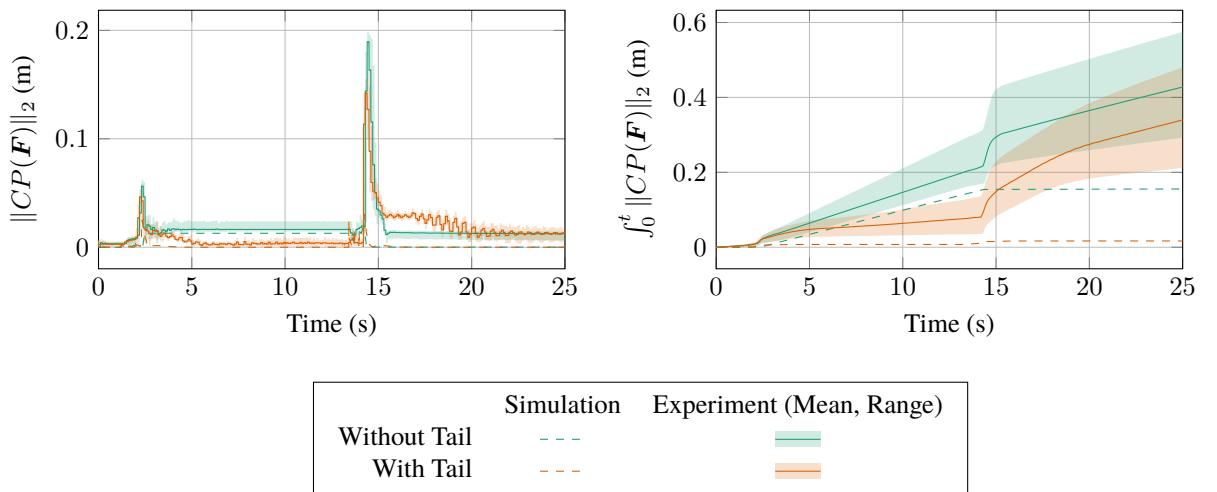
$$1.061 [-0.006 \quad -0.006 \quad 0.006]$$



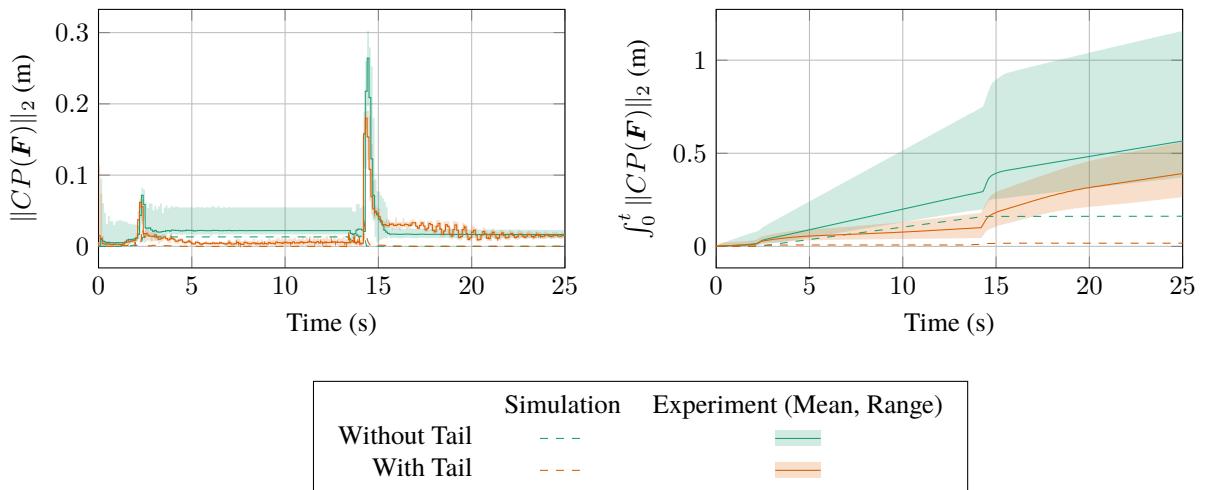
$$1.061 [-0.006 \quad 0.006 \quad 0.006]$$



$$1.061 [0.006 \quad -0.006 \quad 0.006]$$

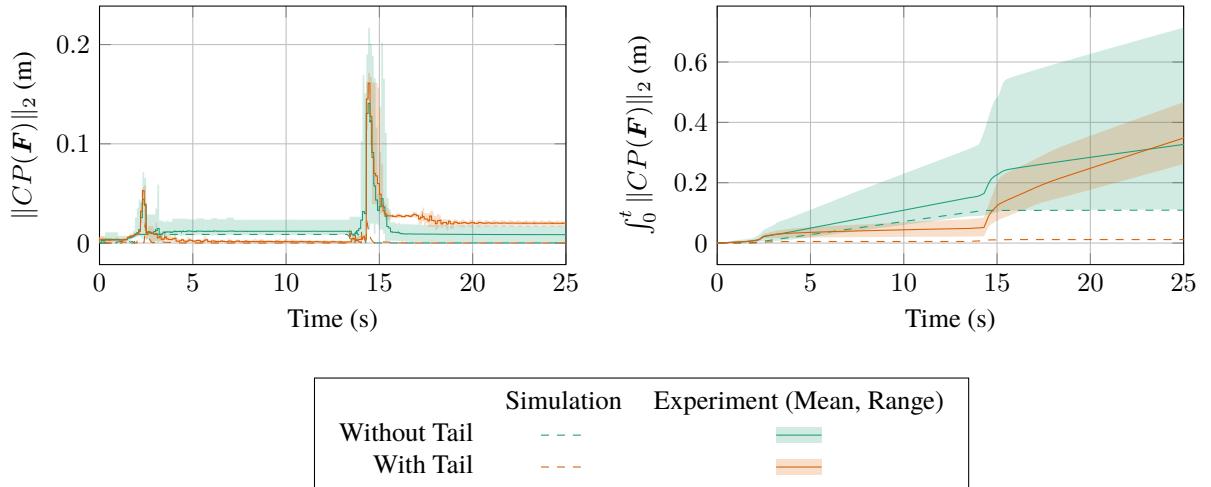


$$1.061 [0.006 \quad 0.006 \quad 0.006]$$

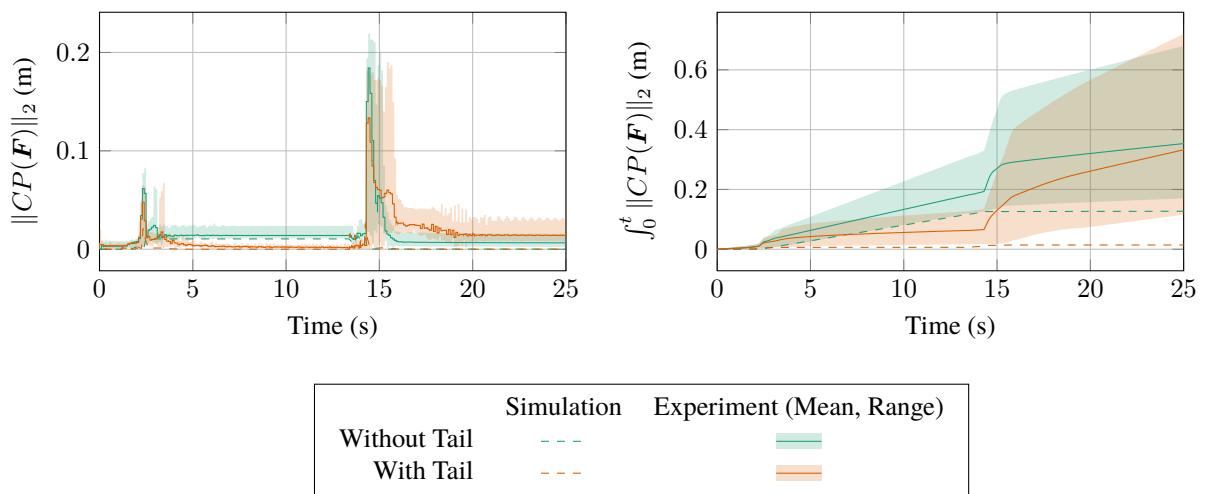


A.3 Balanced Set (\mathcal{B})

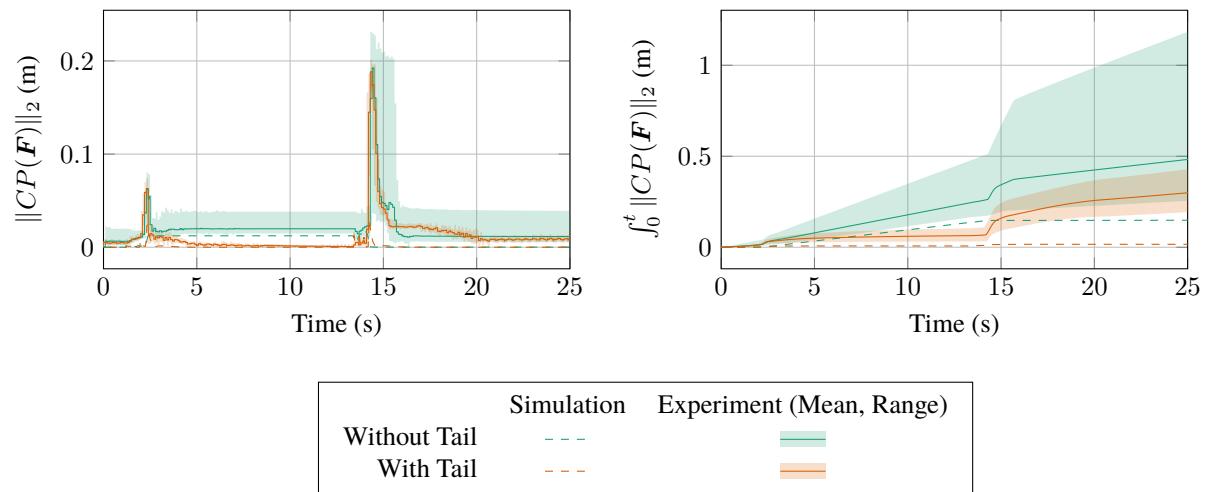
$$0.674 [0.000 \quad 0.000 \quad -0.000]$$



$$0.832 [0.000 \quad 0.000 \quad -0.003]$$



$$0.989 \begin{bmatrix} 0.000 & 0.000 & -0.000 \end{bmatrix}$$



Appendix B

ICRA Conference Paper from Chapter 3

A Novel Triad Twisted String Actuator for Controlling a Two Degrees of Freedom Joint: Design and Experimental Validation

Damian Crosby¹, Joaquin Carrasco², William Heath², and Andrew Weightman¹

Abstract— Actuated universal joints, or equivalent joint systems, are found in a number of robotic applications, in particular mobile snake robots, continuum robots and robotic tails. These joints have two degrees of freedom on two axes, each perpendicular to a third axis and to themselves. Such joints use a variety of actuation methods, including direct drive motors, linear screw drives, cable based systems, and hydraulics/pneumatics. In this paper the authors design and validate a mechanism that uses the Twisted String Actuator (TSA) in an antagonistic triad to actuate the universal joint, using orientation sensors and load cells to create a robust cascading closed loop control system. This results in a light, compact, high-performance actuation system that avoids the extra mass and hardware complexity that alternative actuation methods present, with the additional challenge of nonlinearity.

I. INTRODUCTION

Actuated Universal Joint (AUJ) mechanisms are found in a wide range of robotic applications, such as confined space inspection using continuum robots [1], highly manoeuvrable mobile snake robots [2], and biomimetic robot tails for stability [3]. Mobile snake robots must usually incorporate electric actuators inline with their joints. This results in an AUJ having to shift the mass of the follower segments and all the actuators inside the follower segments, which results in high torque requirements. Continuum robots and robotic tails can reduce the mass and size of the AUJ by moving their actuators away from the AUJs and use cables to transfer the force to the joints, or use hydraulic or pneumatic actuators which tend to be lighter than equivalent electric motors. This comes at the expense of increased mass and bulk at the base of the arm or tail.

First developed by Würtz *et al.* [4] in 2010, the Twisted String Actuator (TSA) uses two or more strings between two fixtures as a linear actuator. When one fixture is rotated (typically by an electric motor), the looped string twists into a helix, decreasing the distance between them. TSA actuators have been used for a hand orthosis [5], elbow joint [6] and foldable robot arm [7] among other functions.

This work was supported by the UKRI EPSRC DTP, award reference 1957210.

¹Department of Mechanical, Aerospace and Civil Engineering, University of Manchester, United Kingdom.

²Department of Electrical and Electronic Engineering, University of Manchester, United Kingdom.

Email correspondence: damian.crosby@manchester.ac.uk

The primary advantage of TSA over similar linear actuators such as a leadscrew is the reduction (lower velocity, higher torque) the TSA provides is not proportional to the mass of the actuator, in fact it is slightly inversely proportional. Generally, to increase the reduction in an actuator requires the addition of a gearbox which increases mass, but in the case of the TSA, by decreasing the string cross-section radius, the reduction increases given a constant unwound length and motor angle, resulting in a greater reduction with no increase, or even a slight decrease, in actuator mass.

While the reduction in a leadscrew can be increased by decreasing the lead on the thread, which also has no increase in mass, this has a limited range and can quickly run up against manufacturing tolerances or material strength requirements. In order to achieve greater or more robust reductions, the screw radius has to be increased, or the driving motor has to have a larger reduction before driving the screw, both of which usually result in more material (typically steel) and therefore more mass.

However, TSA does have some disadvantages, the most significant of which is a nonlinear reduction equation, which is also dependent on the motor angle (and therefore actuator position). The reduction decreases in a nonlinear fashion as the angle increases, with the derivative decreasing as the angle increases. There is also the compliance of the strings to consider, depending on the thickness and material chosen, which becomes a significant factor under high forces. Both of these issues can be addressed with accurate modelling [8] and/or a robust control strategy, as demonstrated in [4]. What is more of an issue is the unidirectional force of the TSA, which can only impart force in tension. This means that for an AUJ, which is a 2 Degree of Freedom (DOF) joint, a minimum of three TSA are required, unless spring return mechanisms are used, which would impart additional force on the TSA and therefore reduce performance. However, the potential high force to mass ratio of the TSA due to the non-proportional reduction may adequately compensate for the additional actuator requirement.

The focus of this research is to investigate if the TSA is a suitable candidate for control of an AUJ considering both the benefits and drawbacks. To this end, the objective is to simulate a model and then construct a physical experimental prototype to validate the proposed control system.

TABLE I: Model coefficients.

Coefficient	Value	Coefficient	Value
l_1	41.8 mm	J	$1 \times 10^{-6} \text{ kg m}^{-2}$
l_2	0 mm	K_L	1000 N m^{-1}
r	13 mm	f_{\min}	3 N
l_u	41.8 mm	ω_s	441.9 rad s^{-1}
r_s	200 μm	I_s	0.19 A
m	72.619 13 g	K_t	$0.0263 \text{ N m A}^{-1}$
C	0.131 5 N mm	τ_s	4.5 mNm
α_s	$1 \times 10^5 \text{ rad s}^{-2}$		
Coefficient	Value		
I	$\begin{bmatrix} 3 \times 10^{-5} & 0 & 0 \\ 0 & 3.2 \times 10^{-5} & 0 \\ 0 & 0 & 1.4 \times 10^{-5} \end{bmatrix} \text{ kg m}^{-2}$		

A. Twisted String Actuator

Given the unwound length l_u and the cross-section radius of the string r_s , the actuator length is given by

$$l_s(\theta_s) = \sqrt{l_u^2 - \theta_s^2 r_s^2} \quad (1)$$

where θ_s is the motor angle, as shown in figure 1. This equation assumes an infinite string stiffness, so is only reasonably accurate under low tension. Although theoretically the stroke of the TSA can be the entire domain of $[0, l_u]$, in reality the thickness of the string prevents a geometric helix from forming once the helix pitch $q < 4r_s$ (or $q < 2nr_s$ for n strings) as mentioned in [4]. This limits the lower bound of the stroke as follows,

$$l_{\min} = \frac{l_u}{\sqrt{\frac{\pi^2}{2} + 1}} \approx 0.46 l_u \quad (2)$$

or approximately 46% of l_u for a two string TSA.

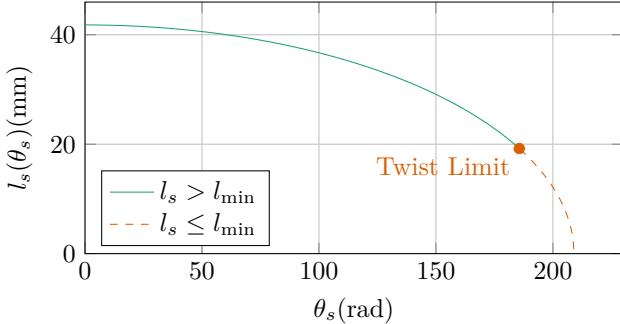


Fig. 1: TSA string length against motor angle with coefficients from table I.

B. Antagonistic Triad

As mentioned in the introduction, because the TSA provides only tensile force, a minimum of three actuators are required for a 2 DOF actuation system. These can be arranged in a triangular configuration to create an “antagonistic triad”, akin to the antagonistic pairs of muscles found in animals. In a pair arrangement, one

actuator contracts while the other relaxes, but in a triad, up to two actuators can share the same action, and one actuator can be inactive if the other two are performing different actions. The geometric structure of the system as shown in figure 2 can be described with two equilateral triangles of inradius r on two planes separated in the z axis. The centroids are then connected via a universal joint from each plane normal to an intersecting point, let the vector $\boldsymbol{\theta} = [\theta_1 \ \theta_2]$ denote the rotation of the second plane relative to the first, in the y and x axes around the intersecting point, and let l_1 and l_2 denote the normal distance from the intersection to the first and second plane centroids respectively. When $\boldsymbol{\theta} = [0 \ 0]$ the triangles are parallel to each other. The distance between the vertex pairs of each triangle is then denoted as $[\lambda_1 \ \lambda_2 \ \lambda_3]$ for the “top”, “left” and “right” vertices of the triangles. When $\boldsymbol{\theta}$ is changed, this will change λ_1 , λ_2 and λ_3 respectively.

To calculate the lengths of the strings for a given $\boldsymbol{\theta}$ of the universal joint, we define a vector function $\Lambda(\boldsymbol{\theta}) = [\lambda_1(\boldsymbol{\theta}) \ \lambda_2(\boldsymbol{\theta}) \ \lambda_3(\boldsymbol{\theta})]$ as follows.

$$\begin{aligned} \lambda_1(\boldsymbol{\theta}) &= \sqrt{(l_1 + l_2 \cos \theta_1 \cos \theta_2 + r \cos \theta_1 \sin \theta_2)^2 \\ &\quad + (r - r \cos \theta_2 + l_2 \sin \theta_2)^2 \\ &\quad + (l_2 \cos \theta_2 \sin \theta_1 + r \sin \theta_1 \sin \theta_2)^2} \\ \lambda_2(\boldsymbol{\theta}) &= \sqrt{(a - b + c)^2 + (l_1 - d)^2 + e^2} \\ \lambda_3(\boldsymbol{\theta}) &= \sqrt{(a + b - c)^2 + (l_1 + d)^2 + e^2} \end{aligned} \quad (3)$$

where:

$$\begin{aligned} a &= -\frac{\sqrt{3}r(\cos \theta_1 - 1)}{2} \\ b &= l_2 \cos \theta_2 \sin \theta_1 \\ c &= \frac{r \sin \theta_1 \sin \theta_2}{2} \\ d &= \frac{\sqrt{3}r \sin \theta_1}{2} + l_2 \cos \theta_1 \cos \theta_2 - \frac{r \cos \theta_1 \sin \theta_2}{2} \\ e &= \frac{r \cos \theta_2}{2} - \frac{r}{2} + l_2 \sin \theta_2 \end{aligned}$$

Surface plots of the functions in (3) are shown in figure 3 for a domain of $[-\frac{\pi}{2}, \frac{\pi}{2}]$.

II. CONTROL SYSTEM

The control system is a four layer cascade design, joining an inverse dynamic control system [9], to the triad force controller in [10], to a proportional controller for each TSA. It uses feedback signals of the joint position from the accelerometers and TSA force from the load cells. A second order setpoint trajectory \mathbf{q} is used as the input, which can either be pre-defined or generated dynamically from user input. Feedback is provided by the AUJ angular position $\boldsymbol{\theta}$ as shown in figure 2, angular velocity $\dot{\boldsymbol{\theta}}$, and TSA tension force \mathbf{f} . Figure 4 shows a complete block diagram of the control system.

A. Actuated Universal Joint Position PID Controller with Acceleration Feedforward

Firstly, a PID controller is used to generate a control signal \mathbf{u} with the input \mathbf{q} as the setpoint, and the AUJ

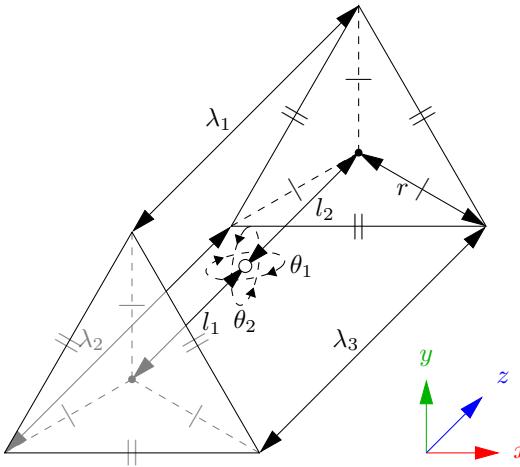


Fig. 2: Kinematic diagram of the antagonistic triad, where the universal joint rotation is defined by $\theta_{1,2}$ on the y and x axes respectively, and the actuator lengths are defined by $\lambda_{1,2,3}$ for the “top”, “left” and “right” actuators. r and $l_{1,2}$ define the anchor points of the actuators.

angular position $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}$ as feedback, plus the addition of a feedforward term for the input acceleration $\ddot{\boldsymbol{q}}$, i.e.

$$\boldsymbol{u} = k_p(\boldsymbol{q} - \boldsymbol{\theta}) + k_i \left(\int_0^t (\boldsymbol{q} - \boldsymbol{\theta}) dt \right) + k_d(\dot{\boldsymbol{q}} - \dot{\boldsymbol{\theta}}) + \ddot{\boldsymbol{q}}. \quad (4)$$

B. Inverse Dynamics

The control signal \boldsymbol{u} from the PID controller is then converted to the desired AUJ torque $\boldsymbol{\tau}$ as follows

$$\boldsymbol{\tau} = D(\boldsymbol{\theta})\boldsymbol{u} + C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + G(\boldsymbol{\theta}). \quad (5)$$

C. Twisted String Actuator Force Optimisation Algorithm

This uses a modified algorithm from [10], which proposes an inverse force transformation algorithm to control an antagonistic triad using force controlled linear actuators, to select an optimal force vector from the desired joint torque. A force matrix \boldsymbol{F} is created from the torque input $\boldsymbol{\tau}$, jacobian J_Λ from the vector function Λ as defined in (3), and minimum force constant f_{\min} . The diagonal components $f_{i,i}$ are equal to f_{\min} , while the other elements in the column are based on a calculation using $J_{\Lambda_{-i,*}}$ where $-i$ is a row removed from the matrix.

$$\begin{aligned} J_\Lambda &= \begin{bmatrix} \frac{\partial \lambda_1}{\partial \theta_1} & \frac{\partial \lambda_2}{\partial \theta_1} & \frac{\partial \lambda_3}{\partial \theta_1} \\ \frac{\partial \lambda_1}{\partial \theta_2} & \frac{\partial \lambda_2}{\partial \theta_2} & \frac{\partial \lambda_3}{\partial \theta_2} \end{bmatrix} \\ \boldsymbol{\gamma}(i) &= -J_{\Lambda_{-i,*}}^{-T} \left(J_{\Lambda_{i,*}}^T f_{\min} + \boldsymbol{\tau} \right) \\ \boldsymbol{F}(\boldsymbol{\tau}, \boldsymbol{\theta}) &= \begin{bmatrix} f_{\min} & \boldsymbol{\gamma}(2)_1 & \boldsymbol{\gamma}(3)_1 \\ \boldsymbol{\gamma}(1)_1 & f_{\min} & \boldsymbol{\gamma}(3)_2 \\ \boldsymbol{\gamma}(1)_2 & \boldsymbol{\gamma}(2)_2 & f_{\min} \end{bmatrix} \end{aligned} \quad (6)$$

The following algorithm then selects one column of \boldsymbol{F} to be the output force vector \boldsymbol{f} , where \top and \perp are boolean *true* and *false* respectively

```

1:  $s \leftarrow [\top \top \top]$ 
2: if  $f_{23} > f_{\min}$  then  $s_2 \leftarrow \perp$  else  $s_3 \leftarrow \perp$  end if
3: if  $f_{31} > f_{\min}$  then  $s_3 \leftarrow \perp$  else  $s_1 \leftarrow \perp$  end if
4: if  $f_{12} \geq f_{\min}$  then  $s_1 \leftarrow \perp$  else  $s_2 \leftarrow \perp$  end if
5: for  $i = 1$  to  $3$  do
6:   if  $s_i \rightarrow \top$  then  $\boldsymbol{f} \leftarrow \boldsymbol{f}_{*,i}$  end if
7: end for

```

D. Twisted String Actuator Force Proportional Controller

The selected forces are then used as an input to a P controller with gain k_{ps} using the measured load cell forces $\hat{\boldsymbol{f}}$ as feedback. The output from this can then be used to control the top, left and right TSA motors, corresponding to the actuators in figure 2.

1) *Simulation Current Control*: In the simulation, each TSA was modelled as a state-space system which takes motor current u as an input and outputs y as the TSA tension force. [4] defines it as such, where J is the motor inertia, C is the motor coulomb friction (modified from viscous friction as the motor only has dry friction), K_t is the motor torque constant, and K_L is the load stiffness. As the original definition is for a fixed load l_u distance from the motor a modified model is required which takes into account the varying length between the motor and load defined by $\Lambda(\boldsymbol{\theta})$. A saturation function, with the compact notation $\text{sat}_x^y z = \max(x, \min(y, z))$ is used to prevent incorrect compression forces when the string is slack. All of the motor coefficients were taken from the Faulhaber 1724TSR datasheet [11] as this is the motor used in the experimental model. An estimated value is used for the load stiffness, this was chosen to be a high number as the model is expected to be very stiff.

$$\begin{aligned} h(\theta_s) &= \frac{\theta r_s^2}{\sqrt{l_u^2 - \theta_s^2 r_s^2}} \\ k(\theta_s, \boldsymbol{\theta}) &= \lambda_n(\boldsymbol{\theta}) - \sqrt{l_u^2 - \theta_s^2 r_s^2} \\ \dot{\boldsymbol{x}} &= \begin{bmatrix} x_2 \\ -\frac{K_t}{J} h(x_1) k(x_1, \boldsymbol{\theta}) - \frac{C}{J} \text{sgn}(x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_t}{J} \end{bmatrix} u \\ y &= K_L \text{sat}_0^\infty k(x_1, \boldsymbol{\theta}) \end{aligned} \quad (7)$$

The state space model was then adapted to include constraints on motor velocity and acceleration set by the motor controller in order to keep the motor within design limits, by replacing $\dot{\boldsymbol{x}}$ with $\dot{\boldsymbol{x}}'$ which contains saturation functions for maximum motor velocity v_s and acceleration α_s .

$$\dot{\boldsymbol{x}}' = \begin{bmatrix} \text{sat}_{-\omega_s}^{\omega_s} \dot{x}_1 \\ \text{sat}_{-\alpha_s}^{\alpha_s} \dot{x}_2 \end{bmatrix} \quad (8)$$

2) *Experimental Velocity Control with Deadband Compensation*: Due to a controller deadband within $\pm 10 \text{ min}^{-1}$, an adjustable deadband compensator is used,

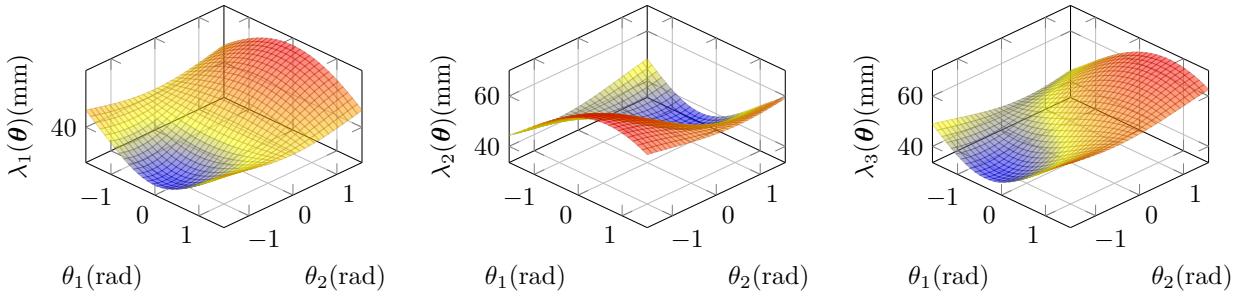


Fig. 3: Surface plots of each element of the vector function $\Lambda(\theta)$, assuming coefficient values from table I. Note that λ_2 and λ_3 are symmetric.

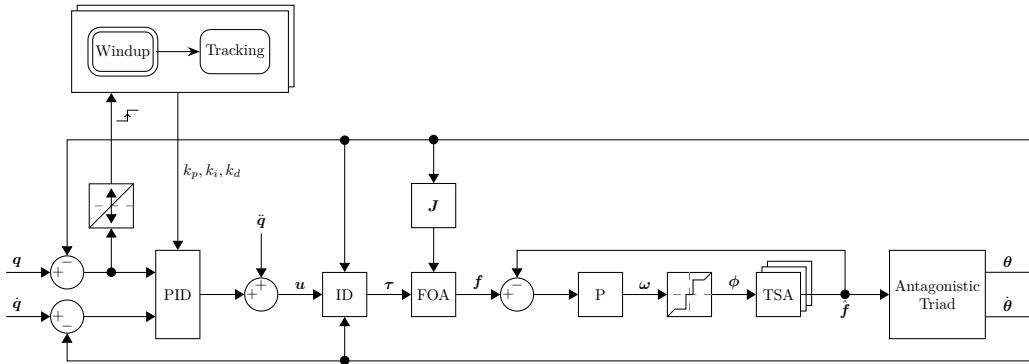


Fig. 4: Block diagram of the complete experimental control system, excluding the hardware velocity controllers for the motors.

TABLE II: PID gains in the simulation and experiment.

Gain	Value	
	Simulation	Experiment*
k_p	800	3×10^4
k_i	3000	350
k_d	50	50
k_{ps}	19	100

* Tracking mode, see section III-B.

$$\phi_i(\omega_i) = \begin{cases} 10 & 10 \leq \omega_i < h \\ -10 & -h < \omega_i \leq -10 \\ 0 & h \leq \omega_i \leq -h \\ \omega_i & \text{otherwise} \end{cases} \quad (9)$$

where ϕ_i is the compensator for the controller i . An adjustment value $h \in [0, 10]$ changes the threshold at which the compensator switches on and off, allowing for a small deadband to remain.

The result from the TSA is then a compressive force acting between each of the three TSA and its corresponding endpoint on the Antagonistic Triad, imparting a torque on the axes of the universal joint.

III. SIMULATION & EXPERIMENTAL RESULTS

A. Experimental Setup

For the experimental validation, a physical prototype of the mechanism was constructed with coefficients from table I as design parameters. This was mounted vertically, in order for the Inertial Measurement Unit (IMU) to measure the orientation of the universal joint. The TSA mechanisms consist of a compact high torque motor attached to the base segment, with a string clamp attached to the motor shaft. On the follower segment, a load cell is mounted on top of a universal joint to ensure a purely axial load, with a capstan bolt attached to the load cell. The string itself is attached to the clamp at both ends using two grub screws for extra security and easy adjustment, and looped through the hole in the capstan bolt. The total mass of the prototype, excluding the mount, is ≈ 176 g. Figure 8 details the construction of the experiment with all the constituent parts.

B. Windup & Tracking States

The tracking controller is activated after a “wind up” stage when the operating conditions are met. The controller parameters for the Windup and Tracking states are $k_p = 800$, $k_i = 3000$, and $k_p = 3 \times 10^4$, $k_i = 350$ respectively. This transition trigger is defined on a per-axis basis, as the first zero crossing of the angle error

(as $\mathbf{q} = 0$ this is effectively $\boldsymbol{\theta}$). A graph showing the difference this state change makes to the AUJ orientation is shown in figure 5.

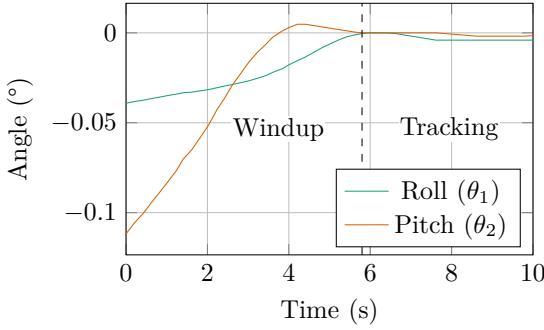


Fig. 5: The initial startup of the mechanism showing the transition between “windup” and “tracking” states.

C. Results

Figure 6 plots the tracking response of both the simulation and experiment. Three trajectories were created to test the capabilities of the mechanism. Two were only in one axis of the universal joint, and the third was in both axes. The deflection angle range was limited to $\pm 11^\circ$ on a single axis, and $\pm 6^\circ$ on both axes. The range can be increased by increasing f_{\min} , which provides, at zero deflection, a longer stroke before the TSA is unwound. A low f_{\min} was chosen to reduce string breakage while tuning the control system.

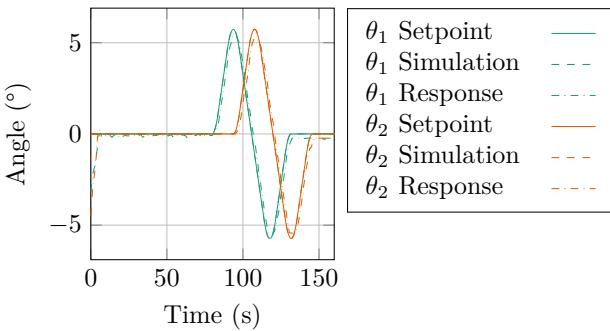


Fig. 6: Plot of the response for a roll θ_1 and pitch θ_2 trajectory. Note the simulation error is very small, so the plot cannot be seen on the graph.

IV. PERFORMANCE COMPARISON

To compare the performance of a TSA AUJ against alternatives, we can measure two metrics, the maximum tension force f_{\max} and maximum stroke velocity \dot{p}_{\max} . This corresponds to the equivalent of maximum torque and maximum velocity in a rotary motor, a larger f_{\max} would be able to actuate a larger follower mass, and a larger \dot{p}_{\max} would be able to rotate the AUJ more quickly. The alternatives chosen for comparison are leadscrews of various rod diameters d_m and pitches λ ,

and a “direct drive” where the motor is rotating the universal joint directly without any reduction or motion transformation.

A. Twisted String Actuator

For the TSA metrics, the equations from [4], in particular $h(\theta)$ and $k(\theta)$ as used for the state space, which can be used to determine f_{\max} and \dot{p}_{\max} . By extracting coefficient r_s as an input to make $f(p, r_s)$ and $\dot{p}(\dot{\theta}, p, r_s)$ the performance of different string thicknesses can be compared for a given unwound length l_u and $\tau_{\max}, \dot{\theta}_{\max}$ over the range of the contraction length p .

$$k^{-1}(p) = \pm \frac{\sqrt{p(2l_u - p)}}{r_s}$$

$$h^{-1}(\theta) = \frac{\sqrt{l_u^2 - r_s^2\theta^2}}{r_s^2\theta} \quad (10)$$

$$f(p) = h^{-1}(k^{-1}(p)) = \pm \frac{\sqrt{(l_u - p)^2}}{r_s \sqrt{p(2l_u - p)}}$$

$$f_{\max} = f(p)\tau_{\max}$$

$$k(\dot{\theta}, \theta) = \frac{\dot{\theta}r_s^2\theta}{\sqrt{l_u^2 - r_s^2\theta^2}}$$

$$\dot{p}(\dot{\theta}, p) = k(\dot{\theta}, k^{-1}(p)) = \pm \frac{\dot{\theta}r_s \sqrt{p(2l_u - p)}}{\sqrt{(l_u - p)^2}} \quad (11)$$

$$\dot{p}_{\max} = \dot{p}(\dot{\theta}_{\max}, p)$$

B. Leadscrew

For the leadscrew metrics, the raising torque calculation [12] can be used as the absolute value of f_{\max} , since the TSA only operates in tension, which can be used to determine the same metrics. The performance of different screw diameters d_m and leads λ can then be compared for a given τ_{\max} and coefficient of friction μ . \dot{p}_{\max} is then calculated by multiplying λ with $\dot{\theta}_{\max}$. The performance of different λ can then be compared for a given $\dot{\theta}_{\max}$.

$$|\tau(f)| = \frac{d_m f (\lambda + \pi d_m \mu)}{2(\pi d_m - \lambda \mu)} \quad (12)$$

$$|f(\tau)| = \frac{2\tau(\pi d_m - \lambda \mu)}{d_m (\lambda + \pi d_m \mu)}$$

$$f_{\max} = |f(\tau_{\max})|$$

$$\dot{p}(\dot{\theta}) = \lambda \frac{\dot{\theta}}{2\pi} \quad (13)$$

$$\dot{p}_{\max} = \dot{p}(\dot{\theta}_{\max})$$

C. Direct Drive

The direct drive metrics are trivially calculated using the lever force and tangential velocity that would be generated at the endpoint of a linear actuator able to impart equivalent angular velocity and torque on the universal joint.

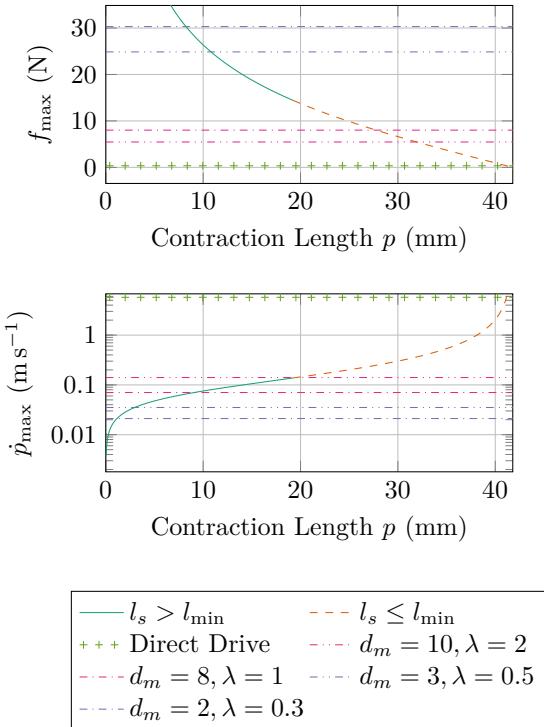


Fig. 7: Performance comparison of the TSA configuration using coefficients from table I to various leadscrew configurations with different d_m and λ , and the direct drive, where $\mu = 0.1$ for the leadscrews.

$$f_{\max} = \frac{\tau_{\max}}{\sqrt{l_2^2 + r^2}} \quad (14)$$

$$\dot{p}_{\max} = \dot{\theta}_{\max} \sqrt{l_2^2 + r^2} \quad (15)$$

D. Comparison

As the values for τ_{\max} and $\dot{\theta}_{\max}$ for the TSA depend on p , but remain constant for the leadscrew, the performance of the TSA is going to be better or worse than a given leadscrew depending on the value p . Figure 7 compares the TSA configuration using the coefficients from table I against a number of common leadscrew configurations that are practical for the dimensions of the AUJ. The TSA outperforms or underperforms different leadscrew configurations depending on p . In simpler terms, the performance of the TSA is dependent on the contraction length. The maximum linear velocity increases with the contraction length, and the maximum tension force decreases with contraction length, both in a nonlinear fashion.

V. CONCLUSION

This research has demonstrated the robust control of the orientation of a universal joint using TSA in an antagonistic triad configuration, with a low tracking error ($\pm 1.8^\circ$) at low speed (max. 0.6°s^{-1}) in 2 DOF control with a vertical base orientation. It has also compared the

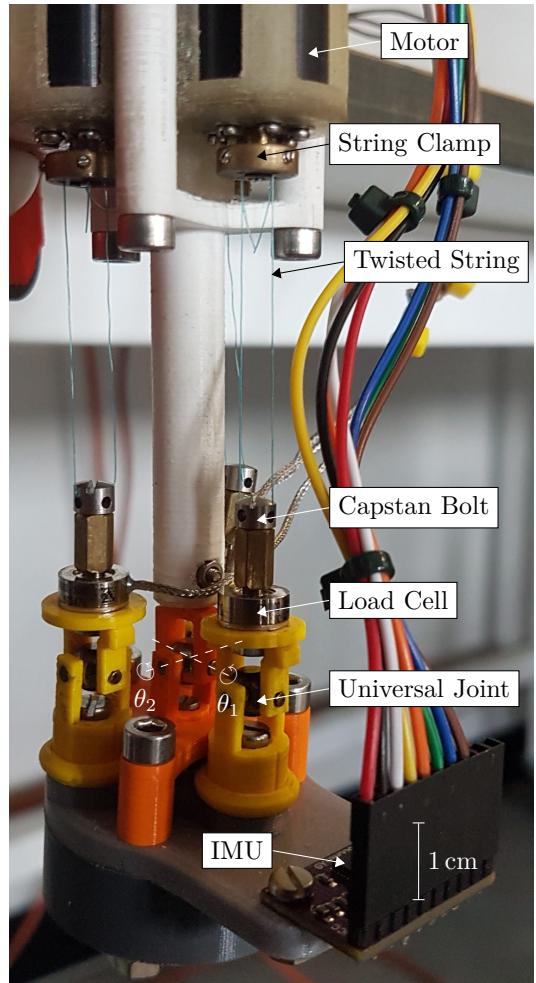


Fig. 8: Annotated photograph of the experimental model, with the roll θ_1 and pitch θ_2 axes marked.

performance of the system to alternative actuation methods. Future developments would include improvements to the orientation sensors, the data from the IMU proved to be unreliable and of poor resolution, so either a superior IMU will be used or an alternative method for sensing the universal joint orientation will be investigated, such as Linear Variable Differential Transformers (LVDTs), hall effect sensors or potentiometers. The system was tested in a vertical base orientation in order for the IMU to be able to measure the joint orientation. With suitable modifications to the inverse dynamics function and a known base orientation, it will be possible to test the controller at non-vertical base orientations. We will also increase the joint velocity, joint range and follower mass (m) in order to test the performance of the system under more strenuous conditions. The controller performance will also be quantified, examining the open and closed loop bandwidth. Eventually, we wish to develop a system comprised of multiple segments, to demonstrate its suitability for applications such as mobile snake robots or continuum robots.

REFERENCES

- [1] R. Buckingham and A. Graham, "Nuclear snake-arm robots," *Industrial Robot: An International Journal*, 2012.
- [2] M. Luo, R. Yan, Z. Wan, *et al.*, "Orisnake: Design, fabrication, and experimental analysis of a 3-d origami snake robot," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1993–1999, 2018.
- [3] W. S. Rone, W. Saab, and P. Ben-Tzvi, "Design, Modeling, and Integration of a Flexible Universal Spatial Robotic Tail," *Journal of Mechanisms and Robotics*, vol. 10, no. 4, Apr. 2018, 041001, ISSN: 1942-4302. DOI: 10.1115/1.4039500.
- [4] T. Würtz, C. May, B. Holz, C. Natale, G. Palli, and C. Melchiorri, "The twisted string actuation system: Modeling and control," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2010, pp. 1215–1220. DOI: 10.1109/AIM.2010.5695720.
- [5] P. Muehlbauer, M. Schimbera, K. Stewart, and P. P. Pott, "Twisted string actuation for an active modular hand orthosis," in *ACTUATOR; International Conference and Exhibition on New Actuator Systems and Applications 2021*, VDE, 2021, pp. 1–4.
- [6] J. Park, J.-i. Park, H.-T. Seo, Y. Liu, K.-S. Kim, and S. Kim, "Control of tendon-driven (twisted-string actuator) robotic joint with adaptive variable-radius pulley," in *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, IEEE, 2020, pp. 1096–1098.
- [7] B. Suthar and S. Jung, "Design and feasibility analysis of a foldable robot arm for drones using a twisted string actuator: Frad-tsa," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5769–5775, 2021. DOI: 10.1109/LRA.2021.3084890.
- [8] S. Nedelchev, I. Gaponov, and J. Ryu, "Accurate dynamic modeling of twisted string actuators accounting for string compliance and friction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3438–3443, 2020. DOI: 10.1109/LRA.2020.2970651.
- [9] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, 2020.
- [10] F. Dessen, "Coordinating control of a two degrees of freedom universal joint structure driven by three servos," in *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, vol. 3, 1986, pp. 817–822. DOI: 10.1109/ROBOT.1986.1087559.
- [11] *Dc micromotors - precious metal commutation*, Dr. Fritz Fauhaber GmbH & Co. KG, Feb. 2020.
- [12] J. Shigley, C. Mischke, and R. Budynas, *Mechanical Engineering Design*, ser. McGraw-Hill series in mechanical engineering. McGraw-Hill, 2004, ISBN: 9780072520361.