Parallel Platform-Based Robot for Operation in Active Water Pipes

Amr Bekhit

Submitted in accordance with the requirements for the degree of
Doctor of Philosophy

The University of Leeds
School of Mechanical Engineering

September, 2015

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

The publication that is referenced in the thesis is *"Kinematic Analysis and Locomotion Strategy of a Pipe Inspection Robot Concept for Operation in Active Pipelines"*, authored by Amr Bekhit, Abbas Dehghani and Robert Richardson, published in the International Journal of Mechanical Engineering and Mechatronics (IJMEM), Volume 1, Issue 2 – Year 2012 – Pages 15-27. DOI 10.11159/ijmem.2012.003.

The work published in this paper is included in the thesis as follows:

- Chapters 1 and 2 expand upon sections 1 and 2 in the paper (Introduction and Assessment of Existing Pipe Inspection Robots).
- Chapter 3 includes the robot requirements that are mentioned in section 1 in the paper.
- Chapter 5 expands upon the robot design presented in sections 3 and 3.1 in the paper.
- Chapter 6 includes the Inverse Kinematics calculations presented in section 3.2 in the paper. It also expands upon section 3.3 in the paper regarding static force analysis.

The paper's primary author was responsible for all of the content in the paper. The 2 co-authors of the paper are the project supervisors and were responsible for reviewing the paper.

Acknowledgements

Those who have undertaken the journey of a PhD will no doubt agree that it can be a difficult and challenging endeavour and is one that cannot be completed alone without the continuous support and encouragement of those around you.

For me, the greatest and most important source of support has been my family: my parents Dr Rowaa El-Magazy and Dr Mohammad Bekhit, and my siblings Dr Aladeen and Mariam Bekhit. It is truly a blessing to be surrounded by a loving and supportive family that never ceases to believe in you even when you start to doubt yourself. They have firmly stood by me during the entirety of this long and arduous seven-year journey and I would have no doubt failed long ago had they not been there to shore me up.

My sincere thanks goes out to my supervisors, Dr Abbas Dehghani and Dr Robert Richardson for their continuous support, advice, encouragement and patience throughout the entire PhD. My sincere thanks to you both.

A special thanks goes to my employers, MechTronic Ltd and especially to Trevor Poulter and Simon Haigh. Despite the pressures of work, you have afforded me a flexibility and freedom regarding the PhD that I'm sure very few employers would have offered, and I am very grateful for that.

My thanks goes to my friends and colleagues throughout the years who have helped in one way or the other, through technical advice, brainstorming, proof-reading and by being part of the fond memories I will retain of this journey: Dr Mohammed Awad, Dr Alfonso Montellano Lopez, Dr Babak Khazravi and Mohamed Eltuhamy. Very special thanks to Dr Mohammed Awad for his continuous support and advice.

Abstract

This thesis presents a novel design for a pipe inspection robot. The main aim of the design has been to allow the robot to operate in a water pipe while it is still in service.

Water pipes form a very crucial part of the infrastructure of the world we live in today. Despite their importance, water leakage is a major problem suffered by water companies worldwide, costing them billions of dollars every year. There are a wide variety of different techniques used for leak detection and localisation, but no one method is capable of accurately pinpointing the leak location and severity in all pipe conditions with minimal labour. A survey of existing pipe inspection robots showed that there have been many designs implemented that are capable of navigating the pipeline environment. However, none of these were capable of fully autonomous control in a live water pipe. It was concluded that an autonomous pipe inspection robot capable of working in active pipelines would be of great industrial benefit as it would be able to carry a wide range of sensors directly to the source of the leak with minimal, if any, human intervention.

An inchworm robot prototype was constructed based on a Gough-Stewart parallel platform. The robot's inverse kinematics equations were derived and a simulation model of the robot was constructed. These were verified using a motion capture suite, confirming that they are valid representations of the robot. The simulation was used to determine the robot's movement limitations and minimum bend radius it could navigate. Several CFD simulations were carried out in order to estimate the maximum fluid force exerted on the robot. It was found that the robot's design successfully minimised the fluid force such that off-the-shelf actuators had the capability to overcome it. The prototype was successfully tested in both a straight and bent pipe, demonstrating its ability to navigate a dry pipe environment.

Overall, the robot prototype served as a successful proof of concept for a design of pipe inspection robot that would be capable of operating in active pipelines.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
ADC	Analogue-to-Digital Converter
AI	Artificial Intelligence
AUV	Autonomous Underwater Vehicle
CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
CIPP	Cured-in-Place Piping
DMA	District Metering Area
DOF	Degrees of Freedom
ECF	Electro-conjugate Fluid
FPC	Flexible Printed Circuit
GCON	General Constraint
GS	Gough-Stewart
GUI	Graphical User Interface
IK	Inverse Kinematics
PIG	Pipeline Inspection Gauge
SMA	Shape-Memory Alloy

List of Publications

Part of the research carried out for this thesis has been presented in international conferences, industrial exhibitions and published in the following papers:

- Bekhit A., Dehghani A., Richardson R., “A Pipe Inspection Robot for Operation in Active Pipelines” in Proceedings of the International Conference on Mechanical Engineering and Mechatronics, Ottawa, Ontario, Canada, 16-18 August 2012, Paper No. 212.
- Bekhit A., Dehghani A., Richardson R., “Kinematic Analysis and Locomotion Strategy of a Pipe Inspection Robot Concept for Operation in Active Pipelines” in the International Journal of Mechanical Engineering and Mechatronics, Volume 1, Issue 1, Year 2012, p15-27.
- A poster presentation along with a live demonstration of the robot was given at the Yorkshire Water Production Technology Day on 25th-26th September 2014. This is an industrial exhibition for the water distribution industry organised by Yorkshire Water.
- Videos of the robot can be found at the following URLs:
    - https://www.youtube.com/watch?v=N4zdWACxfF8
    - https://www.youtube.com/watch?v=CcVgF740vGk
    - https://www.youtube.com/watch?v=hBIhYeuluFs

Chapter 1
Introduction

1.1 Background

Pipelines form a crucial part of the way we live today as they are the primary means of transporting many of the resources that we depend on, such as water, gas and oil. There are millions of miles of pipeline throughout the world and they can be found in both domestic and industrial settings. Many of these pipelines are situated underground so as not to affect everyday life. However, this makes repair and inspection of the pipelines a costly and difficult task, especially since many pipelines have not been designed to optimise automatic repair and inspection tasks [1]. Also, because pipelines rarely have any redundancy, the use of conventional inspection and repair methods, which require the flow to be shut off, can often lead to disruption [2]. Without a reliable method of determining the exact location of a problem, large sections of pipeline need to be excavated and this can be made even more difficult if the pipelines are situated in an urban environment, where the repair work can negatively impact daily life. Neglecting the repair work of such pipelines can lead to leakage of valuable product into the environment, which can lead to damage of the surroundings, expensive clean-up work as well as a loss of business reputation [1], [3]–[5].

1.1.1 The Scale of the Problem

One of the most common and crucial uses of pipelines is in the transportation of clean water to homes and businesses. In order to appreciate the ubiquity and importance of water pipes and the huge maintenance challenge faced by water companies, statistics published by the five largest water companies in the UK and their government regulator, Ofwat, were examined.

1.1.1.1 Water Mains in the UK

The annual reports of the five largest water companies in the UK contain information on the number of customers they serve and the length of water mains that they maintain. This information is summarised in Table 1.1.

Table 1.1 A list of the 5 largest water companies in the UK along with the number of customers they serve and the length of water mains they maintain.

Company	Number of customers (millions)	Length of water mains (1000's km)	Sources
Thames Water	9	31	[6] pages 2 and 12
Severn Trent	7.7	46	[7]
United Utilities	3.2	43	[8] pages 2 and 22
Yorkshire Water	5	31	[9] pages 5 and 28
Anglian Water	4.5	38	[10] pages 2 and 49
Totals	**29.4**	**189**	

Using the ratio of total customers to total water mains shown in the above data, an estimate for the total length of water mains in the UK was calculated. With the UK population estimated at 64.1m people in 2013 [11], the total length of water mains in the UK can be estimated at 412,000km. When this figure is compared to the total road length in the UK of 395,000km [12], it becomes evident that water mains alone are a significant part of the infrastructure, with large amounts of pipeline that need to be maintained in operational order. When it is also noted that this estimate only covered water mains and did not include the sewerage system or other pipelines such as those that deliver gas, the scale of pipelines in our lives becomes even more apparent.

1.1.1.2 Water Leakage

Loss of water due to leaks and failures is inevitable in such a large and extensive pipe network. The scale of the problem becomes clear, however, when it is noted that the estimated leakage levels of water companies in England and Wales in 2009-10 was 3.3 billion litres of water every day [13]. With the average person in the UK consuming 150 litres daily [14], this is enough to supply an extra 22 million people with water on a daily basis.

To put a financial perspective on this, Yorkshire Water reports that in 2014, the operating costs of clean water production were £162m [15]. With the amount of water lost due to leakage representing 25% of the total water delivered by Yorkshire Water [16], this is, almost literally, an estimated

£40.5m that is lost down the drain on an annual basis. Financial losses on a similar scale can be expected for other water companies. Reducing such inefficiencies would have a direct effect on both the profitability of the water companies and the water bills of their customers.

It is clear that there is both a strong financial incentive to minimise leakage and also a moral obligation to prevent such huge amounts of clean water being wasted when 780m people around the globe are still without access to an improved drinking water source [17].

1.1.2 Pipe Inspection Robots

With the scale of the problem so vast, and the inherently difficult nature of underground pipeline maintenance, it is inevitable that the flexibility and intelligence of robotic solutions would be explored in order to provide controlled and targeted inspections and repairs. Indeed, the development of pipe robots has been ongoing for many years, with pipe inspection robots being developed as early as 1986 [18].

Over the years, many attempts have been made to develop robotic systems that can be used to inspect the condition of pipelines and potentially carry out repairs. However, the complexity of pipe networks, restricted space inside pipes and the various elements such as bends, junctions and other obstacles put limitations on the size and shape of such robots [19], [20].

Although there have been several robots designed for the purpose of pipe inspection and repair, most of these focus on operation in empty pipes. The problem with such robots is that they do not take into account the effects of pressurised fluid on the motion and stability of the robot. Although a few robots have been developed for operation inside active gas pipelines, operation in active fluid pipelines has garnered very little attention. The development of a robot capable of operating in such conditions would be of great industrial benefit as the robot would be able to operate inside the pipeline while it is still active and allow the pipeline to continue working unaffected.

1.2 Motivation

As water leakage is such a significant and costly problem, and with pipe networks being so vast, it would be of great benefit if a robotic system could be developed that would allow almost continuously unattended inspections to be carried out on the whole pipe network, and even some automatic repairs.

Ultimately, a robot would be developed that could “live” inside underground water pipes on an almost permanent basis. A swarm of these robots would be installed in pipelines throughout the country, automatically patrolling their relevant sections and performing continuous inspection of the condition of the pipe, feeding that information back to the water company. This would give the water company up to date information about the condition of all its pipes. Such continuous inspection of the pipelines would allow the robots to detect when problems are about to happen and can possibly perform any preventative maintenance automatically, thus minimising service disruptions. If the repair needs the flow of water to be shut off, it would be possible to remotely shut off and reroute the flow of water, then instruct the robot to carry out the repair, resuming the flow once the repair is complete, thus reducing the need to excavate the pipe. In situations where the repair is too severe for the robots to handle, the rich set of information that the robots could send back regarding the location and nature of the problem would allow the repair team to head directly to the source of the problem armed with plenty of information about the situation, potentially minimising the time, cost and effort spent on the repair itself.

Such a robot would not necessarily have to be confined to operation in water pipes. The same technology could also be used in municipal gas pipes, oil pipelines and pipe networks present in factories and plants.

1.3 Aims and Objectives

The development of such a robot would cover a very wide range of different fields and skills. Due to time and resource constraints, this thesis will focus on the design and validation of the robot’s mechanical structure in a dry environment.

1.3.1 Aims

The aim of the research project is to design and develop the mechanical platform for a robot that can operate inside water pipes while they are still active.

1.3.2 Objectives

In order to fulfil the above research aim, the following objectives need to be met:

- Review the existing literature on the pipe environment and how water companies manage it.

- Investigate the various locomotion methods mobile and climbing robots can use, and how existing pipe inspection robots have used them. Identify the most suitable locomotion to use in this application.
- Design and construct a robot based on the chosen locomotion method.
- Analyse the kinematics of the design and the forces it is exposed to in order to assess its feasibility and understand its limitations.
- Verify that the robot prototype conforms to the theoretical understanding.
- Test the robot's locomotion ability in various pipes to validate its performance and ability to navigate the pipeline.

1.4 Contributions of the Thesis

The contributions of the thesis are summarised below:

1. Development of a pipe inspection robot with specific emphasis on the robot's ability to operate inside active pipelines with high flow.
2. The use of a Gough-Stewart parallel structure with non-fixed top and bottom platforms to develop an inchworm-based robot for internal pipe climbing.
3. A methodology for estimating the climbing performance of an internal pipe climbing robot in a bent pipe, which can be used to determine the minimum bend radius.

1.5 Thesis Organisation

The thesis is divided into 8 chapters. The outline of each chapter is presented below:

Chapter 1 introduces the background and motivation for the project and presents the aims and objectives for the development of a pipe inspection robot for operation in active pipelines. The chapter also presents the scope and contributions of this research

Chapter 2 is split into two parts. In the first part, the state of water pipes is reviewed in order to understand the robot's working environment. The methods that water companies use in order to detect, manage and repair leaks is reviewed in order to understand the tasks that the robot would need to carry out.

In the second part, a review of the various locomotion methods used by mobile robots is presented with specific emphasis on their application in pipe

inspection robots. The various robots developed are categorised by their locomotion method and their advantages and disadvantages discussed.

Chapter 3 presents the requirements and specification for the robot based on the information gathered in the literature review. The methodology used to develop the robot is also presented.

Chapter 4 presents the conceptual design process for the robot. The findings of the literature review are used to decide on the most suitable locomotion method and a specification for the ideal pipe robot is presented. Various design ideas are then presented and assessed in order to determine their suitability. Based on this assessment, a single concept is deemed to be the most appropriate and is chosen for further development.

Chapter 5 presents the detailed design implementation of the robot prototype. This involves the selection of the linear actuators, the design of the robot's support discs that provide it with grip, and the design of the various connecting components that form the rest of the robot's structure.

A brief overview of the design of the robot's control hardware and software is also presented in this chapter.

Chapter 6 covers the theoretical and analytical work performed on the robot's design. The robot's IK equations are derived and a simulation model constructed. This is used to understand the robot's movement limitations, especially regarding the sharpest bend that the robot can navigate. CFD simulation is used in order to estimate the fluid force exerted on the robot and understand how this force will be transferred to the robot's actuators.

Chapter 7 focuses on the experimental work carried out. This covers two main parts: the first focuses on validating the IK equations and simulation model derived in the previous chapter by using a motion capture suite. The second half demonstrates the robot's ability to navigate straight and bent pipes.

Chapter 8 summarises the work done in the thesis, assess how well the aims and objectives have been met and discusses recommendations for future work.

Chapter 2
Literature Review

2.1 Introduction

In order to design a robotic system that will work in the inspection and repair of underground water pipes, it is important to understand the pipes, how they are used, what problems they pose to water companies and how they deal with them. It is also important to review existing work on the subject, to understand how other researches have tackled the problem, and what technologies exist that can be used to develop a solution.

In this chapter, the state of water pipes and how water companies manage and repair them is examined. After that, a review of the various locomotion methods used by mobile robots is carried out, with specific emphasis on how these locomotion methods have been used to develop internal pipe inspection robots. As a result of these reviews, the gaps in knowledge are identified and conclusions are drawn that are used to direct the robot's development.

2.2 Water Pipes

The water pipes themselves are responsible for transporting clean water from the treatment plants all the way to the end consumers. Most water pipes are located underground, generally between 1-3m below the surface. Their sizes vary from 75mm for domestic pipes all the way to greater than 450mm in diameter for main trunk pipes. According to information from Balfour Beatty, the water flowing through them can be pressurised from 2-8bars, with maximum velocities of approximately 0.8m/s.

Pipes are typically made from concrete, cast iron, and more recently, plastics, such as polyethylene (PE) and polyvinylchloride (PVC). In the UK, despite upgrades to plastic piping, many existing water mains are made from iron and lead, with some dating back to the Victorian era [21]. Depending on the pipe material, the pipe may have an external and internal coating for protection. One such coating used for a stainless steel water pipe involves bitumen for the inner coating and an outer lining of bitumen/fibre-glass [22].

The pipe networks themselves primarily consist of straight sections, with 90° swept bends and tees and vertical sections [21]–[24]. Although there was no information found on the frequency of such features in a pipe network, it is

reasonable to assume that all of these features are commonly found. Straight sections will likely occupy the bulk of a pipeline. Whenever a pipeline needs to avoid obstacles underground or change direction, swept bends would be used. Any branch pipes that need to connect to the main trunk pipe would do so via tee-junctions. Although most pipelines are expected to be horizontal, changes in ground elevation or certain underground obstacles would require the pipes to also slope vertically. Figure 2.1 shows several examples of underground pipeline maps. As can be seen from the figure, straight sections, swept bends and tees are very commonly found.

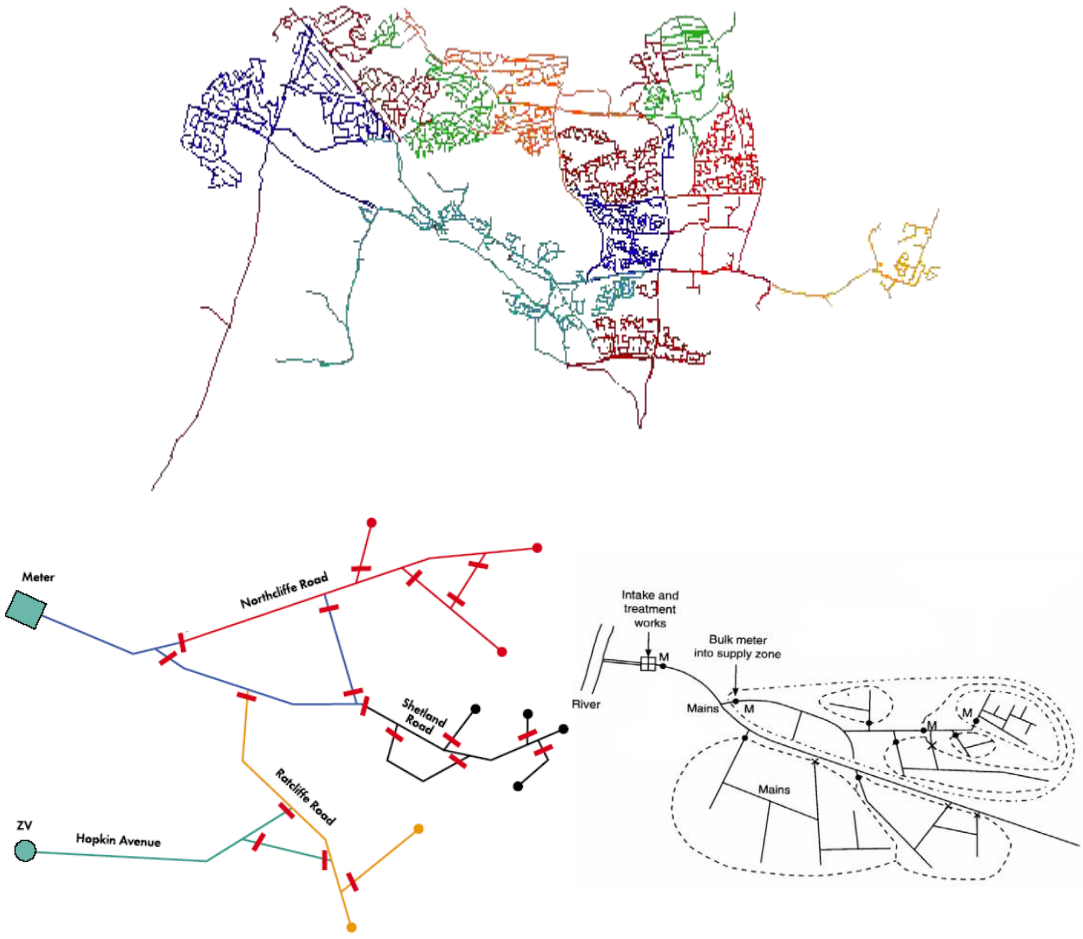


Figure 2.1 Several maps of various underground water pipeline networks [25].

By examining the maps, it becomes apparent that straight sections, tees and bends are common features in an underground water pipe network.

Despite care being taken to make sure that pipes are manufactured and installed correctly and competently, inevitably, leaks will occur. These are caused by a wide variety of different reasons, of which are the following [21], [22], [26]–[30]:

- Corrosion due to the chemical interaction between metal pipes and the soil, or due to the presence of sulphate-reducing and acid producing bacteria.
- Ground movement, which varies from small and gradual shifts (e.g. due to changes in soil temperature and moisture levels and ground settling), to large sudden movements (e.g. from earthquakes).
- The use of high supply pressures, which can exceed the limits of older piping.
- Erosion from existing leaks.
- Damage from third party construction, maintenance and excavation work.
- Inadequate design and construction.
- Ageing infrastructure and poor maintenance.

It is estimated that worldwide, 32 billion cubic meters of water is lost due to leakage in the distribution networks, which is estimated to cost more than $14billion per year [29], [31]. The amount of water lost is typically between 20-30% of the total amount produced, with that proportion rising to 50% in some cases [27].

Water loss from pipes can be classed into two main types [29]:

- Bursts. These are medium to high flow and generally last for a short time because they are quickly reported.
- Leaks. These are low flow incidents and often last for a long time as they are hard to detect. Very low flow leaks are even harder to identify and are considered to be part of the inevitable background leakage.

The negative impact of leaks and bursts is wide-ranging, from the companies that supply the water and maintain the pipes, to the consumers, and even to the environment. Some of the negative impacts that leaks and bursts can have are as follows [21], [24], [27], [29]:

- Loss of profit. The clean water that is lost due to leakage and bursts has been transported from water sources, purified and processed and then pumped to the end consumers. all of which costs the water companies money. The loss of that water is an unnecessary financial burden that increases the cost of the water that reaches consumers.
- Strain on company resources. Having to send repair teams out to repair bursts and leaks costs money that could be invested in other parts of the company. Similarly, the water lost due leaks and bursts means that for the amount of water consumed by users, much more

than that needs to be transported and processed. This puts more strain on the companies' facilities than is actually needed and may lead the company to unnecessarily spend money on upgrading the capacity of their plants and on finding more water sources.

- Liability for damage. Small scale leaks that are left unrepaired for a long period of time can cause significant damage to the surrounding area, such as roads and footpaths and can cause them to collapse in some cases, making the companies liable for their repair.
- Poor energy usage. The transportation and processing of more water than is needed results in excessive energy usage and carbon emissions which are harmful to the environment.
- In areas where water sources are scarce, leaks and bursts squander a rare and valuable environmental resource. In the US, more than 36 states are expected to experience water shortages over the coming years. Similarly in China, 400-600 medium and large cities experience water shortages.
- Poor public image. Consumers view water as a staple commodity, so losing it due to a pipe burst or service interruption due to a repair can have a negative effect on the public image of the water company.

Finding and repairing leaks, as well as preventing them from happening in the first place is therefore of paramount importance to water companies.

2.3 Leakage Management

Water companies can take two basic approaches in dealing with water loss due to leakage and bursts [32]:

- Passive approach. The company only reacts to customer complaints and leak reports. This is simple and cheap, but only works well if the system leak level is low (below 10-15%). Even then, it is still recommended that some sort of leak monitoring reporting system is put in place, otherwise leak levels can rise to 40%.
- Active approach. In this case, the company actively monitors and assesses the network to identify problems before they happen or to reduce the time between a problem appearing and the repair.

Identifying and locating undetected leaks requires a multi-tiered approach that first locates the presence of leaks very broadly, with subsequent localisation techniques honing in on the exact location of the leak.

2.3.1 Broad Leak Assessment

The first step in assessing the state of the pipe network is to perform a broad estimation of the leak levels in the network and the areas that suffer the most. This is generally done using the following methods [32]:

- Step-Testing. In this approach, the network is split into discrete areas that can have their inlets shut off, known as District Metering Areas (DMAs). These typically contain between 500-3000 properties and households. In order to assess the leakage levels for each DMA, the water company will send a technical team out to install flow meters at the inlets and outlets of each one. The flow will then be shut off to each DMA and the drop in the network's total flow rate is monitored. A larger than expected drop indicates a leak in that section. This exercise is typically done at night during the period of minimum usage so that water taken out of the system by consumers does not affect the measurement. Step testing is considered to be very effective at localising leaks and finding which areas suffer the worst. It does, however, require detailed maps of the pipe network and the location of its valves. It requires consumption to be stable during the procedure, which means that in areas with abnormal night usage (such as 24 hour takeaways and all-night licensed venues) errors are introduced into the measurement. It can be very labour intensive if non-telemetry enabled meters are used [21], [32], [33].
- Top down approach. This method involves comparing the total amount of water produced and the total amount of water consumed (via property meters) to assess the level of leakage. A large difference in the amounts produced and consumed indicates severe leaks. This method is considered very coarse and does not give a very accurate representation of the network, due to the fact that there are very few areas were 100% of customer properties are metered [21]. Despite this, it is still useful for verifying the results of step-testing.
- Acoustic Logging. This method involves measuring the distinctive sound generated by leaks that is transmitted through the water or through the pipe wall. Acoustic loggers are installed by attaching them to pipe fittings and are programmed to record and analyse the leak noise at night where background noise is likely at its minimum. The data recorded by the loggers can then be manually retrieved or automatically transmitted to a central server. The data from the

loggers can then be compared to identify which loggers are detecting leaks and the relative amplitude between loggers can be compared to estimate the location of the leak [32], [25].

2.3.2 Leak Localisation and Pinpointing

After assessing the state of the pipe network and its various districts, work can progress on to honing in on the individual leaks in order to exactly identify their locations. There are a wide variety of different techniques that are in use in order to identify the location of individual leaks. These are listed below [32], [25]:

- Transient Wave Analysis. There are a wide variety of methods that rely on measuring pressure transient waves inside a pipe and analysing how these waves reflect off fractures and defects in the pipe in order to determine the presence of leaks, such as the Leak Reflection Method, Inverse Transient Analysis, the Frequency Response Method and more. However, although these methods can demonstrate success in a lab environment or in a very simple pipe network, they have very limited use in complex pipe networks as the various other features in the pipe (such as junctions, welds, valves etc.) also affect the transient waves and can make it difficult to differentiate between a fracture and feature in the pipe [32].
- Ground Penetrating Radar. This method is used to detect the underground voids caused by water that has leaked out of a pipe and can be carried on a vehicle that then drives on the road above the pipe, mapping its features. The main limitations of this method are its limited ground penetration capabilities (up to 2m) and the return of false positives from other metal objects in the ground. In cold climates where water pipes are buried deep underground to avoid freezing, this method cannot be relied upon to give accurate results. It is still considered a good method to use when large diameter, non-metallic pipes are to be examined.
- Tracer Gas Technique. This method involves injecting a non-soluble, non-toxic gas (95% nitrogen + hydrogen/helium) into the pipe and then using gas sensors to scan over the suspected area and detect the leak. Although this method is effective at detecting multiple leaks at the same time, it is very expensive and is considered a specialist process [32], [25].
- Direct listening. This involves directly listening to the sound of leaks via a human operator to determine the presence and location of

leaks. This can range from relatively simple techniques that use a listening stick (similar to a stethoscope) that is placed on a pipe fitting and then listened to by an operator, to electronic microphones with filters and amplifiers. Ground microphones are also available that are placed on the surface above a pipe to measure the amplitude of the leak sound. An operator moves the microphone along the surface, following the pipe, and notes the change in leak noise amplitude, which is used to pinpoint the location of the leak. Such methods are very labour intensive and rely on the skill and experience of the operator [32], [25].

- Leak noise correlators. These involve two microphones placed a certain distance apart on a pipe. The leak noise reaches each microphone at a different time depending on the distance to it, and by comparing the signal from the two microphones, the location of the leak can be determined. Acoustic methods in general are affected by the pipe material (with metal pipes giving better results than plastic), and the pipe size (with large pipes attenuating the signal more than smaller ones). Leak signals that are very quiet are hard to distinguish from background noise, and large pipes can shift the frequency of the noise lower, making it harder again to distinguish them from background noise.
- PIG-mounted techniques. PIGs are devices that are inserted into a an active pipe and rely on the flow to propel them along. The same instruments and sensors mentioned above can be mounted on to pipe PIGs and they can take measurements in-situ, allowing them to get very close to the features that are being measured. For example, the SmartBall device contains an acoustic logger which measures the sound level as the device moves along the pipe, correlating that with the distance travelled, allowing the precise location of leaks and other features to be determined [34]. However the reliance of PIG devices on the flow of water for propulsion means that they have very little control over their location and movement. This is discussed further in section 2.5.5.1.

2.3.3 Repair

Once the leak location has been identified, work can begin on repairing the fault. There are two approaches that companies can use to repair a pipe: Open-cut and trenchless methods.

2.3.3.1 Open-Cut Repair Methods

With open-cut methods, the ground around the entire length of the pipe is excavated to gain access. This excavation is referred to as a “trench”. The pipe can then be repaired by either replacing it completely, or by using a wrap-around bandage around the outside of the pipe [25], as shown below in Figure 2.2. This is considered the standard method for construction and repair. Some companies prefer not to replace the entire pipe in order to avoid the potential health risks associated with leaving the pipe open.

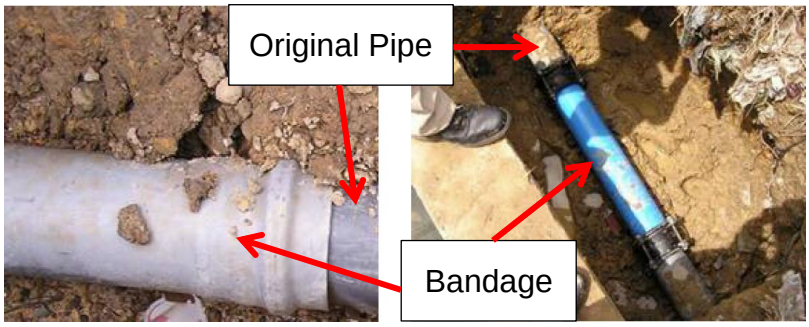


Figure 2.2 Pipes repaired using the "wrap-around bandage" method [25].

2.3.3.2 Trenchless Repair Methods

The other methods that can be used to repair pipes fall under the category of trenchless technologies. These methods are able to perform pipe repairs while only excavating a few small trenches, or even without any excavation at all. This minimises cost by reducing or eliminating excavations and thus reducing the clean-up and restoration process after a pipe has been repaired. They also help to minimise the disruption caused by the excavation of a large trench.

There are a wide variety of trenchless technologies, a summary of which can be found below [35]–[37]:

2.3.3.2.1 Pipe Replacement Methods

These methods involve removing or destroying the existing pipe and at the same time inserting a replacement. There are a number of methods of accomplishing this:

Pipe bursting/splitting is a method where a pilot head (typically a conical shaped tool) is forcibly pulled through the existing pipe, as shown in Figure 2.3. The head breaks up the existing pipe and pushes the fragments outwards, while at the same time, the new pipe is pulled through. The soil type and depth from the surface are important, as they determine whether

significant surface ground movements will be caused by the procedure. Similarly, the effect on adjacent pipelines and utilities needs to be taken into account. There are a wide variety of different pilot heads depending on the type of pipe to be cut. For pipes which are made from brittle materials (e.g. clay, cast iron, concrete), pneumatic, hydraulic or static bursting heads can be used. These heads use pneumatic of hydraulic power to expand the head, whereas the static head simply relies on the force pulling the head through. For ductile materials, a pipe splitting head is used instead, where the head contains a powered cutter that can cut through the pipe material.

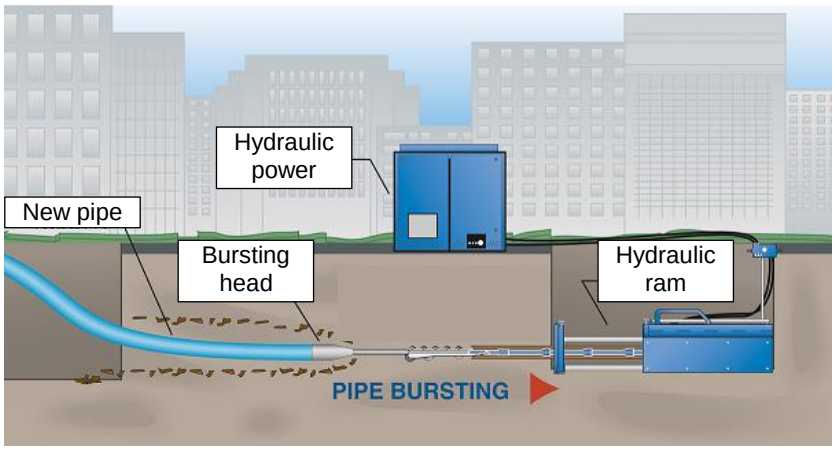


Figure 2.3 Pipe Bursting

In pipe bursting, a hydraulic ram pulls the pipe bursting head through the existing pipe. This forces the pipe to expand, which breaks it up. At the same time, the new pipe is pulled into the space that the old pipe occupied [38].

A similar technique to pipe bursting is pipe reaming/eating. Instead of splitting and moving the old pipe outwards, the pilot head actually bores out the space occupied by the existing pipe, sending the pipe fragments back to the surface to be disposed of. The advantage of this method is that the old pipe is completely removed and replaced with the new one.

Another similar technique is pipe ejection. This method involves digging an entrance and exit trench and then pushing the existing pipe out using the new pipe. As the pipe is pushed out, it is broken up and removed from the exit trench. This method can only be used if the existing pipe has enough structural strength to withstand the pushing force.

2.3.3.2.2 Pipe Renovation Methods

These methods involve repairing the existing pipe without replacing it. Some of these methods are summarised below:

Slip-lining is considered one of the simplest trenchless techniques. With this method, a new pipe of a smaller diameter is pulled through the existing pipe, so that it sits inside it. The gap between the inner and outer pipes is then grouted to mechanically link the two pipes and allow them to behave as a composite and improve structural strength.

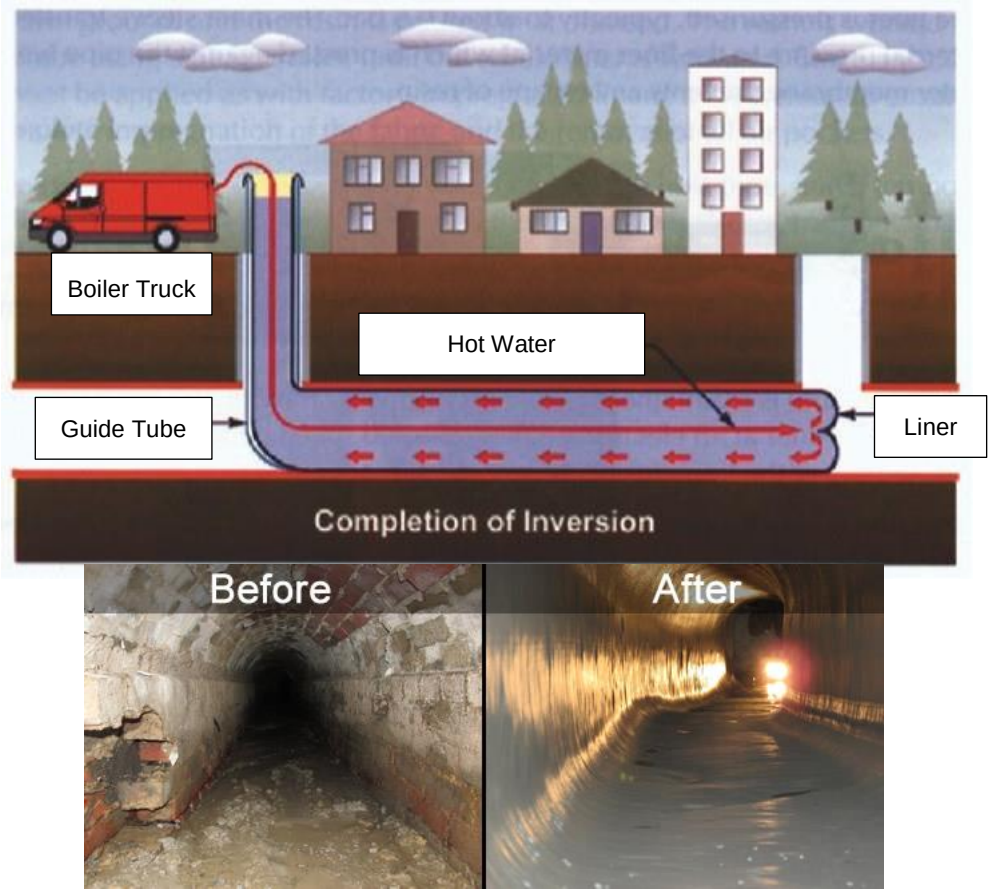


Figure 2.4 Cured-in-Place Pipe (CIPP) Lining.

(Top) Diagram showing how CIPP is carried out. Here, the liner has been inserted into the pipe and hot water is being pumped through it to force it along the pipe [39]. (Bottom) A photo of a sewer before treatment (left) and after treatment (right) [40].

Cured-in-Place Piping (CIPP), shown in Figure 2.4, is considered one of the most reliable trenchless pipe repair methods and can be done without any excavation at all. A polyester fabric-based liner is first fabricated to the requirements of the pipe. The liner is then inserted into the pipe using one of two methods. It can simply be pulled through the pipe using a winch and then inflated, or it can be turned inside out at the entrance of the pipe, then using compressed air or water, it is pushed through the pipe. The liner is then cured using UV light or hot water/steam. Once the liner has cured, a remotely operated cutter can then be used to re-establish the connection to any branch pipes. This method relies on the existing pipe having maintained

its circular or oval shape and cannot be used if the pipe has collapsed or been crushed.

Spray lining uses a rotary spray device that moves through the pipe, spraying a lining material on to the walls, usually concrete, mortar or an epoxy. The pipe interior has to be thoroughly cleaned beforehand using high pressure water. The use of concrete or mortar helps protect against corrosion and structural damage, but is very thick, reducing the internal diameter of the pipe, can block branch connections and takes a long time to cure. The main advantage of epoxy-based methods is the relatively fast curing time.

2.4 Summary of Water Pipes and Leakage Management

From the above review, it is clear that water pipe networks are in constant need of maintenance and repair. Large amounts of water are lost every day worldwide due to pipe network leaks and bursts, and this translates directly into large financial losses. There is therefore a strong financial incentive to find innovative and effective ways to prevent leaks from happening and to quickly and effectively repair them.

Although a water company can choose to operate on a reactive basis, only taking action when a leak is reported, it is better for the long-term health of the pipe network and for the company's own reputation to proceed with an active approach that constantly evaluates the health of the network and instigates repairs before a leak is formed, or before it gets too significant.

Monitoring for leaks is a multi-tiered process with various methods used to identify the location of leaks. These methods vary in their location resolution, with some methods identifying leaks at a district level all the way to methods that can identify leaks to within a few meters. They all have different strengths and weaknesses, which makes no one method suitable for all situations. For example, ground microphones can be used to pinpoint the exact location of a leak but require a skilled human operator, whereas leak noise correlators can be installed on a pipe and left to operate autonomously, but cannot be used effectively with all pipe materials and sizes. Most of the leak detection methods indirectly detect the leak by measuring its side effects rather than measuring it directly, which is likely why they suffer from these limitations. Methods which can be moved to the leak location either have limited control over their own locations (such as SmartBall) or are very labour intensive (such as ground microphones).

When it comes to repairing pipes, the standard procedure involves digging a trench around the damaged pipe to expose it, then to repair it by either replacing the damaged pipe section or installing a wrap-around sleeve. However, there are other trenchless methods that require a very small trench to be dug, or none at all, thus minimising disruption, repair time and cost. These methods can be split into two main types: those that replace the pipe by destroying it and installing a new one in its place, and those that repair the pipe by installing some sort of new lining material on the inside of the pipe.

Leak monitoring and localisation appears to be a viable application for an autonomous robot as the robot simply needs to act as a vehicle for the various on-board sensors, allowing them to get closer to any defects than existing inspection methods can. This allows their location to be pinpointed with high precision and reduces the uncertainty in determining whether or not a problem exists and its severity. The ability of the robot to carry multiple sensing instruments helps to avoid the limitations of individual sensing methods and allows a wide variety of different measurements to be taken at the same location, thus reducing any uncertainty. On-board communication could allow that information to be sent back to a control centre for either automated processing or human verification. The ability of a robot to navigate freely inside the pipe would allow human operators to easily perform detailed inspections of problematic pipe sections remotely, giving them a wealth of information that can be used to accurately assess the condition of the pipe and arm repair crews with the necessary information they need to quickly and effectively carry out their tasks. Using the flow of water in the pipe to generate power would allow the robot to be completely autonomous and self-contained and could potentially allow it to remain in the pipe for very long periods of time.

Based on this review, it appears that when it comes to repairing pipes, there is a large amount of additional equipment and material that is required. Even the most minimally invasive trenchless techniques require heavy hydraulic equipment for pipe bursting or large vehicles to carry the lining material for CIPP and spray lining. This is probably beyond the means of a small autonomous robot operating inside a pipe. It may be possible that the robot’s ability to get very close to defects and monitor their propagation early on would open up the possibility of small, minor repair work being undertaken by the robot itself. It is also possible that the robot could be used in conjunction with existing repair methods, by transporting tools and material

through the pipe. The feasibility of this could be assessed after such a robot is developed and tested inside pipelines.

2.5 Robot Locomotion

In order to design a robot that will work in the pipe environment, existing mobile and climbing robots were studied in order to see what locomotion methods have currently been explored and to draw inspiration from existing designs.

The following is a review of the locomotion methods used by mobile robots in general, with particular emphasis on their application in pipe inspection robots. The review is split according to the different categories of locomotion methods that have been studied so far.

2.5.1 Wheeled Locomotion

Wheels are one of the most commonly used locomotion methods when it comes to mobile robots. They are mechanically simple, easy to drive and control, energy efficient and are easy to miniaturise [41]. The flexibility of wheeled locomotion allows them to be used as the foundation for more complex, compound locomotion systems that give a robot much more mobility than using a standard, “car-like” configuration. For example, intelligent design of the mechanical linkages between wheels and the chassis is the reason behind the ability of space rovers to navigate rough terrain and keep all their wheels in contact with the ground [42], [43]. The addition of passive and active wheels to snake-like robots greatly reduces the friction coefficient between the robots and the ground, simplifying the kinematic model that governs the robots’ motion and allowing them to navigate horizontal ground relatively easily [44], [45]. Tracks combine many wheels and a track to form a single overall “wheel” that excels in traversing uneven terrain.

The simplicity and flexibility of wheeled locomotion has been widely used in pipe robots. As with other mobile robots, the use of wheels has been implemented using a variety of different methods depending on the robot’s specific intended application. For this review, robots whose primary method of locomotion involves actively driven wheels or tracks are considered to be wheeled robots. Robots that use passive wheels as part of another locomotion method are categorised under that locomotion method.

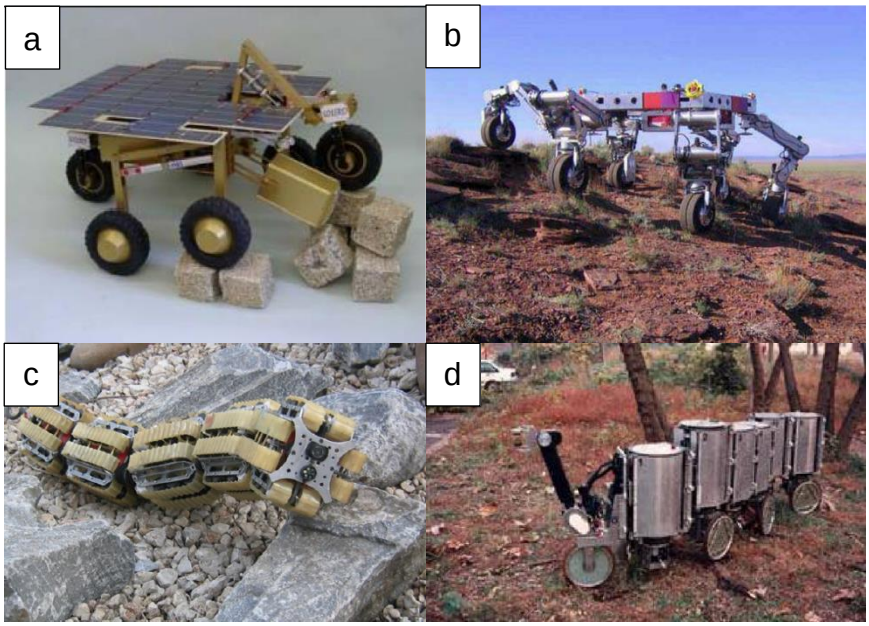


Figure 2.5 Examples of various wheeled robots.

a) SOLERO robot [42]; b) ATHLETE Wheel-leg hybrid system [43]; c) OmniTread [45]; d) Koryu-II [45].

For an internal pipe environment, a robot needs to make contact between its driving wheels and the pipe's internal surface in order to propel itself forward. A range of different techniques have been used by pipe robots to achieve this, and this can be used to broadly split such robots into three categories: those that rely on the weight of the robot to ensure contact (gravity wheel press), those that rely on an adhesion method to maintain wheel-pipe contact (adhesive wheel press) and those that press their wheels against the pipe wall using an actuator or elastic element (active wheel press).

2.5.1.1 Gravity Wheel Press

These robots are simpler than their active cousins and behave similar to regular wheeled vehicles in that they rely on their own weight to maintain contact between their wheels and the pipe wall. The majority of these robots are simple vehicles that provide a means of taking sensors and other instruments into the pipe. Examples of such robots have been discussed in [5], [46]–[52].

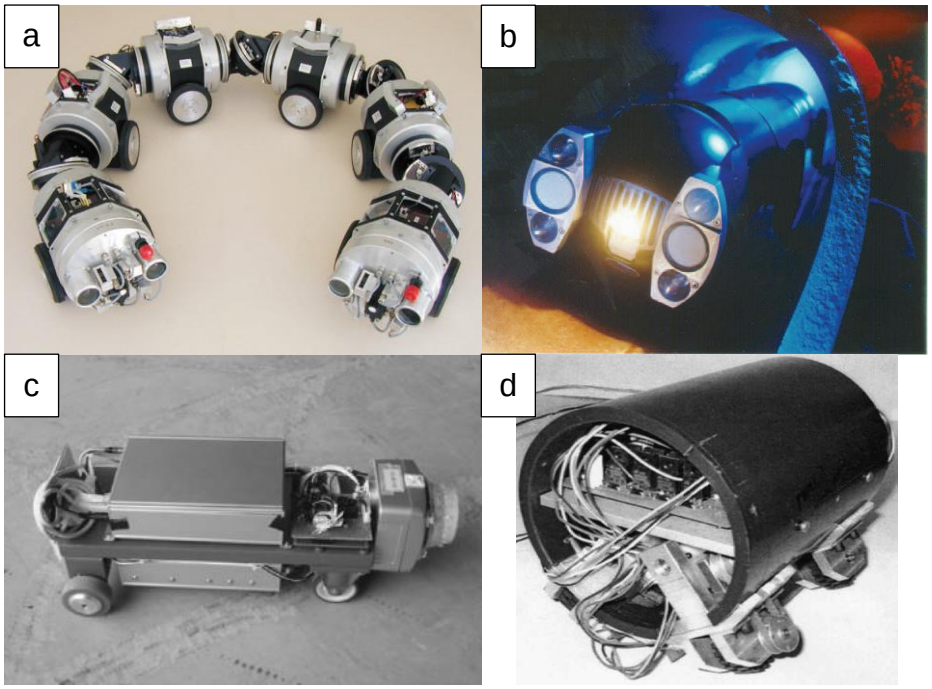


Figure 2.6 Examples of gravity wheel press vehicles.

a) MAKRO sewer inspection robot [52]; b) KARO sewer inspection robot [52]; c) Robot for testing wireless communication inside a pipe [49]; d) Gas pipe inspection robot [5].

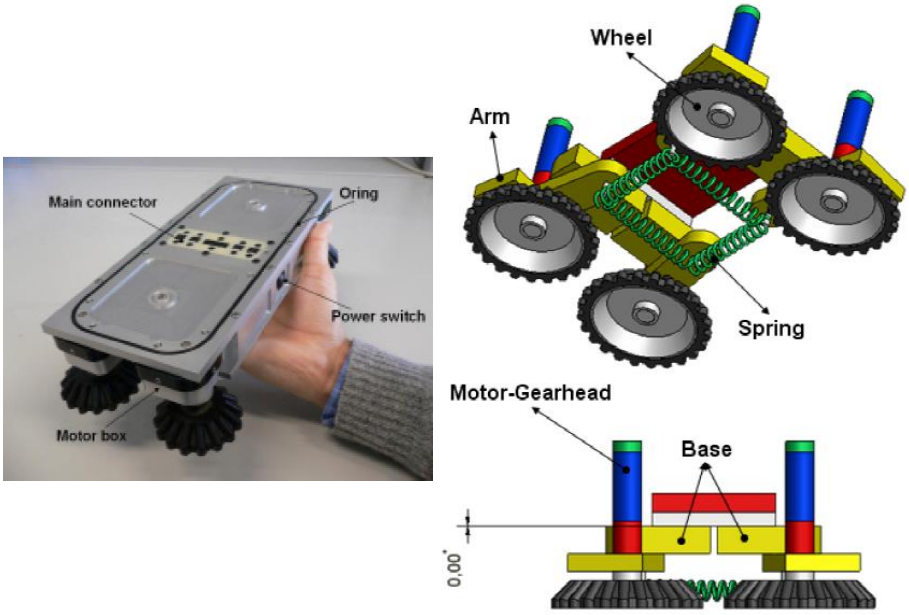


Figure 2.7 KANTARO Wheeled Robot [50].

One robot in particular stands out. The KANTARO robot presented in [50], [53] uses a unique wheel configuration, whereby a set of horizontally aligned, contra-rotating wheels drive the robot. These wheels are connected to the main chassis via passive springs. This configuration allows the robot

to passively overcome sharp bends and junctions with very little control effort.

The advantage of gravity-wheel-press robots is that they theoretically have no upper limit on the size of pipe which they can navigate. However, without a method of actively pressing the wheels against the pipe wall, they can only navigate horizontal pipe networks with limitations on the maximum incline that they can traverse. Such robots would not be able to navigate vertical pipe sections and would not be capable of operating in pipes with high rates of fluid flow as the robot would be swept away.

2.5.1.2 Adhesive Wheel

Adhesive wheel robots rely on a method of actively attracting the wheels to the pipe wall. Examples of robots using this method have exclusively relied on magnetism. Robots developed using this method are described in [54], [55].

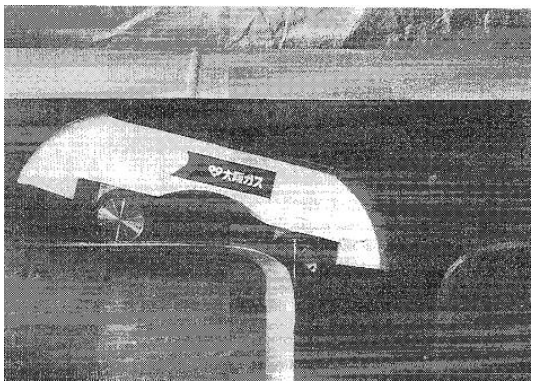


Figure 2.8 Osaka Gas Co. pipe inspection robot [54].

The robot designed by researchers at the Osaka Gas Co and the Kansai Research Institute (Figure 2.8) uses magnetic wheels to adhere to pipe walls and can navigate both horizontal and vertical pipes and bends. One of the unique features of the robot is their magnetic dual-wheel system. Of the problems faced by the robot in its environment is the presence of multiple bumps and protrusions inside the pipe, such as joins and welds. Using the dual magnetic wheel concept, a magnetic wheel is surrounded by a larger outer tyre. During normal operation, the dual magnetic wheel behaves like a regular wheel, propelling the robot forward on flat surfaces. When the wheel meets a sharp obstacle the outer tyre stalls, which would normally halt the robot. However, the inner wheel continues to rotate and can climb up the inside of the outer tyre, bringing it over the obstacle as shown in Figure 2.9.

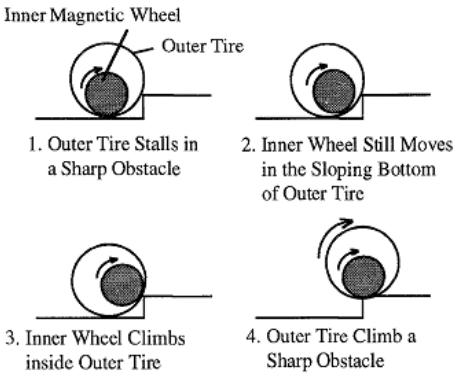


Figure 2.9 Dual magnetic wheel concept [54].

Using a dual wheel system, with a magnetic wheel inside an outer tyre, the wheel is able to overcome obstacles that would normally cause the wheel to stall.

Using this method, the robot is able to passively overcome certain obstacles, simplifying the mechanical design and the control system [54]. The efficacy of this system to adhere to the surface of the pipe, especially while upside down, is not discussed in the paper. Position 2 shown in Figure 2.9 could be a critical point because the magnetic force is spread between two contact points [1], potentially compromising the robot's gripping force.

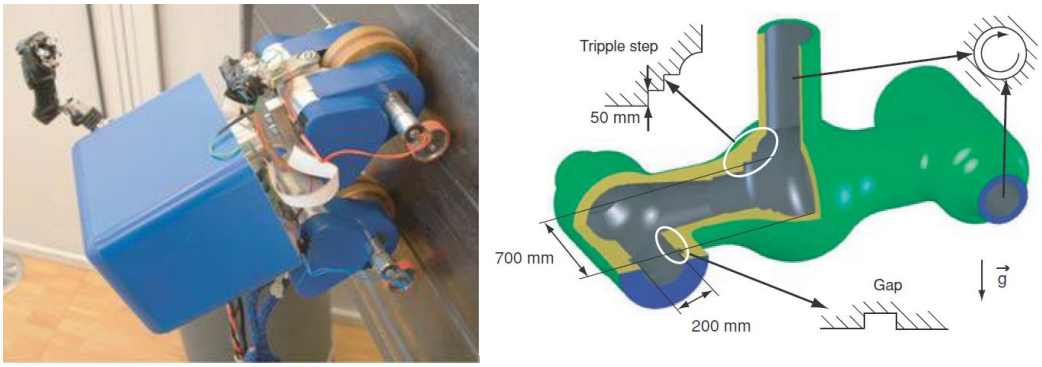


Figure 2.10 Magnebike inspection robot and its intended environment [55].

Further research into modified magnetic wheels was performed by [1] in the development of the Magnebike pipe inspection robot shown in Figure 2.10. The focus of their study was on how the magnetic force can be modulated on the wheel. When navigating a complex environment, there are several situations in which there are multiple points of contact between the wheel and the surface. It is necessary for the robot to remove the magnetic attraction force at some of those contact points in order to successfully navigate the obstacle, as shown in Figure 2.11. In order for the robot to continue moving downwards in this figure, the force F_{mag_1a} needs to be removed.

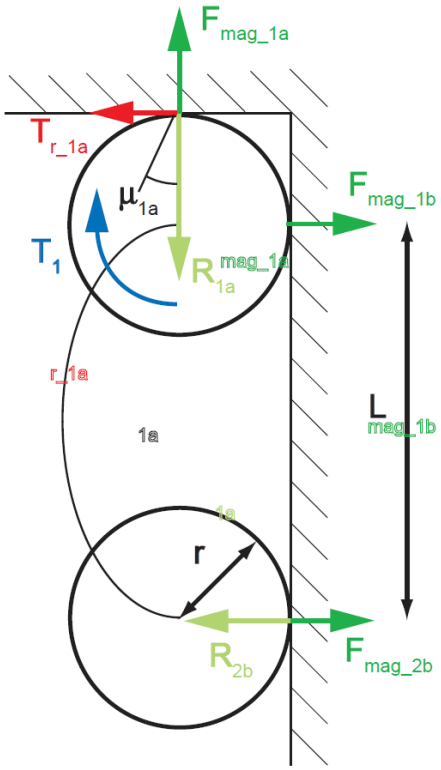


Figure 2.11 Forces acting on the Magnebike robot [1].

In order to continue moving downwards, the robot needs to remove the force F_{mag_1a}.

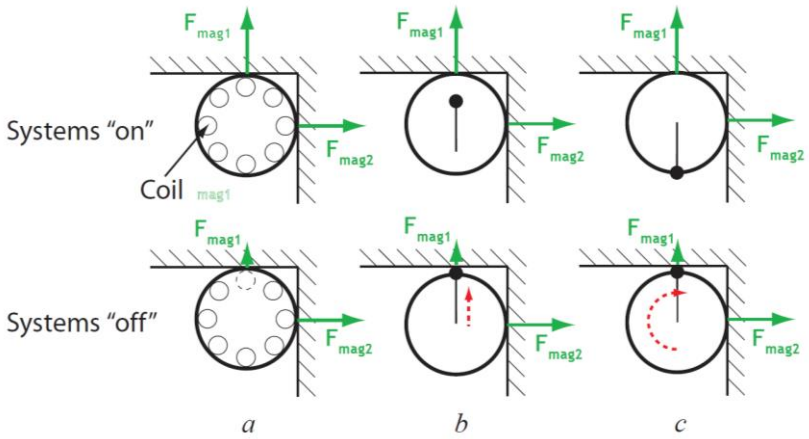


Figure 2.12 Local magnetic force modulation techniques [1].

a) Electromagnets are installed around the wheel's circumference, allowing the location of the magnetic force to be fully controlled; b) a linear lifter is used to push the wheel away from the surface at a particular point, thus reducing the magnetic attract force F_{mag1}; c) a rotary lifter serves the same function as the linear lifter, but can rotate to lift the wheel at any point.

Several methods of solving this problem were investigated. The dual magnetic wheel unit proposed by [54] was considered but there was no

information regarding the reliability of this method to keep the robot in contact with the surface when it is upside down. The use of global force modulation was considered, where the permanent magnetic wheel is replaced by a coil, allowing the attraction force to the surface to be changed. This, however, is not suitable as the force is reduced along the whole wheel and not just at the contact point, increasing the risk of the robot losing its grip of the surface. In order to avoid these problems, local force modulation is required. Several ways of achieving this were studied and are shown in Figure 2.12.

In concept (a), several coils of wire are distributed along the wheel. Each coil can be energized separately, allowing the attraction force around the circumference to be controlled. However, this method involves complex wiring and is less efficient than permanent magnets. Concept (b) uses a linear actuator to push the wheel away from the surface. This method only works if the contact point on the wheel is always the same. Concept (c) is similar to (b), except that the actuator rotates coaxially with the wheel, allowing the contact point to be anywhere along the circumference of the wheel. A magnetic wheel unit was designed using permanent magnets and rotary lifters, shown in Figure 2.13.

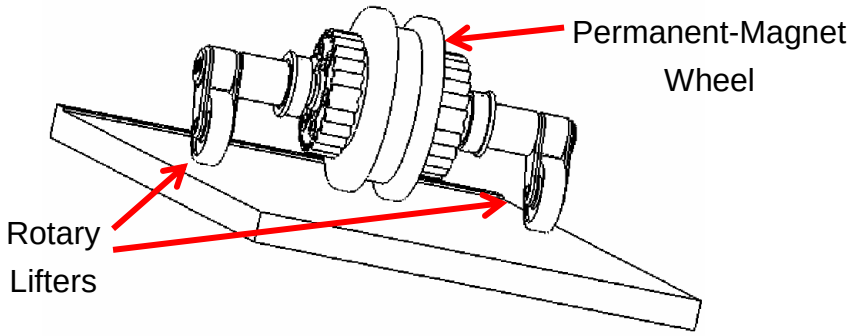


Figure 2.13 Magnetic wheel showing the method of force modulation using rotary lifters on either side of the wheel [1].

The rotary actuators on either side allow the magnetic force to be modulated anywhere along the circumference of the wheel. The actuators also serve as lateral stabilisers for the robot. The actuators are designed so that when the wheel is lifted off the surface to its maximum extent, there is still some magnetic force remaining. This is to prevent the wheel from losing the magnetic attraction force and to increase its reliability. The use of the actuators as stabilisers allows the robot to be reduced to two magnetic driving wheels, simplifying the design of the robot and allowing the robot to

be smaller in size. The robot is capable of successfully navigating complex obstacles using only two wheels, with 5 active DOFs and 1 free joint [56].

The design of both these robots means that they are not restricted by pipe diameter. However, their use of magnets inevitably limits their operational environment to those which are constructed primarily of ferrous materials. Also, since magnetic attraction force weakens with increasing distance, the ability for such robots to remain in contact with the pipe wall if it is covered in a thick coating or dirt is limited.

2.5.1.3 Active Wheel Press

Active wheel press robots operate by pressing their wheels against the pipe surface through elastic elements, such as a spring, using an actuator, or a combination of both. A common design for accomplishing this involves a robot with a central body and three wheel assemblies spaced 120° apart.

A typical example is the MRINSPECT series of robots. MRINSPECT consists of a set of three pairs of wheels circumferentially mounted around a central body. The wheel pairs are mounted on a link-spring structure which ensures that the wheels press against the surface of the pipe. Each wheel pair has its own driving unit and clutch, which helps to simplify the robot's control system by only allowing the critical wheels to be driven. This also increases the energy efficiency of the robot. The robot navigates through the pipe using differential steering and is able to navigate straight sections, bends and junctions [19], [57]–[59]. Similar approaches were used in the development of the inspection robots in [41], [60]–[65]. The robots presented in [66], [67] also use the same approach, but use tracks instead of wheels to increase the robot's traction and improve its obstacle overcoming ability.

MOGRER is another wheeled robot that expands to fill the pipe. Like MRINSPECT, the robot's wheels are mounted on articulated links to ensure that they remain in constant contact with the pipe wall. However, rather than using standard wheels and differential steering, MOGRER uses a steerable wheel and a skid wheel for each of its wheel pairs. This allows the robot to navigate the various features in the pipe and to move spirally inside it in order to orient itself correctly when navigating a bend or junction [68], [69].

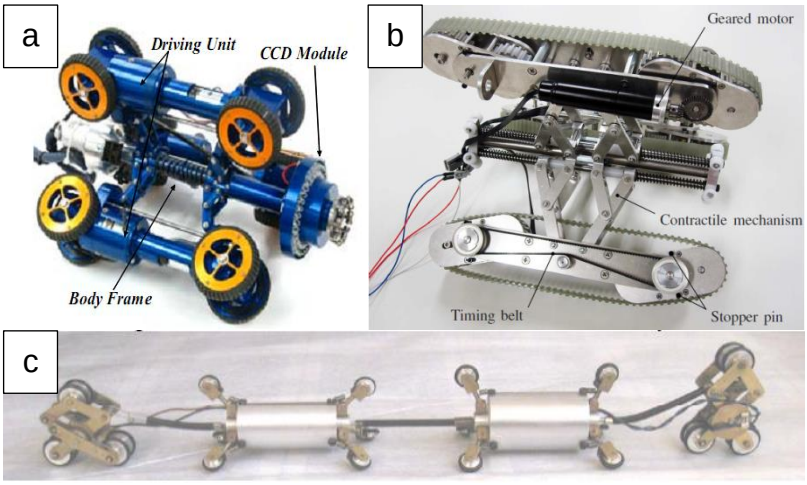


Figure 2.14 Active-wheel-press robots.

a) MRINSPECT V [19]; b) Tracked Parallelogram Crawler Robot [67]; c) Modular Microrobotic System [62].

A slightly different approach was used by [70]–[72] in their robot, shown in Figure 2.15. Rather than having wheels aligned along the axis of the pipe, the wheels in their robot are offset at a fixed 60° angle and are all driven from one DC motor in order to improve energy efficiency. The robot can move forwards and backwards along the pipe through a helical motion and can overcome steps. A modification of that approach, shown in [73], uses a single belt rather than multiple wheels, in order to improve the robot's ability to overcome obstacles.

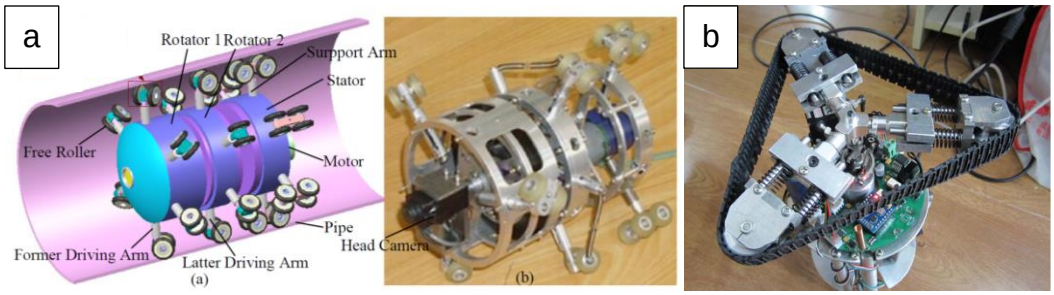


Figure 2.15 Helical motion-based robots.

a) 3D model and prototype of a helical drive robot [72]; b) Helical Belt Drive Robot [73].

A recent development of this principle uses multiple modules connected together via conical springs, as shown in Figure 2.16 [74]. As the individual modules twist, the conical springs can expand and contract, providing the robot with inchworm-like motion. The segmented nature of the design means that a long robot is still able to navigate bends in the pipe. However as the angle between the modules is not controllable, the robot cannot navigate junctions and can only navigate single branch pipes.

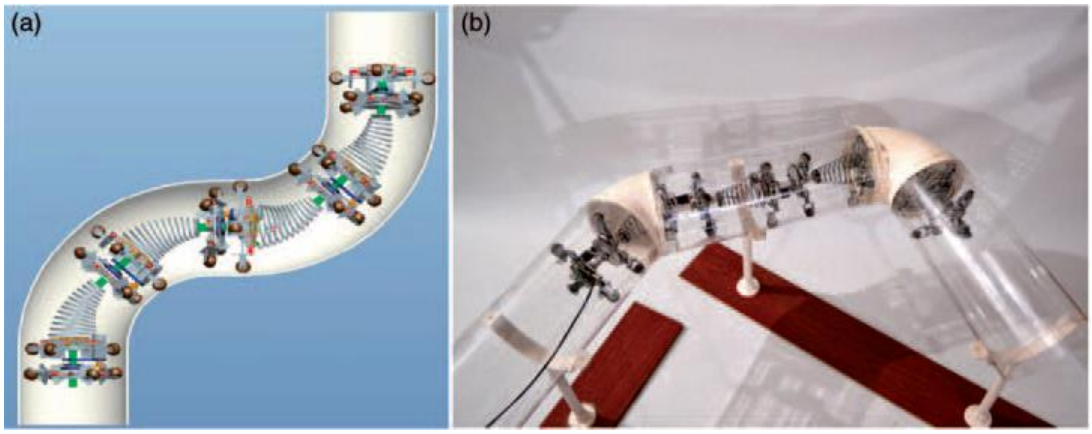


Figure 2.16 Segmented helical-drive pipe robot [74].

The Explorer robot and the robot presented in [75] are segmented active wheel press robots that are used for the inspection of gas pipelines. These robots are made up of several segments, with each segment having its own payload and sensors and with some having driving wheels. This allows the configuration of the robot to be changed and for the robot to be easily adapted in the field for different tasks. The joints between the segments are also motorised, allowing them to navigate bends and junctions. Unlike other pipe robots, both of these robots were designed to operate in pressurised, active gas pipes. The segments are designed to protect the internal workings from the high pressure inside the pipe and the shape of the robots designed to provide minimum resistance to the flow of gas [76], [77]. The main difference between the two robots is that Explorer is entirely electrically driven, whereas the robot presented in [75] uses compressed air to extend its drive arms and push the wheels against the pipe wall. The Explorer robot has been demonstrated successfully in field trials in gas pipelines.

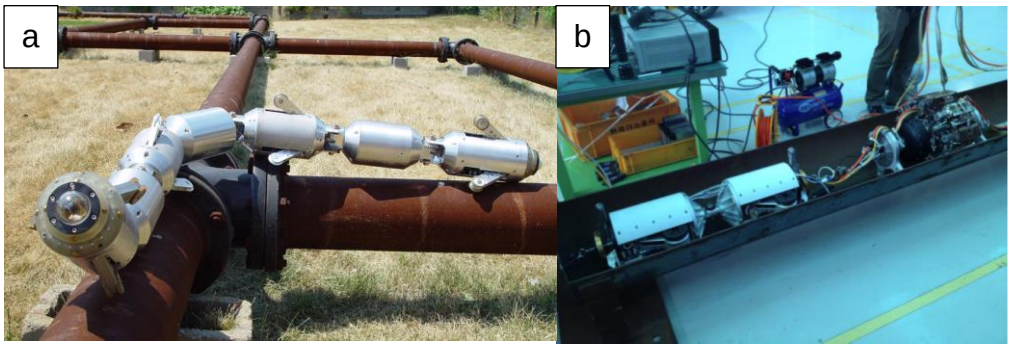


Figure 2.17 Active-wheel-press robots for operation in active gas pipelines.

a) The Explorer Gas Pipeline Inspection Robot [77]; b) Modular gas pipeline inspection robot [75].

In correspondence with the designers of Explorer, they reported that the wheeled drive used in the robot was not effective at navigating vertical

sections and tended to jam due to the pressure exerted by the supporting arms when they push against the pipe wall. This is because the force pushing the wheels against the pipe wall increases the effective load on the wheels, thus making them harder to drive.

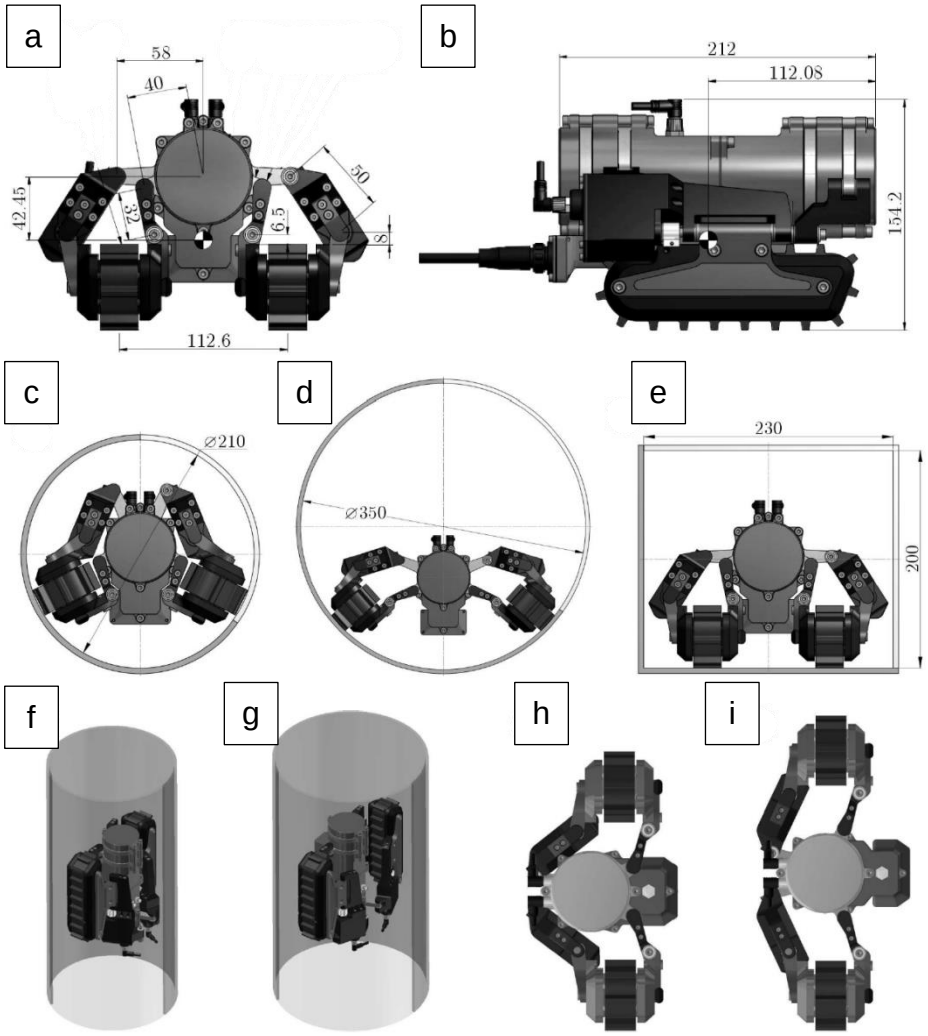


Figure 2.18 Hybrid gravity/active-wheel press robot.

Front (a) and side views (b) of the robot. The robot is capable of navigating pipes of varying diameters in both the horizontal (c, d) and vertical configuration (f, g), as well as travelling on flat terrain in non-circular pipes (e). The robot can retract (h) and extend (i) its tracks to help it conform to varying pipe sizes and climb vertical pipes [78].

A unique hybrid design was presented in [78], which is able to combine the benefits of gravity wheel-press and active wheel-press in one design. The unique feature of this design is that the tracked units of the robot are able to tilt by 90° as well as extend away from the main robot body. This allows the robot to move from a gravity wheel-press configuration that can move quickly in a wide variety of horizontal pipes, to an active wheel-press one that can climb vertical pipes of varying diameters.

2.5.2 Inchworm Locomotion

Like wheeled robots, inchworm-type robots are relatively simple to control and allow the robot to navigate the various features inside the pipe. The design of such robots can generally be categorised into two categories based on their method of expanding to fill the pipe: some robots employ a passive method, while others employ an active method.

2.5.2.1 Passive Inchworm

With passive inchworm robots, the mechanical design of the robot is such that when the robot is moved in one direction, its “arms” allow this movement to take place, but when the robot moves in the opposite direction, the arms prevent the movement, similar in principle to a ratchet mechanism. The simple nature of this movement means that such robots are easy to control and usually have very few parts. However, this simplicity means that they are unable to navigate junctions.

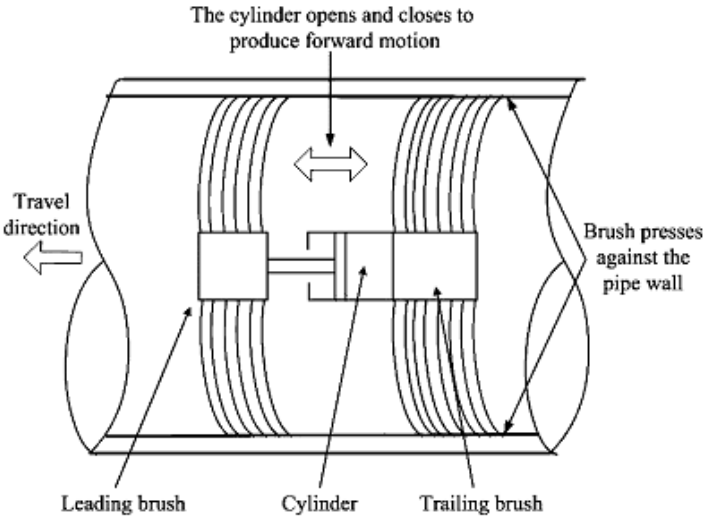


Figure 2.19 Locomotion method of a passive inchworm robot.

The robot moves by repeatedly extending and retracting its cylinder. The arrangement of the brushes allows for movement in one direction, but not the other, similar to a ratchet. The converts the oscillating movement of the cylinder into a linear movement in one direction [79].

Examples of such passive robots have been developed by [79]–[81]. Each of these robots uses some sort of vibration source as the main driving force coupled with a passive mechanical system pressing against the pipe wall, such as a brush or a set of angled bristles. However, these robots are limited to motion in one direction only, except for the one developed by [80] as they have two sets of feet facing in opposite directions. None of these robots can navigate junctions.

2.5.2.2 Active Inchworm

Active inchworm robots are the more commonly found type. With these, the robot has active control over its gripping mechanism and can attach to and detach from the pipe wall at will. This makes the robots more complex than their passive counterparts, but affords them more flexibility in their motion.

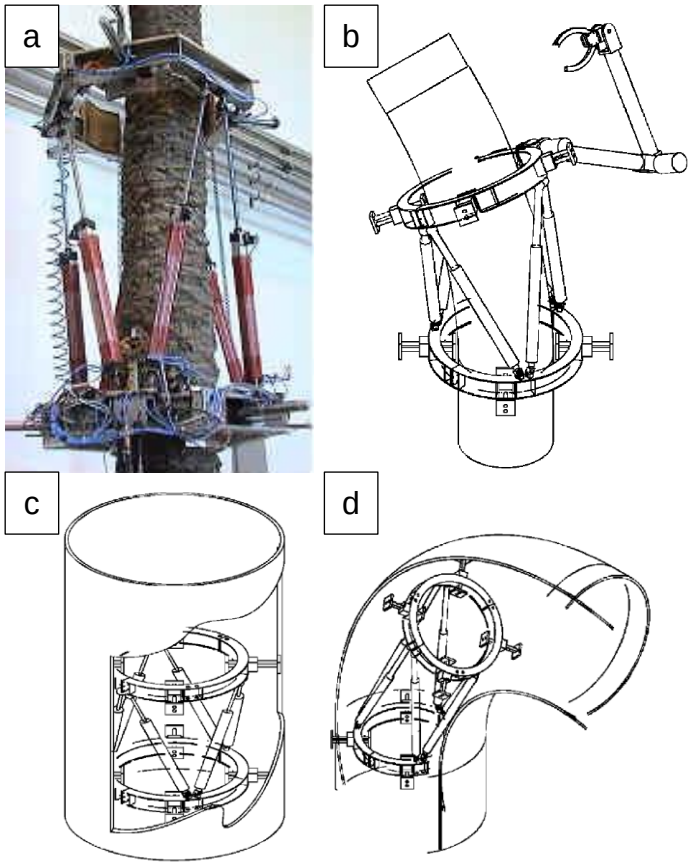


Figure 2.20 The TREPA parallel platform-based climbing robot.

a) TREPA prototype climbing up a tree trunk [92]; Some proposed theoretical configurations for external (b) and internal (c, d) pipe climbing [85].

Examples of such robots can be found in [82]–[91]. These robots all use some sort of linear motion for propulsion coupled with full control over the extension and retraction of their limbs, which allows them to easily move forwards and backwards along the pipe. Although all these robots work on essentially the same principle, their implementations are very different.

The TREPA series of robots presented in [85], [86] have used the principles of parallel robotics to construct a robot that is capable of climbing beams and tree trunks. The parallel actuators are responsible for moving the robot along the structure and allow it to change direction, while a separate set of actuators are used to clamp the robot to the pipe. Although not tested inside pipes, the authors have explored the concept of using the robot on the

outside and inside of pipes. A similar robot using a different parallel platform configuration was presented in [91]. It appears to perform similarly to the TREPA robots and has been tested in external pipe situations.

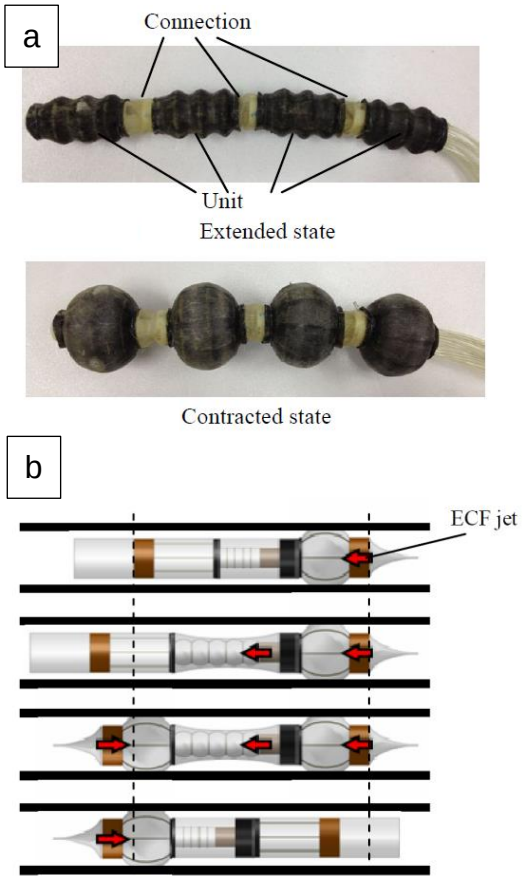


Figure 2.21 Peristaltic Inchworm-based Robots.

a) Pneumatic-based peristaltic inchworm robot [87]; b) ECF-based peristaltic inchworm robot [90]. For both of these robots, the bellows are inflated and deflated in sequence in order to propel the robot forward.

The robots presented in [87], [88], [90] use peristaltic inchworm motion. All of these robots use a set of inflatable bellows as their primary locomotion actuator. Inflation results in an increase in the bellows' width and a simultaneous decrease in its overall length. Therefore, the act of inflation allows the bellows to both grip the pipe interior and to move the robot in the longitudinal direction. In [87], [88] the bellows are composed of air muscles, with a conventional air supply being used to inflate and deflate the bellows. The robot presented in [90] is different in that rather than using air, the robot uses an electro-conjugate fluid (ECF), which is a smart fluid where a voltage applied across it can induce a flow. The robot uses this principle to move the fluid into and out of the bellows, thus controlling their shape. All of these robots consist of a series of bellows connected together via rigid sections.

The nature of these robots means that they are passively steerable, meaning that the robots will naturally follow any bends in the pipe without any active control. However, without a means of actively controlling the angles between segments, the robots are unable to navigate junctions. They also completely fill the pipe interior during operation and so would completely block the flow of fluid.

The robots presented in [82], [89], although different in their implementations, follow the same basic principle whereby a single linear actuator moves the robot forward and a separate linear actuation mechanism allows the robot to grip and release the pipe. The robot in [82] is powered by compressed air and uses one pneumatic cylinder to move the robot through the pipe, and two sets of four bar linkages powered by pneumatic cylinders to act as the robot's grippers. A universal joint is used to connect the main body cylinder and the gripper assemblies, giving the robot passive steering capability. In the robot presented in [89], the main pneumatic cylinder is replaced with a flexible worm drive, that moves the robot along the pipe. The robot uses a set of magnetic wheels to attach to the pipe wall. A set of brakes can be applied to the wheels of the front and rear halves of the robot to lock that half in place. As with some of the other inchworm robots, although these robots are able to passively steer, the lack of active control of the angle between the two parts prevents them from navigating junctions.

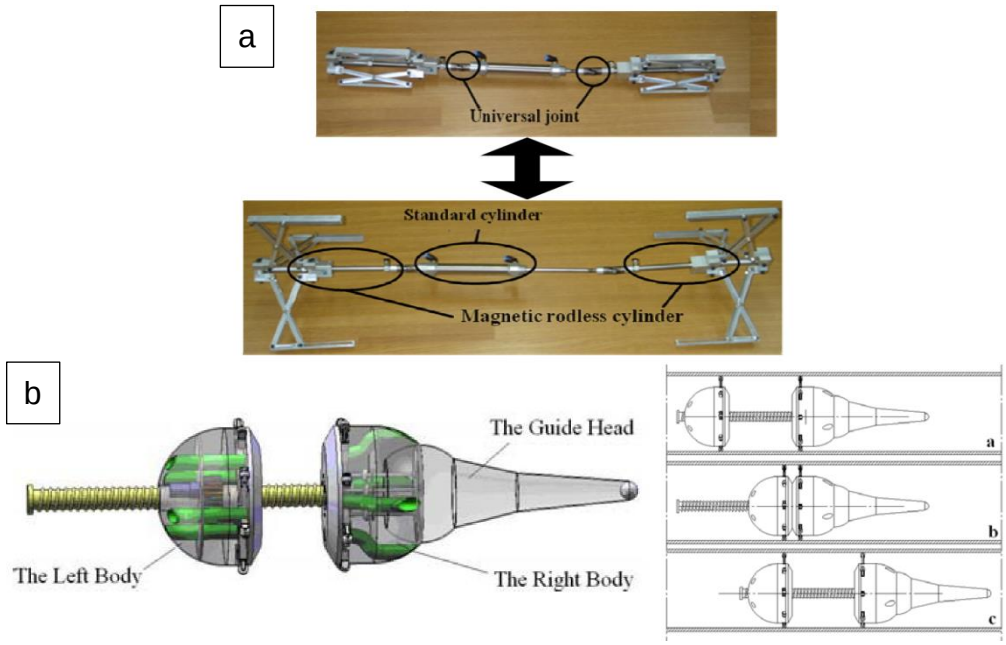


Figure 2.22 Pneumatic-based (a) and magnetic-based (b) inchworm robots [82], [89].

A unique inchworm-based robot is presented in [83]. This robot is constructed as a wire-frame cube, where each edge of the cube is constructed from a shape-memory alloy (SMA) actuator. This allows the length of each edge to be individually controlled via an electric current. The corners of the cube are covered in a frictional material to aid in gripping. Like other inchworm robots, the robot is able to use its front and back faces to grip the pipe wall, while its longitudinal edges are used to propel it forward. As the robot does not have a distinctive “front” and “back”, it does not share the same limitations that most other robots have regarding how it can change direction. This feature is exploited in the robot’s ability to navigate bends and junctions. Once the bend or junction has been entered, rather than rotating the robot, it simply moves sideways into the new pipe section, as shown below in Figure 2.23. However, although this works for junctions and sharp bends, it would not be possible in swept bends.

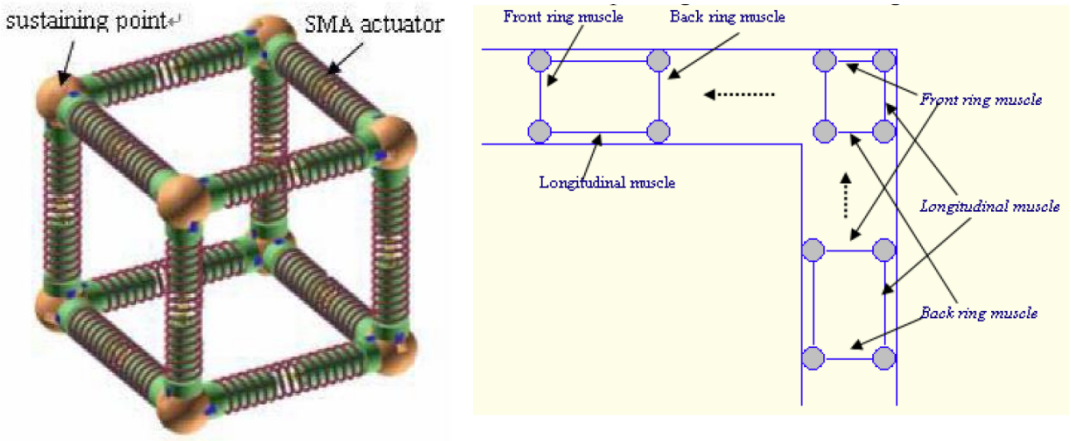


Figure 2.23 SMA based robot, showing how it navigates bends [83].

A more recent robot presented in [93] uses an octahedron as the basis of its frame. The robot is designed for operation in confined spaces, and has been tested in a pipe network, with the robot capable of navigating straight sections and T-junctions. Like with the SMA cube robot presented earlier, this robot is symmetrical and does not have a defined front and rear, allowing it to move in any direction without needing to turn itself. However, the authors report that the robot’s structure is a highly redundant, over-actuated system, which increases the control complexity and means that some motor commands will overstress the structure and potentially cause damage.

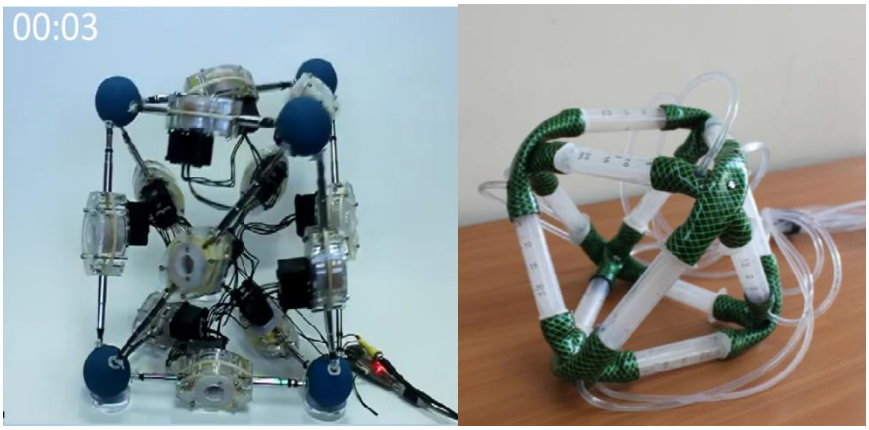


Figure 2.24 Two prototypes of the octahedron-based pipe robot [93].

2.5.3 Snake Locomotion

The ability of snakes to navigate highly variable, unstructured terrain and to pass through small spaces has garnered them a lot of research interest with the aim of harnessing those abilities in order to develop robots that can navigate difficult and constrained environments. There are a wide variety of different configurations that have been used to realise snake-like locomotion. These can be broadly split into two categories: robots that utilise purely undulation for their locomotion, and robots that utilise wheels or tracks to assist their locomotion. A review of the various different locomotion methods used by snake robots can be found in [45].

Snake-like locomotion has been used to develop robots that are capable of climbing pipes, both internally and externally, as can be seen in [20], [94]–[101]. These robots consist of several similar modules connected together using actuated joints. This gives the robots many degrees of freedom and a large degree of redundancy. Movement is accomplished through two methods. Most of these robots ([20], [94]–[98]) use travelling wave locomotion, where a sinusoidal wave is propagated along the length of the robot. This causes the robot to make contact with the pipe wall at the peaks and troughs of the wave, which moves it forward. The robot presented in [99], however, uses a different approach, whereby the robot coils itself into a helical spring shape and rolls up the pipe, in what is called the Rolling Helix Gait. Although this locomotion method maybe more complex to model and implement than the travelling wave method, it allows much more of the robot's surface area to be in contact with the pipe wall, thus giving a much better grip.

With undulation based locomotion, the friction between the robot's body and the pipe wall is critical in determining its performance. Changes to the outer

surface of the modules can be used to increase the performance of the robot in certain environments. This is achieved either through changing only the surfaces of the modules in contact with the terrain (compliance), or by encasing the whole robot in a skin. Compliance is used to add additional friction to the robot and has been observed to significantly aid in climbing vertical surfaces. Because it does not cover the whole robot, it neither limits the robot's movement nor does not hinder heat sinking. However, it does not protect the robot from adverse environmental conditions, such as dust or moisture. A skin, on the other hand, provides much better protection as it encases the entire robot. However, poorly constructed skins can limit the motion of the robot's joints and can lead to overheating problems [95].

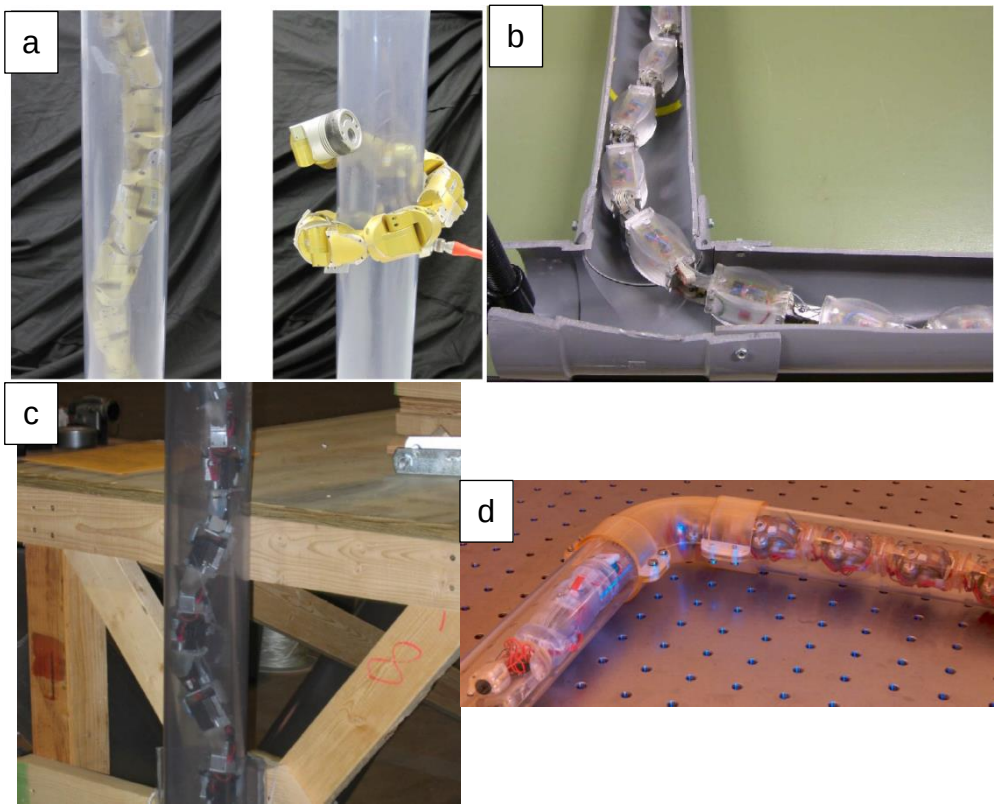


Figure 2.25 Examples of various snake locomotion robots.

References: a) [99]; b) [98]; c) [95]; d) [100].

The robot presented in [94] uses a slightly different approach than the rest by placing wheels around each segment. This allows the robot to navigate horizontal pipes similar to a simple wheeled vehicle and also navigate vertical pipes using travelling wave locomotion.

Figure 2.26 Snake robot using passive wheels [94].

The modular nature of snake robots means that different locomotion methods can be implemented using different modules, which can then combined in the final system. The snake robot presented in [100], [101] has multiple locomotion modules that allow the system to use either helical wheeled locomotion, inchworm locomotion or travelling wave locomotion.

The compact nature of snake robots means that they may be able to navigate severe constrictions and obstructions in the pipe better than other robot types. Their long, segmented structure may also allow them to navigate a very wide range of pipe diameters, as the robot is able to coil itself into a wide range of diameters.

One issue with snake robots is their payload carrying capacity. Since each segment in the robot is used for motion, any payload would need to be added in a separate segment, or integrated into existing segments by making one or all of the segments larger in order to accommodate the extra hardware. However, the first two solutions lead to inconsistency in the robot's configuration, with one segment not contributing to the motion of the robot or being larger than the rest, further increasing the complexity of the motion. Making all of the segments larger leads to inefficiency in space. Also, the nature of the undulation locomotion used by snake robots means that each segment is subjected to large displacements. This could make it difficult for sensors to take consistent readings of their environment. This issue was discussed to a certain extent in [96] with regards to the stabilisation of a camera module placed at the front of the robot. The stabilisation algorithms allowed the camera to remain parallel to the longitudinal axis of the pipe but could not prevent the camera from moving in

the lateral direction as the robot moved. The same control algorithm could not be used for sensors placed in the middle of the robot.

2.5.4 Legged and Crawler Robots

As with snakes and similar animals, legged animals also exhibit a high degree of mobility in nature, with the ability to navigate very complex and irregular terrain. Like snake locomotion, legged locomotion has also attracted a lot of research interest as researchers attempt to harness the stability, load bearing capability and dynamic mobility that legged creatures exhibit in nature. A wide variety of legged robots have been developed, from full size humanoid and quadruped inspired robots to small insect-like crawlers, to forestry tractors [102]–[104].

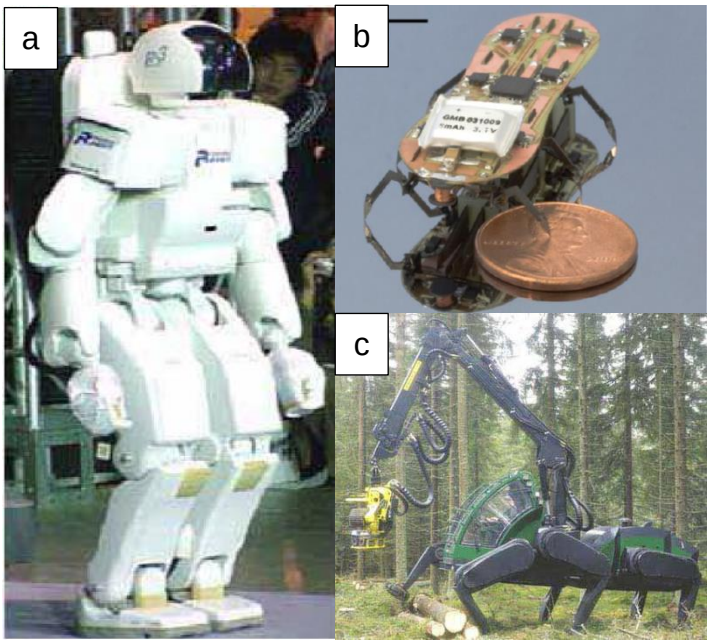


Figure 2.27 Examples of various legged robots.

a) Honda robot [103]; b) Harvard Ambulatory MicroRobot [102]; c) The Walking Forest Machine [103].

The use of legged locomotion has not garnered much attention when it comes to pipe inspection robots. The crawling robot MORITZ, presented in [4], is an eight-legged crawler designed for operation in pipes. The robot is composed of two crawler modules with four legs each, which are connected together via two actuated bending joints.

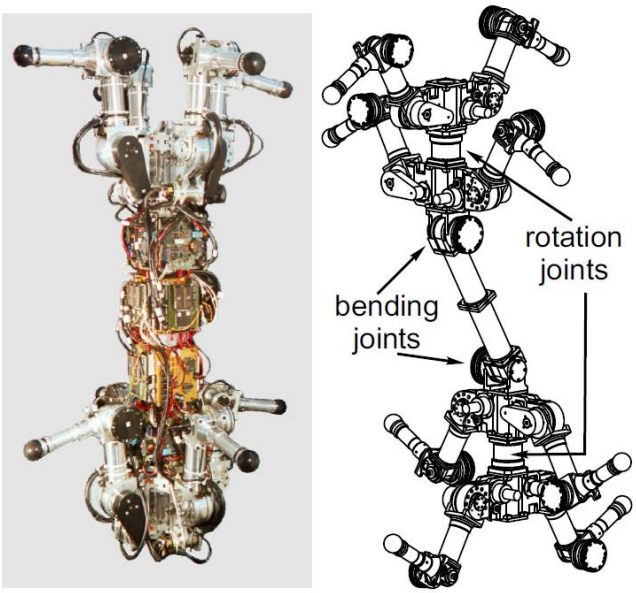


Figure 2.28 MORITZ legged robot [4].

The robot climbs up pipes by pressing four of the legs against the pipe wall to hold the robot in place while the other 4 legs move up the pipe. In order to navigate junctions found in pipe systems, the robot would use its two bending joints to twist its body, allowing it to turn into junctions, as shown in Figure 2.29.

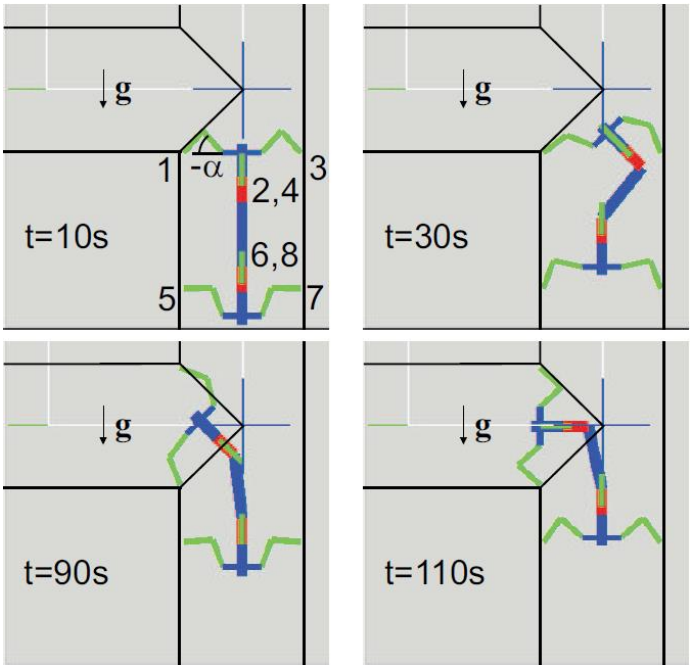


Figure 2.29 MORITZ's gait pattern for navigating junctions [4].

The ability of multi-legged insects to navigate unknown terrain quickly and reliably, despite having limited intelligence, was explored in the context of a legged pipe crawling robot in [3]. Insects are able to accomplish such movements by using simple reflexive/reactive behaviours instead of pre-

planned movements. In this study, a biologically-inspired layered control system was designed with the aim of simplifying the normally complex control systems of legged robots. The control system is split into four layers (from high-level to low level): the overall planning layer, a reactive layer for performing the robot's high level manoeuvres (such as steps, dodging etc), a reflexive layer for dealing with "emergencies" (such as collisions or kinematic limitations), and a low level servo control layer that is responsible for ensuring that each leg is pressed against the pipe wall with the desired force.

The bottom three layers of the control system were tested in simulation and were shown to perform well in a wide variety of different pipe configurations. A basic hardware prototype was also constructed and preliminary testing proved that the robot was capable of scaling vertical pipes.

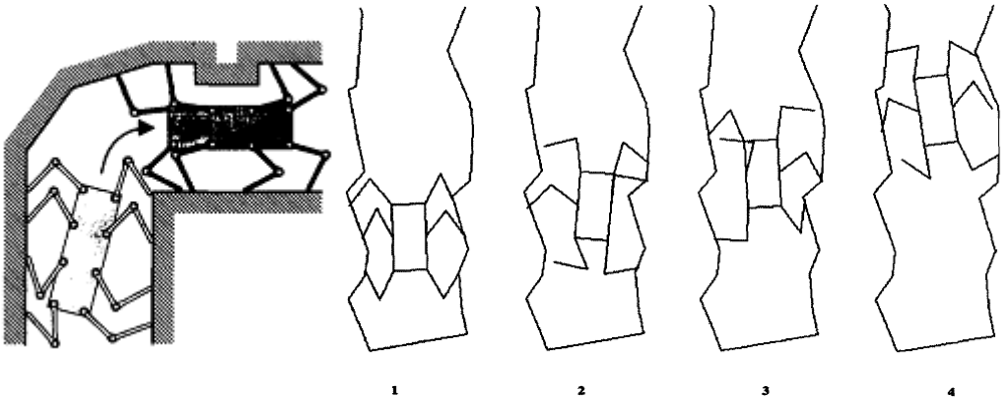


Figure 2.30 Spider-like legged robot [3].

2.5.5 Submersible and Swimming Robots

With water covering approximately 71% of the Earth's surface, it is not surprising that research effort has been exerted towards developing robots that can operate underwater. Underwater robots have fulfilled a wide variety of different roles from underwater survey and research, to search and rescue to inspection and repair. Underwater robots can be broadly categorised into two types depending on their locomotion method: those that attempt to harness the high mobility and agility found in aquatic animals by mimicking their design and movements, and those that use more "artificial" locomotion techniques, such as propellers, jets and fins [105]–[108].

For the purposes of this review, submersible and swimming pipe inspection robots are considered to be those that rely on the fluid inside the pipe for their propulsion. Such robots can be further split into two categories: those

that are passively propelled by the fluid (PIG devices) and those that actively use the fluid for propulsion (Actively Propelled Submersible Devices).

2.5.5.1 PIG Devices

PIG devices are some of the oldest in-pipe tools that have been used in pipelines as early as the 1870s, with the earliest ones composed of rags tied together with twine. Modern PIG devices are essentially plugs that are transported down the pipeline via the fluid flow, with the design of the PIG determining its function. The PIGs are inserted into the pipeline via a separate launcher tube, with the fluid flow then carrying the PIG down the line until it is ultimately caught by a receiver tube and then extracted.

PIGs vary widely in their complexity, from devices that are essentially moving brushes to help scour and clean a pipe, to sophisticated “Smart PIGs” that are outfitted with a wide array of sensors used for inspecting the status of the pipe, with some even including some form of velocity control where the PIG needs to scan the pipe at a constant rate [109]–[111].

An example of a smart PIG device used in water pipes is the SmartBall [34]. This device contains a wide variety of sensors encased in a waterproof, ball-shaped device. The SmartBall is inserted into an active pipeline and the flow transports the device along the pipe. The device takes continuous measurements while travelling, correlating the distance travelled with the measurements, allowing the precise location of problems to be identified.

Figure 2.31 SmartBall PIG System [112].

Despite the sophistication that might be found in some smart PIGs, these devices are generally quite simple and rely solely on the fluid flow to propel them along. At most, some form of active braking may be applied in order to regulate the speed. Because of their simplicity, PIGs are not able to navigate junctions and cannot move against the flow of fluid.

2.5.5.2 Actively Propelled Submersible Devices

Despite the natural association between pipes and water/fluids, there has been comparatively little research effort directed at developing pipe robots to operate in submersible environments under active control. Such robots would need to overcome the challenges involved in designing a complex electro-mechanical system to work underwater and in controlling such a device in the presence of strong fluid disturbances.

The two gas-pipeline inspection robots mentioned in section 2.5.1.3 were designed to operate in active pipelines, but the lower density and viscosity of gas compared to water might mean that such robots would not be able to operate in active water pipelines unmodified.

The robot concept presented in [113] was designed to specifically address the issue of an internal pipe inspection robot operating inside an active pipeline. The designers identified the limitations and inefficiencies of using an active wheel press-based design (discussed previously in the context of the Explorer robot in 2.5.1.3) and instead opted for the robot to be primarily driven by the flow of the fluid itself, essentially turning it into a PIG-type robot. A set of radial stabilising arms would keep the robot centred in the pipe. In order to control the speed of the robot, a set of magnetically controlled brakes would be used to apply a controllable braking force to the pipe wall, allowing the robot to control its speed. In situations where the flow is stagnant, the robot can be fitted with a propeller, allowing it to propel itself under its own power at a speed of 0.2m/s. However, unless a significantly powerful propulsion system is used, the robot would probably not be able to move against the flow. The robot’s segments would be connected together using a flexible polymer, giving it the ability to passively steer, although the lack of active joint control means the robot is unable to navigate junctions. The magnetic brake, polymer body and propeller propulsion systems were tested individually to validate their performance, but were not integrated together into a final robot.

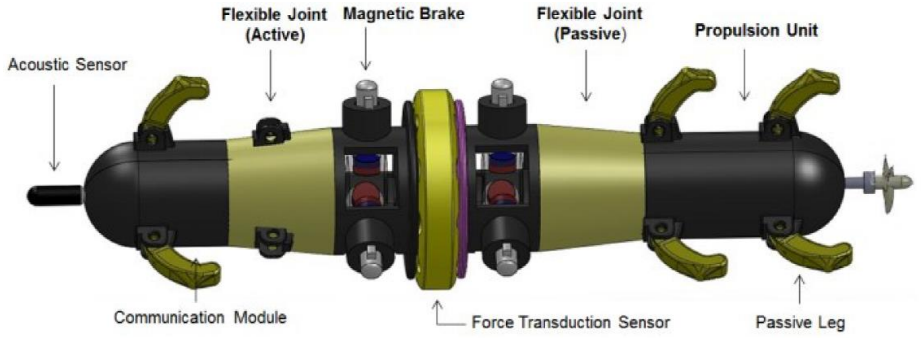


Figure 2.32 Submersible pipe inspection robot concept [113].

The PICTAN vehicle [114] is an autonomous underwater vehicle (AUV) that has also been designed to perform inspection tasks in active water pipes. Unlike the vast majority of pipe inspection robots that rely on making mechanical contact with the pipe wall to aid in movement and stability, this robot is a purely swimming robot that is equipped with thrusters around its body to help it control its attitude and position inside the pipe.

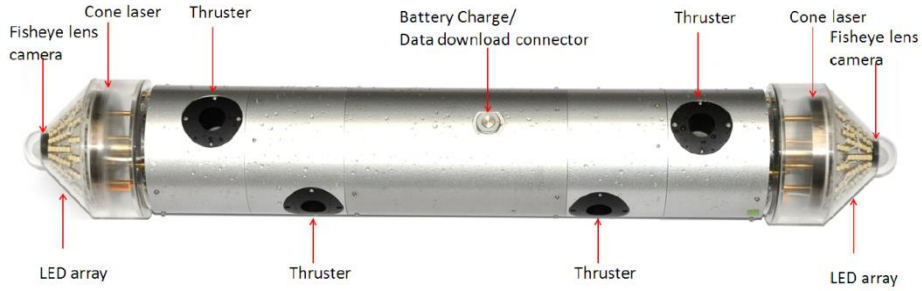


Figure 2.33 PICTAN pipe inspection AUV [114].

Like the previous robot presented in [113], PICTAN also primarily relies on the fluid flow for propulsion, with the thrusters allowing it to control its attitude and position. The PICTAN is equipped with an interesting vision system that uses a cone laser to project laser light on to the pipe interior ahead of the robot, then a camera to capture the resulting image and analyse it. Shining the laser on to the pipe wall produces a circular light pattern that is captured by the camera. By analysing where the centre of the light pattern is relative to the centre of the image, and comparing the images from the front and rear cameras, the robot's attitude and position inside the pipe can be calculated. As well as working in still water, the robot has been tested in a pipe with the flow rate at 30mm/s.

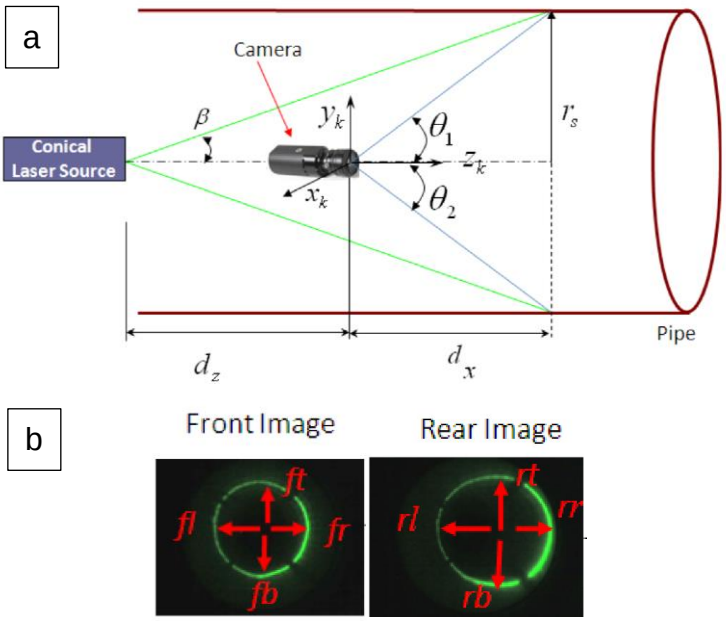


Figure 2.34 The PICTAN robot "vision" system.

The PICTANs vision system is composed of a camera and a conical laser (a). The laser produces a circular pattern on the pipe wall, which is recorded by the camera. By using the dimensions and position of the circular pattern in the image (b), the robot's attitude and position can be determined [114].

2.6 Comparison of Pipe Inspection Robots

Pipe inspection robots have been developed using a wide variety of different locomotion methods and design topologies, with varying degrees of flexibility when traversing pipelines. Robots from each locomotion category have been designed that are able to navigate all the main features of a pipe (straights, bends, T-junctions).

Legged and snake locomotion methods attempt to harness the sophisticated and flexible motion that such animals exhibit in nature. Robots that use such methods tend to have many degrees of freedom and are highly redundant. This allows them to excel in unstructured and unpredictable environments. However, what is apparent from the review on pipe networks is that pipes are generally very structured and predictable environments, despite the occasional presence of obstructions such as valves and partial blockages. The high locomotion ability of legged and snake robots would not be fully utilised in such an environment and therefore the added mechanical and control complexities are not warranted. Robots using simpler locomotion methods have been able to successfully navigate pipe networks and therefore, those two locomotion methods were not considered to be the most

appropriate solutions to this problem. This is also reflected in the relatively small amount of pipe inspection robots developed using these methods.

When it comes to submersible or swimming robots, two main categories of robot were explored in this review. PIG devices are very simple and rely solely on the fluid to transport them along the pipe with very little, if any, control over their movement. At most, some form of controlled braking may be implemented in some smart PIGs, but they are unable to move through junctions and cannot move against the flow of fluid. Submersible robots have been developed using controllable propulsion methods. One design was in essence a smart PIG that relied on the fluid for propulsion but used a magnetic braking system to control its speed and was equipped with a propeller for use when the fluid flow was stopped. This robot was not able to navigate junctions and would likely not have been able to move against the flow of fluid using its propeller. Another robot was implemented as an AUV using water jets to adjust its position in the pipe. Although the robot was tested inside a pipe with water flowing at 30mm/s, it is not expected that the robot would be able to control its position with water flowing at much higher rates, such as 800mm/s and in turbulent flow, as it does not have any mechanical method of bracing itself against the flow.

The use of wheels as a robot's primary locomotion method appears to be the most commonly used locomotion method for pipe inspection robots. Their relative simplicity and effectiveness allows high speed movement to be achieved using simple control systems. Robots using this method have also been shown to successfully navigate all the main features of a pipe. However, the practical experience of the Explorer robot, which has been tested in live, pressurised gas pipelines, demonstrates the main problem when using wheels for a climbing robot. In order to provide a good grip for the wheels, the robot needs to push them out against the pipe wall. However, this makes it more difficult for the motors driving the wheels to actually drive them forward. This is especially the case when the robot is climbing vertically and having to pull its own weight up the pipe, and would be an even greater problem if the robot were to operate with liquid flowing past and having to move against the flow. It was therefore deduced that wheeled locomotion is not the most efficient locomotion for climbing.

Inchworm locomotion was another relatively popular locomotion method for pipe inspection robots, with robots using this method able to navigate the main features of a pipe. Like wheeled locomotion, it is a relatively simple locomotion method to control, but can be slower than using wheels due to

the non-continuous nature of the movement. However, unlike in wheeled locomotion, where the force pushing the wheels against the pipe wall acts against the wheel driving force, the forces in inchworm locomotion can be completely decoupled from each other. This allows the robot to apply a very large force to hold its place without affecting the main forward force at all, making inchworm locomotion more efficient for climbing robots. This efficiency becomes significant in situations where the robot will be subjected to a high load, such as a high flow rate, where it will need to apply a large gripping force. Inchworm locomotion was therefore deemed to be the most appropriate locomotion method to use for a pipe inspection robot that needs to navigate pipe networks with high flow rates.

2.7 Summary

In this chapter, a review was carried out on the condition of pipelines, the reasons why defects occur and the various methods used by water companies to both detect leaks and to repair them. A review of the locomotion methods used by mobile robots was also presented and examples of how each of these methods has been used by in-pipe inspection robots were shown.

By understanding the pipe networks and how water companies detect and locate leaks and subsequently repair them, it became apparent that a robotic system could be of benefit to the industry, particularly in the field of leak detection and localisation. This is due to the robot's ability to autonomously get very close to the source of the leak and carry multiple instruments in order to take a wide range of different measurements, greatly reducing the uncertainty in determining the location, type and severity of the leak. It was not immediately apparent that such a robot would be able to replace existing repair methods, although it may be able to assist them. Operational experience inside the pipe would be needed in order to determine how the robot could be used for doing physical repairs.

Reviewing the various locomotion methods used by mobile robots and how these have been applied to existing pipe inspection robots showed that there are a wide variety of different designs that have been used in this field, with wheeled- and inchworm-based designs being the most common. However, very few of these were designed to operate in active pipelines and thus did not factor in the effects of the fluid flow. The few robots that were designed to operate in live water pipes would not have been able to fully control their direction of motion if the flow were very high. From the review, it was

deduced that inchworm locomotion was the most appropriate locomotion method for this application due to its relative simplicity, its flexibility in navigating the various pipe features, and its efficiency when it comes to climbing with very high loads or drag forces.

Chapter 3
Specification and Methodology

3.1 Introduction

In this chapter, the requirements and specification of the robot are presented, along with the methodology that was followed in developing the robot.

3.2 Requirements

From the literature review, it became evident that developing a flexible, autonomous robotic platform that is capable of navigating the various features of the pipe while carrying different measuring instruments would be very beneficial in determining the precise location and nature of defects in the pipe, as well as assessing its general condition. The ability of this robot to operate while the pipe is still active allows for minimal disruption in service and can be used as a source of power for the robot, allowing it to operate for extended periods of time.

Based on this, in order to develop the ideal robotic platform, the following set of requirements were derived:

- The robot must be able to navigate all the features of the pipeline and overcome any obstacles found there.
- It must provide minimum resistance to the flow, firstly in order to reduce the forces acting on the robot and allow it to use smaller and cheaper actuators, and secondly so as to have minimal effect on the operation of the pipeline.
- It must be waterproof, so that the presence of the fluid does not adversely affect the operation of the robot’s mechanical and electronic systems.
- It must not have any adverse effect on the quality of the water, so that it can be used in active clean water pipes.
- It must be able to generate and store its own power as it can be expected to operate for extended periods of time in a pipeline.
- It must have the tools and dexterity needed to carry out as many of the required inspection and maintenance tasks as possible, thus eliminating the need to shut down and excavate pipelines in all but the most serious situations.

- It must be completely autonomous and able to move through a pipe network in a systematic and efficient manner, assessing the condition of the pipeline, performing maintenance tasks where required and reporting information back to a base station.
- It must be reliable in order to avoid any expensive operations to rescue the robot from the pipe. If a failure is encountered, the robot must fail in a predictable manner that allows it to be extracted as easily as possible.

3.3 Target Specification

The development of such a robot would cover a wide range of different disciplines and skills. For this thesis, it was decided to focus solely on the locomotion aspect of the project, specifically, the development of the robot's mechanical design that would then be used as the basis for further development. Based on this and the water pipe properties mentioned in 2.2, the following specification was derived that the robot's mechanical design should adhere to:

- The Robot must be able to navigate pipe diameters within the range of 75-450mm.
- The robot must be able to navigate vertical and horizontal straight sections, swept bends and tees.
- The robot must be able to navigate against a maximum flow rate of 0.8m/s at a pressure of 8bar.

The following assumptions were made about the pipe network in order to simplify the design and to initially focus on the robot's ability to move inside an active pipe:

- There would be no obstructions inside the pipe.
- Only a single diameter and bend radius would be tested in order to avoid the issues faced by actuators with very large extensions.

3.4 Methodology

In order to achieve the aims of the project, the following methodology was used:

After understanding the nature of the robot's working environment and reviewing the existing literature on pipe robots, the first step involved coming up with various conceptual ideas for the robot's design. Each of these concepts was analysed according to various criteria and the most suitable concept was chosen.

After deciding on a concept, the detailed design was carried out. This involved estimating the forces that the robot would be exposed to and using that information to select the most appropriate actuators for it. The Solidworks 3D CAD package was then used to design the mechanical connecting components of the robot and to assemble the complete design in a virtual environment. The electronics of the robot were designed based around an off-the-shelf motor controller and a custom-designed add-on circuit that would overcome the shortcomings of the motor controller. The PC control software was then written to allow the motor controllers to be configured and controlled. The robot's various components were then manufactured and purchased and the robot prototype was assembled.

Once the detailed design was finalised, the theoretical analysis stage was begun where various aspects of the robot were analysed in order to confirm that the initial design estimates were valid. The robot's inverse kinematics were derived algebraically and then a simulation model was constructed in the MSC Adams simulation package. This was used as a preliminary way of verifying the equations' validity. The model was then used to determine the working envelope of the robot and understand its movement limitations, in particular, the robot's limitations regarding moving in a bent pipe. The forces exerted on the robot due to the fluid were estimated via simulation. Various models of the robot inside the pipe were constructed using the ANSYS CFX CFD package and the forces exerted on it were calculated for various different scenarios in order to find the maximum force. This was then used to assess the feasibility of the robot overcoming the fluid force.

After the theoretical analysis stage was completed, the practical verification stage begun. First, the inverse kinematics equations derived previously were verified using a motion capture system. This also confirmed that the Adams simulation model was valid. The bend radius limitations determined previously were then used to construct a pipe bend that would be suitable for the robot. The final step in the practical verification involved testing the robot inside the constructed pipe bend and in a straight pipe to demonstrate that the robot was capable of navigating the basic features in a pipeline.

3.5 Summary

In this chapter, the overall project requirements, the target specification for this project, and the design methodology were outlined.

Based on the literature review, a set of overall project requirements were drawn up that the ideal robot would need to meet in order to fulfil its tasks.

For this project, it was decided to focus on the robot’s mechanical structure. From this, a target specification was outlined that this project would need to achieve. The design methodology that was used to develop the robot was then presented.

Chapter 4
Conceptual Design and Assessment

4.1 Introduction

In the previous chapter, a specification was laid out that determined the requirements that a pipe inspection robot needed to meet in order to be successful in it's designed role. This chapter presents the conceptual design process that was used to convert the requirements outlined earlier into a design concept that meets those requirements best.

Based on the requirements, a set of primary design criteria were thought up that the design would need to optimise if it were to be successful. Based on these criteria, several design concepts were considered with each concept building on the previous one until a final solution was found.

4.2 Design Criteria

In the previous chapters, a study of the pipe environment and existing solutions resulted in a set of design requirements and specifications for the robot. In order to apply those requirements and specifications in the design process, a set of specific design criteria were defined and examined in detail in order to understand how those criteria should be applied to the final design. This was done by examining each of the requirements and then breaking them down into the physical properties that the robot must have in order to meet those requirements and how best to optimise those properties.

The requirements that were examined are listed below:

- Ability to overcome the fluid flow:
    - Drag Force
    - Pressure Drop
- Reliability and simplicity.
- Use of inchworm locomotion.

4.2.1 Overcoming the Fluid Flow

Since the robot will be designed to operate in active pipelines, it needs to be able to minimise the effects of the fluid flow on its operation. The fluid flowing past the robot will create a drag force in the direction of the fluid flow, with the magnitude of that force being linked with the robot's design via several parameters, which will be discussed. In order to successfully overcome this

force, each of these parameters needs to be optimised in order to minimise its contribution to the force.

The fluid force is expected to come from the following main sources:

- The drag force generated by the fluid flowing past the robot
- The force generated by the pressure generated across the robot due to the flow restriction it creates in the pipe.

The parameters affecting each of these sources is explored below.

4.2.1.1 Drag Force

As fluid flows past an object, the fluid molecules collide with the object, transferring some of their kinetic energy to it. This manifests itself as a drag force pushing the object in the general direction of the flow. The equation governing drag force is shown below:

$$F_{Drag} = \frac{1}{2}\rho v^2 C_D A \quad (1)$$

The variable definitions are defined in Table 4.1.

Table 4.1 Drag equation variable definitions

Variable	Definition
F_{Drag}	The drag force
ρ	The fluid density
v	The fluid velocity
C_D	The drag coefficient
A	The frontal area

Out of the four input variables, the frontal area and drag coefficient can be directly controlled through the robot's design. Although the fluid velocity is not directly controllable through the robot's design, it is possible to minimise it as will be explained below. The fluid density is the only variable that the design cannot affect at all.

Figure 4.1 presents a graph showing how the various variables affect the drag force.

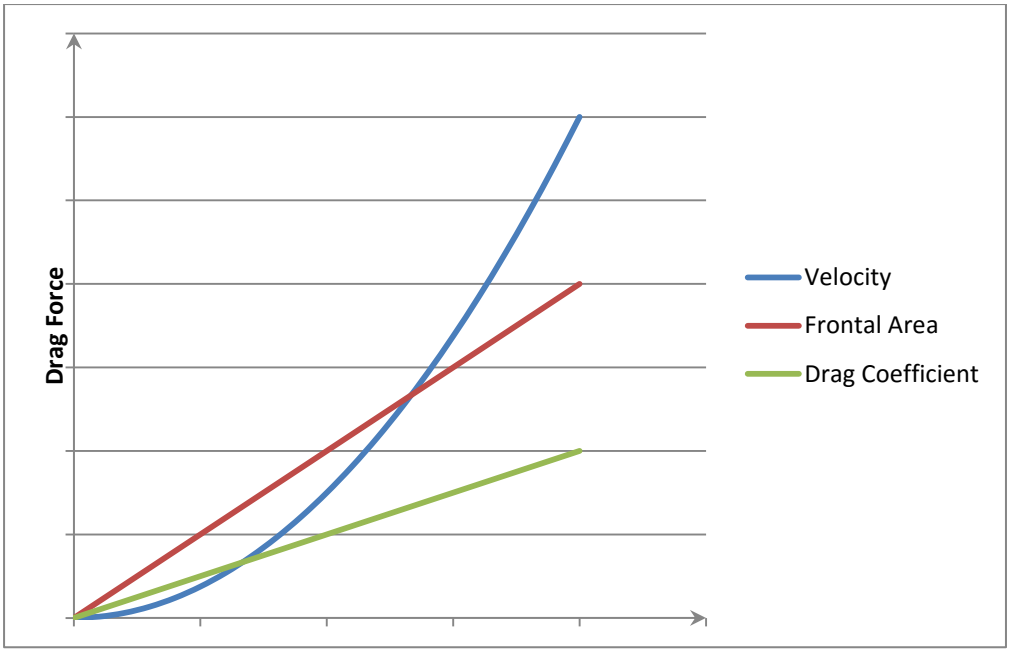


Figure 4.1 The relationship between the drag force and its variables.

The relationship between the drag force and the velocity is exponential, whereas it is linear for both the frontal area and drag coefficient.

As can be seen from the graph, increasing any of the variables increases the drag force. Therefore, the aim of the design should be to minimise all of those variables as much as possible.

4.2.1.1.1 Frontal Area

The frontal area represents the area of the object that is projected on to a plane whose normal is in the direction of the fluid flow. The same object can have very different frontal areas depending on how the object is oriented relative to the flow. This is demonstrated in Figure 4.2.

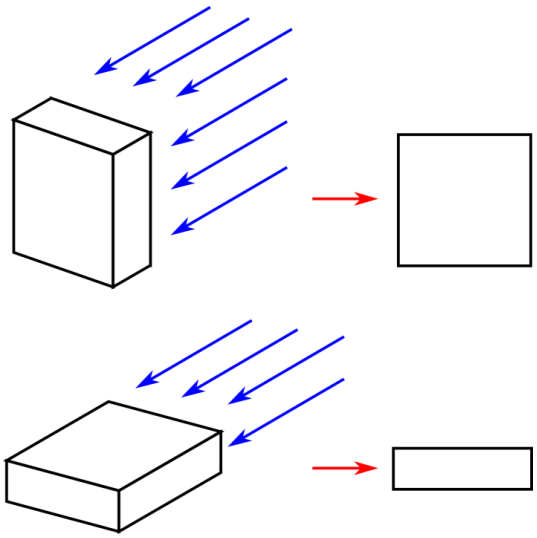


Figure 4.2 The frontal area of the same object can vary significantly depending on its orientation relative to the fluid flow.

Other than junctions (which form a very small proportion of pipe networks), pipes will naturally restrict the flow of water to one direction only, which is along the pipe's longitudinal axis. Therefore, when designing the robot, it will be important to minimise the frontal area as viewed along the longitudinal axis and to try and arrange the mass of the robot so that it does not increase the frontal area. Figure 4.3 shows how two objects with different masses and volumes can have the same frontal area. This demonstrates that despite one object having more mass and volume than the other, it is possible to keep the frontal area small through careful arrangement of the object's mass.

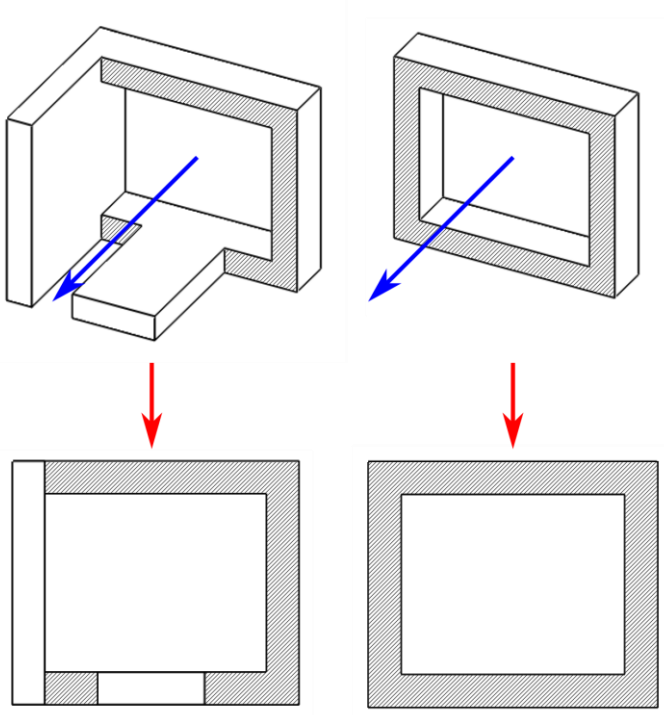
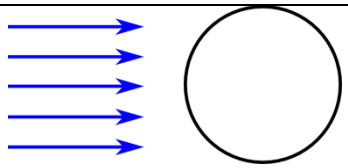
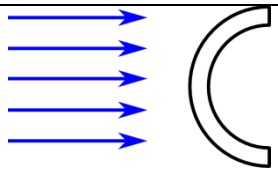
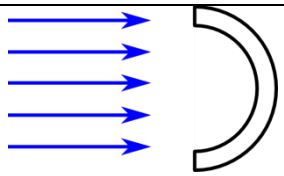
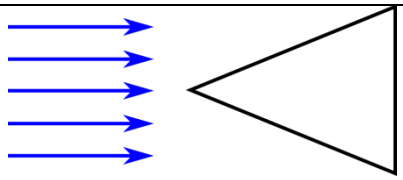


Figure 4.3 Objects with different shapes and surface areas can have the same frontal area depending on their design and the direction of flow.

4.2.1.1.2 Drag Coefficient

The drag coefficient is a dimensionless number that represents how aerodynamic an object is and how easily the fluid will flow past it, and is often determined experimentally. As opposed to the frontal area, this value is heavily dependent on the shape of the lateral cross section of the object, and not simply its projected area. Table 4.1 shows how shapes with the same frontal area can have very different drag coefficients.

Table 4.2 Comparison of the drag coefficient of several shapes with the same frontal area. All the shapes have the same circular frontal area. Data taken from [115] p127 and [116].

Description	Shape	Drag Coefficient
Sphere		0.47
Hemisphere pointing outwards		0.34
Hemisphere pointing inwards		1.33
Cone		0.34

While designing the robot, it will be necessary to ensure that the components are streamlined in order to minimise the drag coefficient. Unlike most other objects that need to travel through a fluid, the robot design will need to take into account the fact that the flow past it will primarily come from two directions, rather than one, which will affect how the robot is streamlined.

Although this design variable is important, it is expected that it will not have too much effect on the selection of the design concepts for this thesis. This is because, as mentioned previously in 1.3, the robot would not be tested in a wet environment at this stage, whereas drag will primarily come into play when optimising the detailed design to operate in fluid.

4.2.1.1.3 Flow Velocity

The fluid velocity is not something that the robot can control and is determined by the operator of the pipeline. However, the velocity of the flow inside a cylindrical pipe is not uniform. This is because at the interface between the fluid and the pipe wall, boundary layers are formed due to the friction between the fluid and the pipe wall. This reduces the fluid velocity at

the pipe wall, with the velocity gradually increasing towards the centre of the pipe. Figure 4.4 shows how the velocity profile of the fluid changes in relation to the distance from the surface.

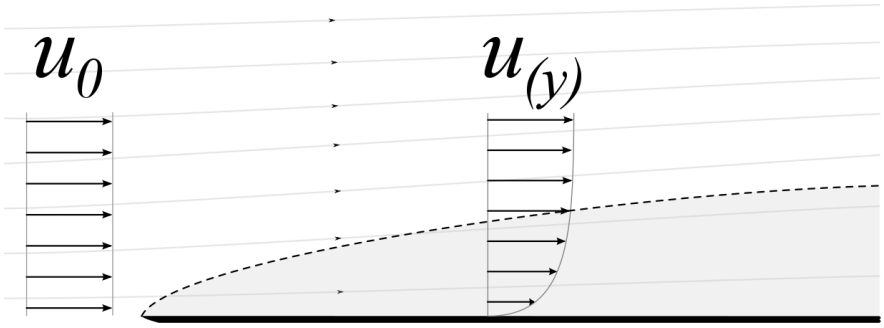


Figure 4.4 Change in fluid velocity depending on the distance from a surface [117].

As fluid with an initial velocity u_0 flows past a surface, the velocity of the fluid (indicated by the magnitude of the arrows) closest to the surface decreases. The fluid velocity increases as the distance from the surface increases.

This property can be taken advantage of during the robot design by focusing the mass of the robot towards the edges of the pipe rather than the middle so that the fluid flowing past it is at a lower velocity. Since the drag force is directly proportional to the velocity, this will contribute towards lowering the overall drag force.

4.2.1.2 Pressure Drop

The presence of the robot inside the pipe will reduce the volume that the fluid can occupy when flowing past, thus essentially acting as a restriction. The increased frictional losses, turbulence and other losses caused by the fluid flowing past the robot will decrease the energy in the fluid and thus its final pressure, resulting in a pressure drop across the robot, which will act on its frontal area and generate a force acting in the direction of the fluid flow. Figure 4.5 demonstrates how the robot can be represented by a restriction in the pipe.

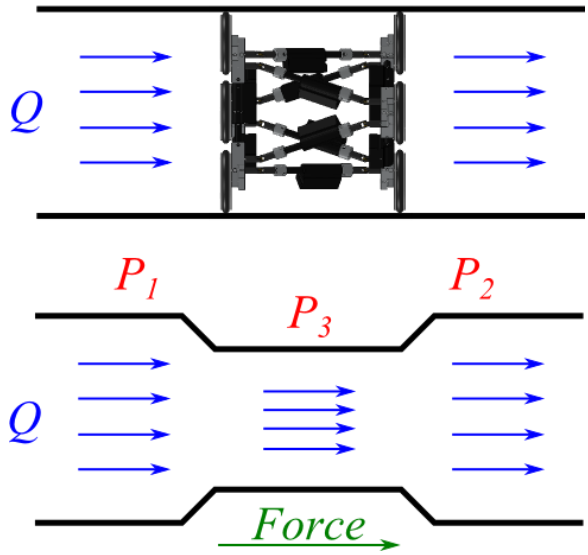


Figure 4.5 Representation of the robot as a restriction in the pipe.

If ideal conditions are assumed, a comparison of the fluid conditions both with and without the robot inserted can be modelled as shown in Figure 4.6.

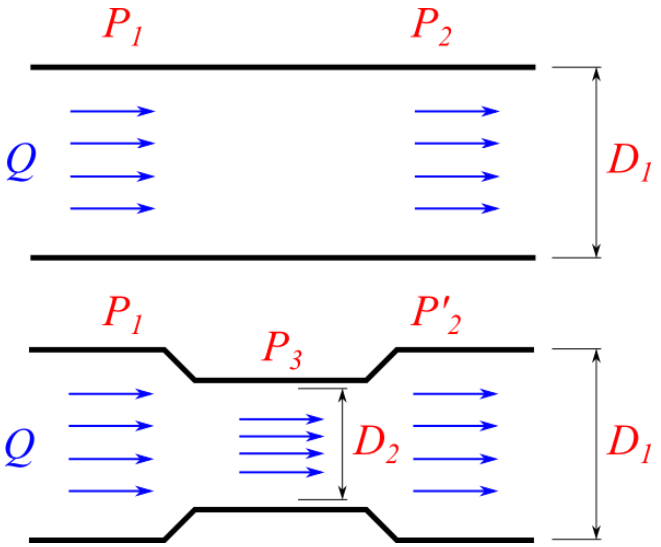


Figure 4.6 A comparison of the flows and pressures in a pipe with and without the robot inserted.

In the top half of Figure 4.6, a pressure differential, $P_1 - P_2$, across a straight pipe of diameter D_1 induces a volumetric flow Q through the pipe. In the bottom half, a restriction is inserted into the pipe with the same inlet pressure and flow.

According to Bernoulli's principle, an increase in the speed of the fluid is accompanied by a reduction in the pressure in order to keep the total energy in the fluid constant [118]. For fluid flowing in a pipe with a cross sectional area A at a fixed volumetric flow rate, Q, and an overall pressure differential of $P_1 - P_2$, a decrease in the cross sectional area of the pipe would result in

an increase in the fluid velocity, v. This can be seen in equation (3) below, where the velocity and area are inversely proportional to each other.

$$Q = Av \tag{2}$$

$$\frac{Q}{A} = v \therefore v \propto \frac{1}{A} \tag{3}$$

Therefore, as the speed increases in the narrow section, the pressure decreases as per Bernoulli's principle, resulting in a pressure P_3, which is lower than P_1. This is called the Venturi effect. After the fluid exits the restriction and the area increases, the fluid speed decreases back to its original value and likewise the pressure increases to P_2'. As no energy has been lost from the fluid, the pressure P_2' will match P_2. Therefore, treating the problem from an ideal standpoint, adding a restriction has no effect on the overall pressure drop across it.

In reality, the increased frictional losses, turbulence and other losses caused by the fluid flowing past the shape of the robot will reduce the energy of the fluid and cause the pressure P_2' to be lower than P_2. However, determining the magnitude of these losses would require simulation or practical experimentation, which, for the purposes of assessing design concepts, was deemed to be too complex.

By considering the problem intuitively, it can be deduced that the tighter the restriction (i.e. the smaller the value of D_2 relative to D_1), the higher the fluid losses as it goes through the restriction and therefore the higher the pressure drop across it. The pressure P_3 also decreases as D_2 decreases and therefore, it was decided to use the use the pressure P_3 to represent a worst case scenario for the pressure drop across the robot.

The pressure P_3 can be calculated using the Venturi equation, which is shown below, where A_1 and A_2 are the areas of the pipes at diameters D_1 and D_2 and ρ is the density of the fluid.

$$Q = A_1 \sqrt{\frac{2}{\rho} \cdot \frac{P_1 - P_3}{\left(\frac{A_1}{A_2}\right)^2 - 1}} \tag{4}$$

The equation can be rearranged to make the pressure difference the subject as follows:

$$P_1 - P_3 = \frac{\rho Q^2 (A_1 - A_2)}{2{A_1}^2{A_2}^2} \tag{5}$$

The fluid density, ρ, the flow, Q, and the pipe cross sectional area, A_1 are a property of the pipe environment and cannot be controlled. However, the effective restriction area A_2 can be controlled as this represents the reduced cross sectional area of the pipe due to the presence of the robot inside it and is a direct by-product of the robot's size and shape. Assuming all other parameters are fixed, the relationship between A_2 and the pressure difference can be seen in the graph in Figure 4.7.

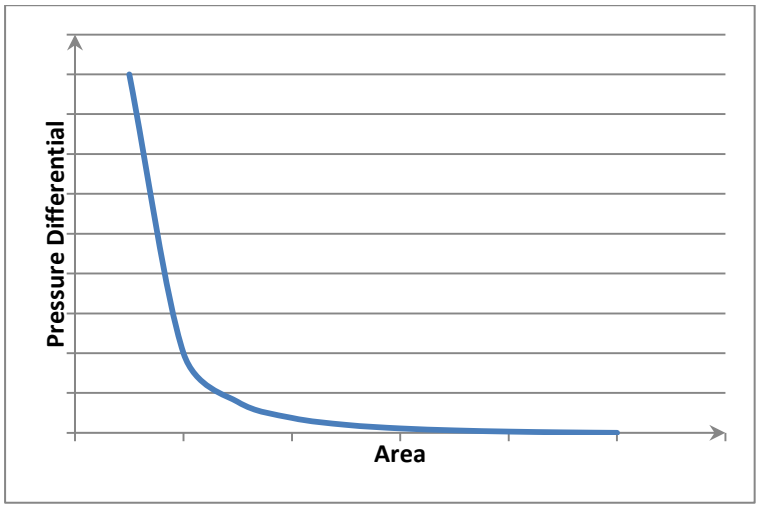


Figure 4.7 Relationship between the restriction area and the pressure difference in a Venturi.

As can be seen from the graph, increasing the cross sectional area at the restriction decreases the pressure differential at the restriction, thus reducing the force that would be applied to the robot. Therefore, minimising the amount of space the robot takes up inside the pipe will help reduce the overall force exerted on it.

4.2.2 Reliability and Simplicity

An important property that a pipe inspection robot must have is its ability to work reliably inside the pipe environment. Any failures would result in manual intervention in order to extract the robot from the pipeline, which could be complex and costly. If the robot does fail, it needs to be able to fail in a predictable fashion that allows it be recovered as easily as possible. For example, in the event of a sudden power failure in the robot, it would be better if the robot were able to maintain its position inside the pipe, rather than lose its grip and be washed away with the flow. A design that relies on the presence of power to maintain its shape would be less reliable than one that does not. The mechanical design of the robot will play an important part in determining how intrinsically reliable the robot will be in a failure situation.

The simplicity of the robot is also an important factor that can be used to determine the effectiveness of a particular solution. A robot with a simple design and fewer parts is an inherently more reliable and cheaper solution than a more complex design composed of many parts. A simple robot with few actuators is less likely to develop a fault than a more complex robot.

4.2.3 Wheeled vs Inchworm Locomotion

In the literature review, it was mentioned that wheeled locomotion is less efficient for climbing and that inchworm locomotion was more appropriate. This will be further elaborated in the following section.

The wheels on a climbing robot serve two purposes: the main purpose is to drive the robot through the environment. The second purpose is to hold the robot in place so that it does not fall.

When it comes to holding the robot in place, the wheels are pressed against the climbing surface in order to generate a frictional force that counteracts the force of gravity and any other forces that might be pulling the robot down. Figure 4.8 shows a free body diagram for a wheel being used for climbing, where P is the force pushing the wheel against the climbing surface, N is the reaction force and W is the weight of the robot pulling down.

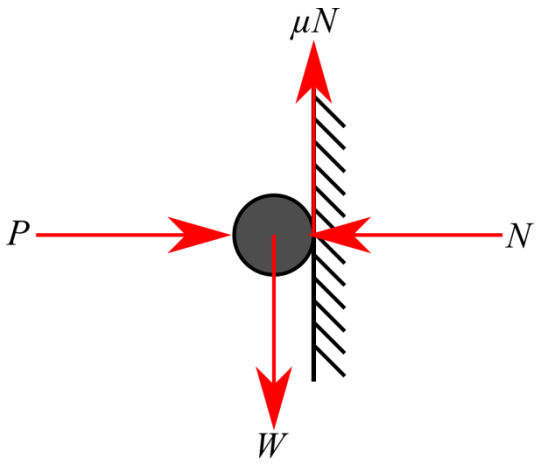


Figure 4.8 Wheeled climbing free-body diagram.

The contact between the wheel and the climbing surface produces a frictional force, μN, which is dependent on the coefficient of friction, μ, between the wheel and the climbing surface, as well as the normal force exerted on the wheel, N. The normal force is in turn dependent on the pushing force, P. This frictional force acts in the opposite direction to the robot's weight, holding the robot in place. If the weight of the robot increases, or another load is exerted on the robot (e.g. water flowing in the

opposite direction to the robot), the frictional force needs to be increased. The robot can only do this by increasing the force pushing the wheels against the climbing surface.

However, this force is also acting as a load on the wheels, making it more difficult to drive them. The rolling resistance of a wheel can be calculated as follows:

$$F_{rolling} = cN \quad (6)$$

Where c is the rolling resistance coefficient, and N is the same normal force present in Figure 4.8. Therefore, the higher the force pushing the wheels against the climbing surface, the higher the normal force and thus the higher the rolling resistance. Other frictional forces will also be affected by the increased load on the wheels, such as the increased friction inside the axle bearings of the wheels. Therefore, the increased load on the wheels means that on top of having to overcome the weight of the robot and any other loads (such as the fluid drag force), wheeled climbing robots also have to overcome the additional loading caused by gripping the climbing surface.

Inchworm robots do not suffer from this because the mechanisms used to grip the climbing surface and those used to propel the robot are often mechanically decoupled from each other. One set of actuators is used to fix part of the robot to the climbing surface, while another independent set pulls the rest of the robot up. The force that the gripping actuators exert does not affect the load experienced by the actuators pulling the robot up. This is apparent when the active inchworm-based pipe inspection robots described in section 2.5.2.2 are examined. This feature of inchworm-based climbing robots allows very high gripping forces to be generated without negatively affecting the rest of the robot, and makes climbing more energy efficient. For this application, where the robot will need to withstand the additional drag forces caused by the fluid flow, the ability to generate high gripping forces without negatively affecting the rest of the robot is very important.

4.3 Conceptual Design and Development

Once the main design criteria were determined, design concepts that met those criteria could be thought up. In this section, the development of the design concepts from the initial concept to the final idea is presented.

4.3.1 Single Ring Concept

The first concept involved a single mechanical “ring”, composed of prismatic and revolute joints that would be controlled using linear actuators and servo

motors. The motivation behind this was to keep as much of the robot's physical mass towards the edge of the pipe where the flow was slowest, in order to reduce the drag force, while trying to keep the robot as simple as possible. The concept is shown in Figure 4.9. The design consists of four linear actuators arranged in a diamond shape and connected together using actuated rotary joints. At the joints, rubber feet would be attached that would be used to grip the pipe wall.

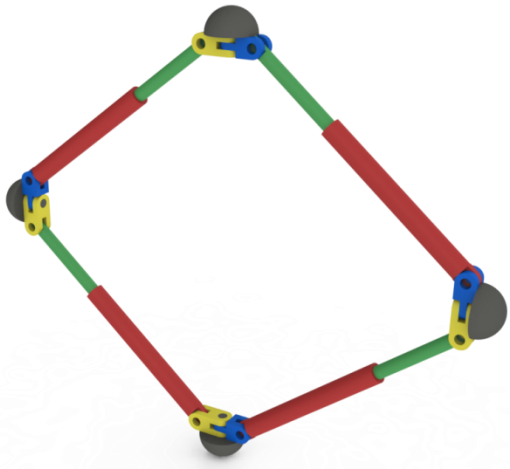


Figure 4.9 Single ring robot concept.

In order to move, the robot would move one of its contact points to a new location while the others hold the robot against the pipe. It is for this reason that a minimum of four contact points are needed so that at least three are holding the robot in place at any one time. An example of how the robot would move along in a straight pipe is shown in Figure 4.10.

There was concern regarding the stability of the robot when moving in this fashion. There are many points in the robot's motion where a contact point would be pushed against the pipe wall by one linear actuator while simultaneously being pulled away from the pipe wall by another linear actuator as it is contracting, which would reduce the robot's holding force. This is due to the fact that the same actuators used to push the ring outwards to hold the robot in place are also being used to move parts of the ring forward along the pipe, as shown in Figure 4.11. Also, positions 4-7 in Figure 4.10 highlight situations where the force pushing the contact point against the pipe wall is not directly perpendicular to the tangent of the pipe, which could lead to the contact point slipping sideways.

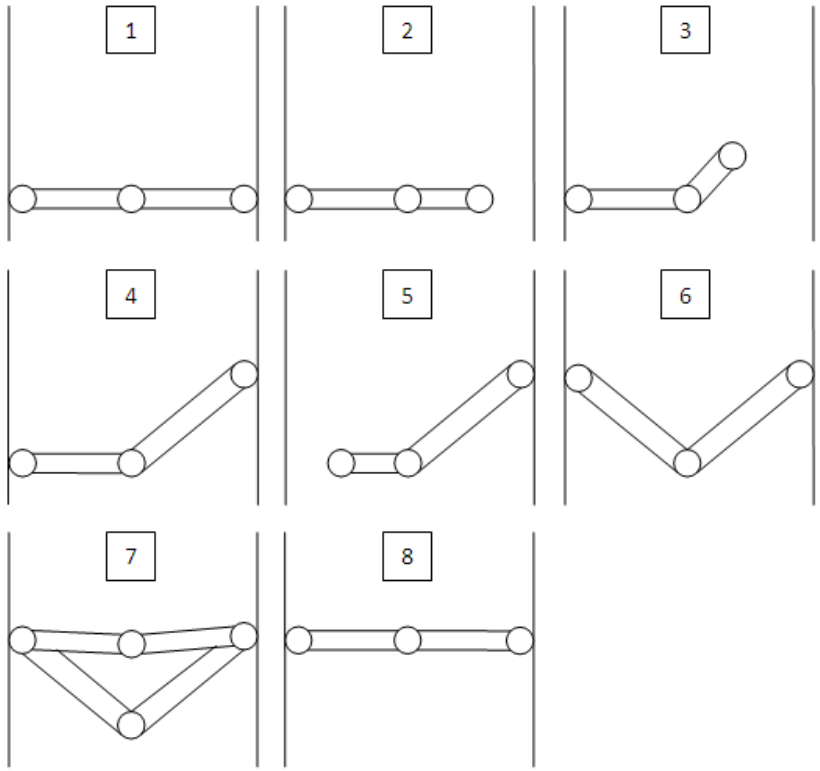


Figure 4.10 Motion of the single ring robot as seen from the top.

In this image, the robot starts off fully expanded into the pipe (1). It then contracts one of its corners (2) and moves it forward (3). The corner is then re-extended to grip the pipe wall (4). The same process is then repeated for the opposite corner (5-6), and then for the top (7) and bottom (8) corners.

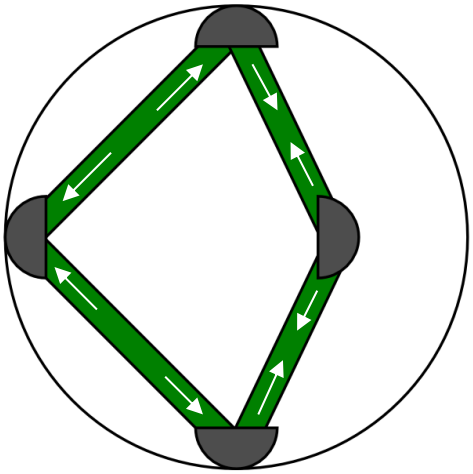


Figure 4.11 Forces acting on the single robot when retracting a corner.

With the contact point on the right being pulled in, opposing forces are acting on the top and bottom contact points, which could lead to slip. The white arrows show the direction of the forces exerted by the linear actuators.

It would be more efficient and reliable to have the linear actuators acting perpendicular to the tangent of the pipe wall and have other actuators acting perpendicular to that in order to move the robot, so that the two mechanisms are decoupled from each other. That way, the benefits of inchworm locomotion during climbing as described in section 4.2.3 would be realised. It would not be possible to achieve this using the single ring design and so the concept of two rings was investigated.

4.3.2 Double Ring Concept

With the double ring concept, two mechanical rings are connected together using a series of linear actuators. Unlike the single ring concept, all the contact points of one ring are used to hold the robot in place while the actuators between the rings move the other ring to a new location. This can provide a more stable and reliable locomotion method since the linear actuators forming the rings are only ever pushing the contact points against the pipe wall and, as the linear actuators between the rings are perpendicular to them, their forces do not compromise the holding force. Since the stabilising ring will always have all of its contact points in contact with the pipe wall, the ring can be reduced to a triangle rather than a diamond, in order to keep the number of actuators to a minimum.

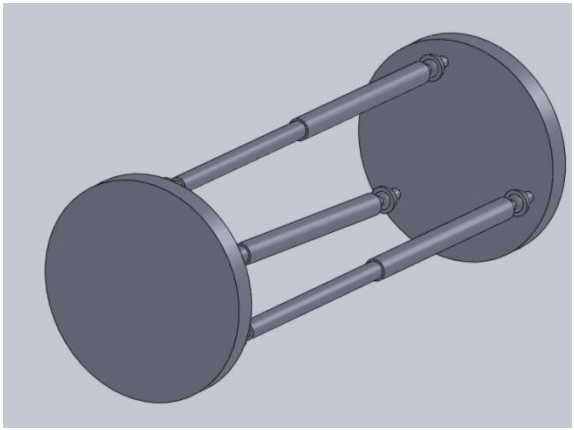


Figure 4.12 Double ring concept

The solid discs in the diagram would be made from three linear actuators and would allow the discs to expand and contract.

One of the main design questions related to the double ring robot is the positioning of the actuators between the rings. Ideally, the actuators should be placed as far out towards the edge of the ring as possible. This is so that they can generate as much torque as possible when rotating the rings, such as when navigating corners. However, doing so will prevent the robot from navigating corners because the linear actuators will collide with the pipe wall as shown in Figure 4.13. Moving the actuators in towards the centre of the

robot solves this problem, although there are still limitations regarding the sharpness of the corner that the robot is able to navigate. The further that the actuators are moved in towards the centre of the robot, the sharper the corners that the robot can navigate. With the actuators more towards the edge, the robot can only navigate gentle corners and would need to navigate sharp corners in several steps. Alternatively, the linear actuators between the rings can each be replaced with two linear actuators connected together using an actuated joint. This allows the robot to bend its body around the corner, without reducing the torque that can be applied to the rings. However, since this requires more actuators, both the cost and complexity of the robot would increase.

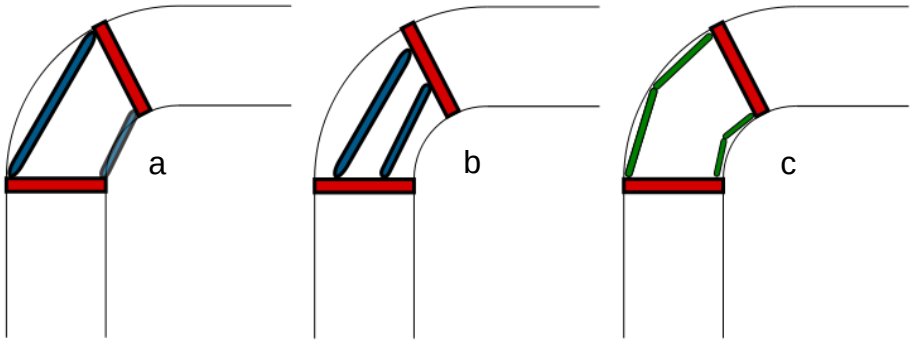


Figure 4.13 Linear actuator configurations used to avoid collision with the pipe wall.

In a), the robot's actuators are at the outer edge of the ring. When navigating a bend, the actuators would collide with the pipe wall. This can be avoided by moving the linear actuators inwards (b) or making them segmented (c).

Another design question relates to the joints connecting the linear actuators between the rings to the rings themselves. In order to allow the robot to manipulate the rings in 3D space, 2 DOF joints are needed between the linear actuators and the rings. However, if the joints are left unactuated, the robot would be underconstrained and would not be able to maintain its shape under the forces exerted on it by the water flowing past and could collapse into an unrecoverable state. Some form of actuation needs to be applied to the joints, either actively, such as through servomotors, or passively by inserting springs into the joints in order to assist them in returning to their original positions. Once again, this increases the robot's complexity.

4.3.3 Cube Inchworm Concept

An alternative to the double ring design described above was investigated, where the robot is in the shape of a cube with a linear actuator used in place

of the cube's edges. This design is very similar to the one demonstrated by [83]. The advantages of such a design in their chosen environment is that the robot is capable of Cartesian motion and so can navigate bends and junctions simply by moving straight into them and then immediately changing direction and moving out of them, as shown in Figure 4.14. This can significantly simplify the control algorithms of the robot and possibly its mechanical design as it does not need to twist or rotate its body in order to navigate the different pipe features.

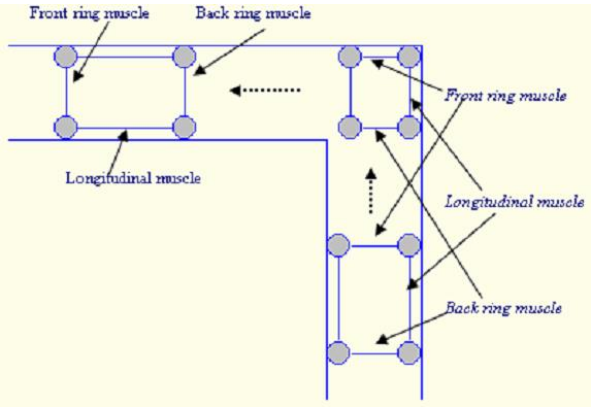


Figure 4.14 Inchworm robot developed by [83] navigating an L-shaped junction.

However, this simplified motion would only be possible in pipe networks where the bends have sharp corners. In most pipe networks, this is not the case and the corners are formed from swept bends. A robot navigating such corners would still need to be able to rotate its body in order to move through the bend and thus would require the necessary degrees of freedom to do that. However, the Cartesian motion would still be useful when navigating junctions. Navigating junctions is expected to be more complex than navigating bends since in a bend, the robot is entirely surrounded by the pipe wall and so could use it as a guide to assist in navigating the bend. However, in a junction this is not the case, which could require different motion plans to those used for navigating corners. If the robot is capable of Cartesian motion, this would greatly simplify navigation through junctions.

As with the double ring design, the cube ring robot will also suffer from the problem of its linear actuators colliding with the pipe wall during bends. Again, this can be solved by moving the actuators in towards the centre of the robot, or by adding joints to the cube's edges, allowing it to bend. Moving the actuators in towards the centre is a simpler method of doing so and can be accomplished by adding posts at the corners of the cube as shown in Figure 4.15. A 3D CAD model of the cube robot is shown in Figure 4.16.

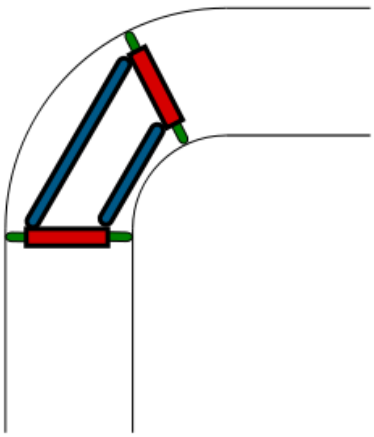


Figure 4.15 Posts added to the corners of the cube allow it to navigate corners.

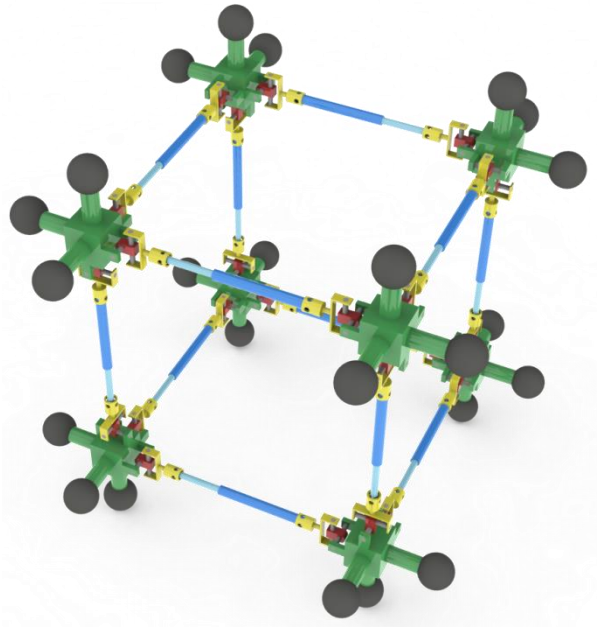


Figure 4.16 3D Model of the cube inchworm.

In order to allow the different sides of the robot to expand and contract, joints need to be added between the linear actuators and the corners. Initially, 2 DOF joints were used, giving each corner a total of 6 DOF. However, further examination of the mechanism revealed that this led to many duplicated degrees of freedom. For example, examining the corner shown in Figure 4.17 shows that two of the joints share the same vertical axis, albeit offset from each other.

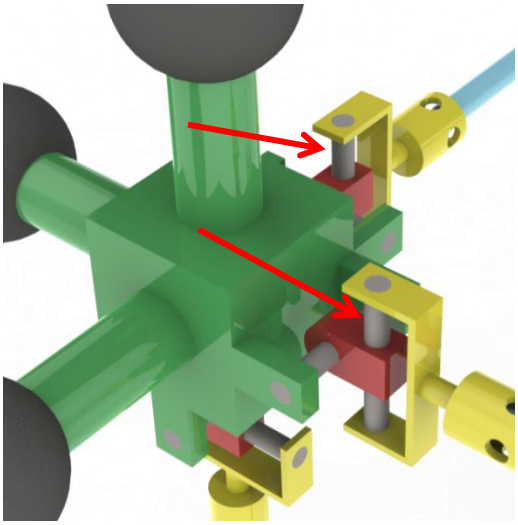


Figure 4.17 Examination of the joints at the corners reveals duplicate degrees of freedom.

Here, the axes of the two joints indicated by the arrows are in the same direction, resulting in redundancy.

Further optimisation of the joints led to a second design concept where the degrees of freedom were reduced to three at each corner. This is shown in Figure 4.18.

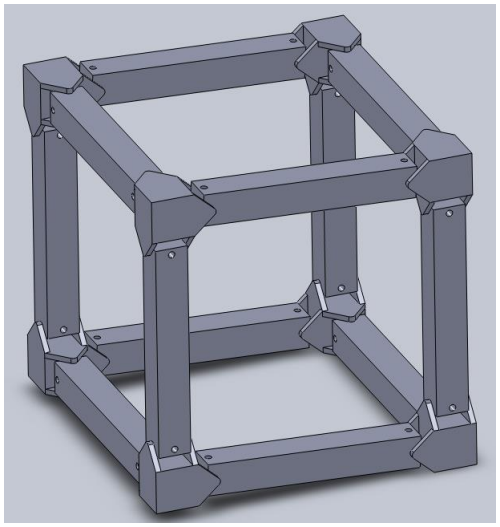


Figure 4.18 Simplified cube inchworm concept.

The joints were configured so that the two joints on each edge have the same rotation axis. This requires the robot to actuate an integer multiple of two adjacent and parallel linear actuators in order to produce motion. Although both designs can perform similar motions, the reduction in degrees of freedom in the simplified design leads to some differences in the orientation of the corners during motion. This can be seen in Figure 4.19. During certain motions, the corners are forced to rotate to remain parallel to the linear actuators. Since the corners are also the contact points used to

hold the robot against the pipe wall, such motion would lead to the robot shifting its position in the pipe and may lead to instability.

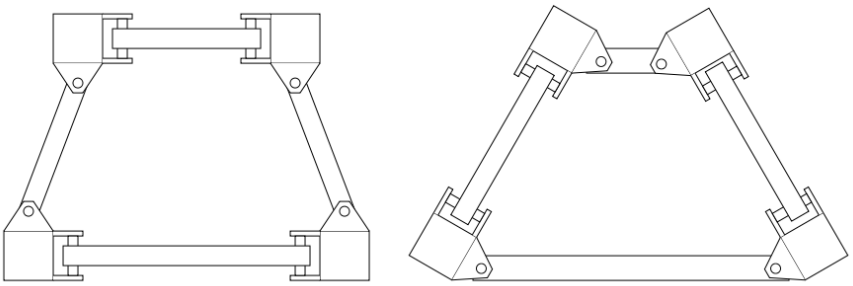


Figure 4.19 The reduction in DOF forces the corners to rotate during certain motions.

Another issue with this design is the very large number of actuators that needed to control it. On top of the 12 linear actuators that would be present along each of the edges, the robot requires 3 rotary actuators per corner in order to constrain them, resulting in a total of 36 actuators.

4.3.4 Gough-Stewart Concept

Due to the problems faced by the previous concepts, further conceptual design work was carried out in order to find a suitable design. The final design concept that was considered was based on a Gough-Stewart platform and was inspired by the work presented in [85], [86]. The design is based on a normal Gough-Stewart platform, but rather than the top and bottom platforms being fixed, they are composed of a “ring” of linear actuators, allowing their size to change. The concept can be seen in Figure 4.20.

The use of a Gough-Stewart platform provides the benefits of the previous concepts while solving the problems that they face. As with all of the ring-based designs, the majority of the robot’s mass is concentrated towards the edge of the pipe where the flow is slowest, thus helping to minimise the drag force. Its “skeletal” nature means that it does not take up much space inside the pipe and has a low frontal area, further minimising the drag force. As the design is also a double-ring design, the actuators that grip the pipe wall act perpendicularly to the ones pushing the robot forward, decoupling the two. The robot’s ability to navigate pipelines of varying diameters will be primarily dependent on the extension ability of the linear actuators. Using linear actuators with high extensions will increase the diameter range that the robot is capable of navigating.

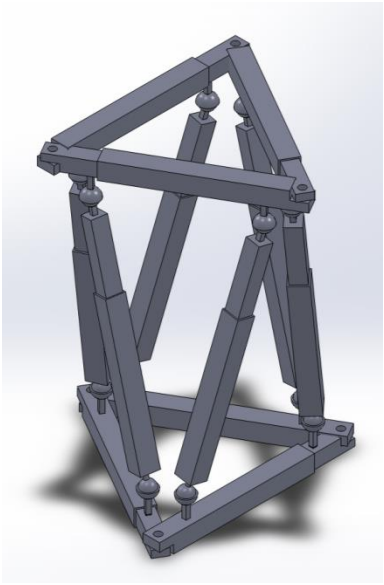


Figure 4.20 Gough-Stewart-based concept.

Unlike the previous two concepts, the number of actuators required by the design to be fully constrained is comparatively low, with only 12 linear actuators needed (6 for the Gough-Stewart platform itself, and 3 for each of the top and bottom rings). The Gough-Stewart platform itself is able to move the top and bottom rings with 6 degrees of freedom, which should give the robot enough flexibility to traverse the pipe environment and to manipulate tools and instruments. An additional benefit over the other two concepts is that there is only one type of actuator needed, which would simplify the robot's control system.

Due to these advantages, the Gough-Stewart-based concept was chosen for the final design.

4.4 Summary

In this chapter, the conceptual design process was presented.

First, the requirements and specifications that were laid out in the previous chapter were converted from abstract requirements into practical design criteria that could be applied to the conceptual design process. Each of these design criteria was examined in order to understand how it affects the overall design and how it should be optimised in order to meet the requirements. The conceptual design process was then carried out by considering various design ideas in light of the requirements and then determining what problems each of the concepts faced.

The conceptual process started with the single ring design, which aimed to minimise the drag force by keeping the mass of the robot at the edge of the

pipe. However, there were concerns regarding how well the robot would be able to grip the pipe wall during movement.

The concept was then upgraded to use two rings with linear actuators connecting between them. This concept decoupled the gripping actuators from the actuators propelling the robot forward, improving the robot's ability to grip the pipe wall. The mechanical changes that would need to be made to the robot in order to navigate bends were explored. This could be done by either using actuated rotary joints to allow the two halves of the robot to rotate relative to each other, or the linear actuators between the two rings could be moved inwards in order to prevent the robot colliding with the pipe wall during bends. The main issue faced with this design was the uncertainty regarding how the linear actuators would be connected to the rings and that those joints would likely need to be actuated. This would increase the robot's complexity.

The cube inchworm concept was then explored, inspired by work that was seen during the literature review. This design used a set of linear actuators arranged in a cube fashion. The ability of the robot to move in a Cartesian fashion was expected to simplify the robot's ability to navigate junctions. An initial design concept was modelled and then further simplified to eliminate redundant degrees of freedom. The main issue faced by this design was the instability of the gripping mechanism, because the robot's corners could change their positions during certain motions. Another issue is the large number of actuators that the robot would require in order to be fully constrained.

The final concept was the Gough-Stewart-based design, which was also inspired by work found in the literature review. This concept is based on a standard Gough-Stewart parallel platform but with the top and bottom platforms replaced with linear actuators, allowing their size to be changed. This concept solved the problems faced by the other concepts: the use of a double-ring design allows the gripping mechanism to remain stable, and the unique arrangement of the linear actuators in the Gough-Stewart platform allows 6 DOF of movement using a relatively small number of actuators. The mechanical design allows the robot to be fully constrained using only linear actuators, simplifying the robot's control system. It was therefore decided to base the robot's final design on this concept.

Chapter 5
Detailed Design

5.1 Introduction

In the previous chapter, the various conceptual designs were presented, analysed and assessed. A final concept, based on a Gough-Stewart platform, was deemed to be the most suitable. This chapter presents the steps taken to implement the detailed design based on that concept. The design is split into two main sections: the mechanical design and the electronic control design.

The mechanical design focuses on the design and selection of the various mechanical components that form the robot's structure. This involves the selection of the linear actuators based on calculations determining their required force and working length, and the design and selection of the various joints and connecting components used to form the robot's structure.

The electronic design focuses on the design of the electronic circuitry that is used to control the linear actuators and take measurements from them. This involves the selection of an off-the-shelf motor controller, along with identifying its shortcomings and designing additional circuitry that addresses them.

5.2 Robot Overview

The mechanical structure of the robot is formed around a Gough-Stewart platform composed of six linear actuators along with various connecting parts. The platform's top and bottom rings are each formed from three linear actuators and connecting parts. On each ring there are three support discs placed at the corners.

In order to control the robot, each motor is connected to an off-the-shelf, dedicated motor controller board. A custom-designed auxiliary board provides high resolution current measurement and also allows the motor controllers to communicate over an RS485 bus.

5.3 Mechanical Design

The mechanical design process involved the following steps:

- Actuator selection;
- Structural component design and selection;

- Support disc design.

5.3.1 Actuator Selection

The linear actuators form the primary structural foundation around which the rest of the robot is built. Therefore, they were the first component that was determined in the design. The choice of linear actuator was primarily limited by two design factors: the required size of the robot, and the required force that the robot would need.

5.3.1.1 Actuator Size

The actuator size was determined by the size of pipe that would be used for testing. It was decided to use a 200mm outer diameter/194mm inner diameter pipe as it falls within the range of pipe diameters used in the water industry, was widely available and was reasonably large, thus avoiding the complications of having to manufacture a miniature robot. Therefore, the robot's top and bottom rings would need to be designed so that they can fit inside a 194mm pipe. The diagram below shows a representation of the pipe diameter and the robot ring.

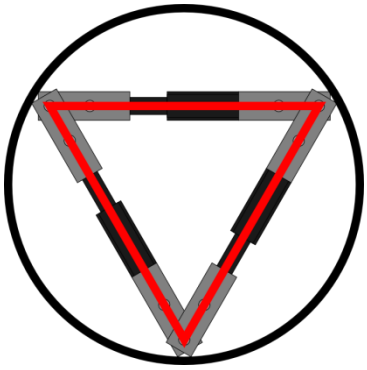


Figure 5.1 Representation of the robot ring inside the pipe.

The robot's ring can be approximated as an equilateral triangle inside the pipe.

If the robot's ring is considered to be an equilateral triangle touching the pipe's internal surface, the circular pipe cross section would represent the circumcircle of the triangle. The relationship between the length of the triangle side S and the diameter of the circumcircle d is given by the formula:

$$d = \frac{2S}{\sqrt{3}} \tag{7}$$

$$\therefore S = \frac{d\sqrt{3}}{2} = \frac{194\sqrt{3}}{2} = 168mm \tag{8}$$

Since the ring sides would be formed of mechanical connecting components as well as the actuators, the actuator size would need to be less than the

maximum of 168mm. In order to leave enough room for the mechanical connecting parts, it was decided to search for actuators whose working range fell within approximately half that value, at 80-90mm.

5.3.1.2 Actuator Force

The force produced by the linear actuators needs to be high enough to ensure that the robot is capable of propelling itself through a pipe, maintaining its position inside a pipe and overcoming any forces exerted on it from the flow of fluid. It was expected that the static forces exerted on the robot would be the primary forces that would need to be overcome, with dynamic forces probably being small in comparison. This was because the functional prototype would not be tested in a wet environment, and the use of leadscrew-driven linear actuators would mean that the robot's movement speed was expected to be low. Therefore the static forces were the only forces considered when selecting the actuators.,

It was estimated that the situations in which the actuators would be required to output the highest force would be one of the following:

- For the ring actuators: the force exerted when pushing out against the pipe wall in order to hold the robot in its current position.
- For the Gough-Stewart actuators: the force exerted when pulling or pushing the robot along its longitudinal axis against gravity, with one of the rings braced against the pipe wall.

A brief static analysis was performed in each case to estimate the maximum force required.

5.3.1.2.1 Ring Actuators

The ring actuators are used to force the ring against the pipe wall, generating a frictional force that holds the robot in place. The maximum force required is when the gravitational force is acting along the robot's longitudinal axis and only one ring is supporting the robot. The total force would therefore be distributed equally amongst the ring's three contacts points. Figure 5.2 shows a free body diagram of the forces acting on one of these contact points.

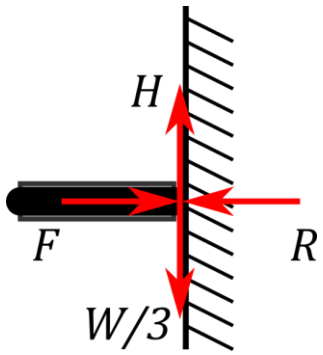


Figure 5.2 Free body diagram showing the forces acting at the contact point between the pipe and the robot's ring.

The holding force, H, that holds the robot against its own weight, W, is generated by the linear actuator force pushing the disc against the pipe wall, F, and the coefficient of friction between the rubber on the support disc and the pipe wall, μ, as governed by the following equations:

$$F = R \tag{9}$$

$$\frac{W}{3} = H = \mu F \tag{10}$$

In order to estimate the required force, both the weight of the robot and the coefficient of friction between the ring and the pipe wall need to be estimated. As the robot was not yet designed and constructed, it's weight was estimated to be approximately 1kg, based on the weights of other robots whose sizes were similar to the expected robot size. Regarding the coefficient of friction, since that is normally determined experimentally, a sample coefficient of friction of 0.5 was used (based on a pipe wall of plastic against rubber [119]). The actuator force is calculated as follows:

$$\frac{W}{3} = \mu F \therefore \frac{1 \times 9.81}{3} = 0.5F \tag{11}$$

$$F = \frac{9.81}{3 \times 0.5} = 6.54N \tag{12}$$

Since two actuators push each support disc, the holding force is distributed amongst them according to their mechanical arrangement, as shown in Figure 5.3.

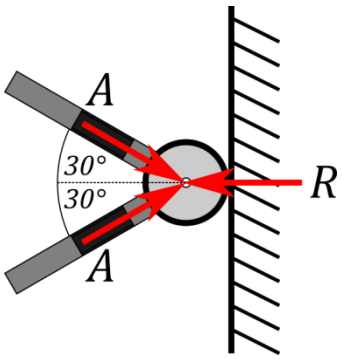


Figure 5.3. Free body diagram showing how the forces from two actuators combine to form the total holding force.

The force required by each actuator is calculated as follows:

$$R = 2A\cos(30) \tag{13}$$

$$A = \frac{R}{2\cos(30)} = \frac{6.54}{2\cos(30)} = 3.78N \tag{14}$$

The required ring actuator force is therefore 3.78N.

5.3.1.2.2 Gough-Stewart Actuators

The Gough-Stewart actuators are used to move the rings through the pipe to their next holding position. The maximum force exerted on them will be when they are lifting the robot up in a vertical pipe against the force of gravity. The actuators would be lifting approximately half the robot's weight, and so based on the previous estimate of 1kg weight, the actuators would need to lift a total of 500g, resulting in an approximate weight of 5N.

Since the Gough-Stewart actuators are almost always at an angle to the robot's longitudinal axis, the force needs to be resolved through that angle in order to determine the actual force applied to the actuator. The nature of a Gough-Stewart platform means that the angle that the linear actuators form with the vertical is not fixed, but rather it changes continuously throughout the motion of the platform. The actuators would need to exert the most force to lift the robot when their angle to the vertical is largest. When moving up a straight pipe, it was realised that the actuators form the largest angle when one ring is fully extended, the other is fully retracted and the Gough-Stewart actuators are fully retracted, as shown in Figure 5.4. For this calculation, the angle of the actuators was measured from the conceptual 3D model of the robot shown previously in Figure 4.20. This resulted in an angle of 24°.

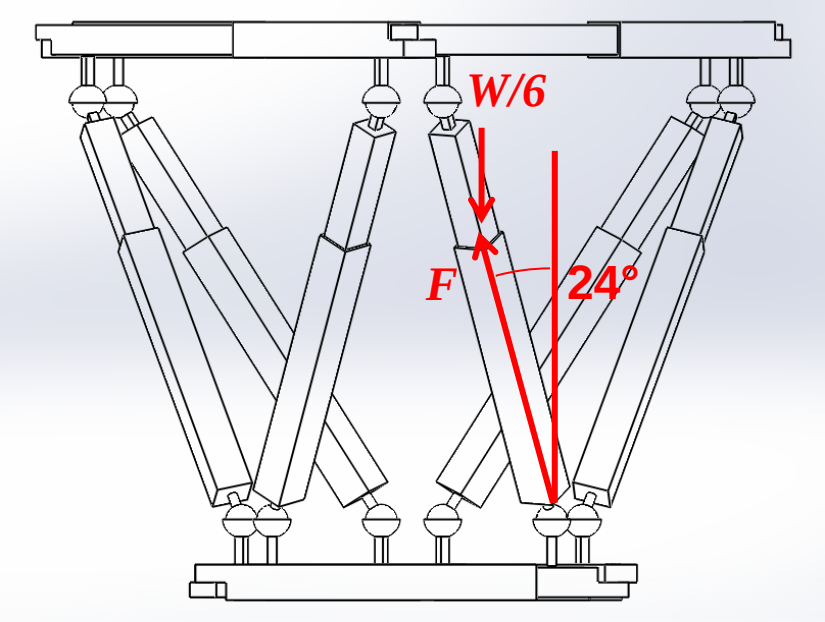


Figure 5.4 Robot arrangement that produces the maximum angle from the vertical for the Gough-Stewart actuators (shown here exaggerated).

Based on this information, the force required by the Gough-Stewart actuators can be calculated as follows:

$$\frac{W}{6} = F\cos(24) \tag{15}$$

$$F = \frac{W}{6\cos(24)} = \frac{5}{6\cos(24)} = 0.912\text{N} \tag{16}$$

The force required by the Gough-Stewart actuators is significantly less than the force required by the ring actuators to hold the robot in place (3.78N), and as such the ring actuator force will be used when choosing a linear actuator.

5.3.1.3 Linear Actuator Selection

From the above estimates, the important criteria for selecting the linear actuators were determined. These are as follows:

- The total working length must encompass the range 80-90mm.
- The actuator force must be at least 3.78N.

Based on these two criteria, a search was undertaken to identify suitable actuators for the prototype. The search focussed primarily on the miniature linear actuators from Firgelli Technologies. After searching through their product range, it was initially decided to use the L12-50-210-12-P, which provided a working length of 94-144mm and 45N of force. However, due to lack of availability at the time, the PQ12-63-12-P (36.5-56.5mm and 20N of

force) was purchased instead. Although this actuator had a smaller working range than the previous one, this could be compensated for by increasing the size of the mechanical connecting components that form the ring.

Figure 5.5 Firgelli PQ12 Linear Actuator.

5.3.2 Structural Components

With the linear actuators selected, the design and selection of the remaining structural components could be undertaken. The remaining components were as follows:

- The joints connecting the Gough-Stewart actuators to the top and bottom rings.
- The connecting parts between the Gough-Stewart actuators and the aforementioned joints.
- The connecting parts between the ring actuators that complete the ring structure.
- The support discs mounted on the top and bottom rings.

Special care was taken when designing the parts to ensure that they are simple to manufacture using conventional machining equipment.

5.3.2.1 Gough-Stewart Joints

The Gough-Stewart platform requires at least a 2 DOF joint and a 3 DOF joint on either end of each actuator, which connect it with the top and bottom rings. Suitable joints that would fulfil these requirements are spherical joints (3 DOF) and universal joints (2 DOF). Despite the availability of both joints, it was found that the spherical joints tended to have a smaller range of working angles than the universal joints. It was therefore decided to use only universal joints. In order to compensate for the loss of 1 DOF when a universal joint is used instead of a spherical one, one of the universal joints on each actuator would not be rigidly joined to its corresponding ring, but rather would be connected via a revolute joint, which replaces the missing

DOF. The addition of the revolute joint is equivalent to adding an extra link between each actuator and that joint. Despite this, the total number of degrees of freedom remains the same as before, at 6 DOF. This is shown in the DOF calculation below:

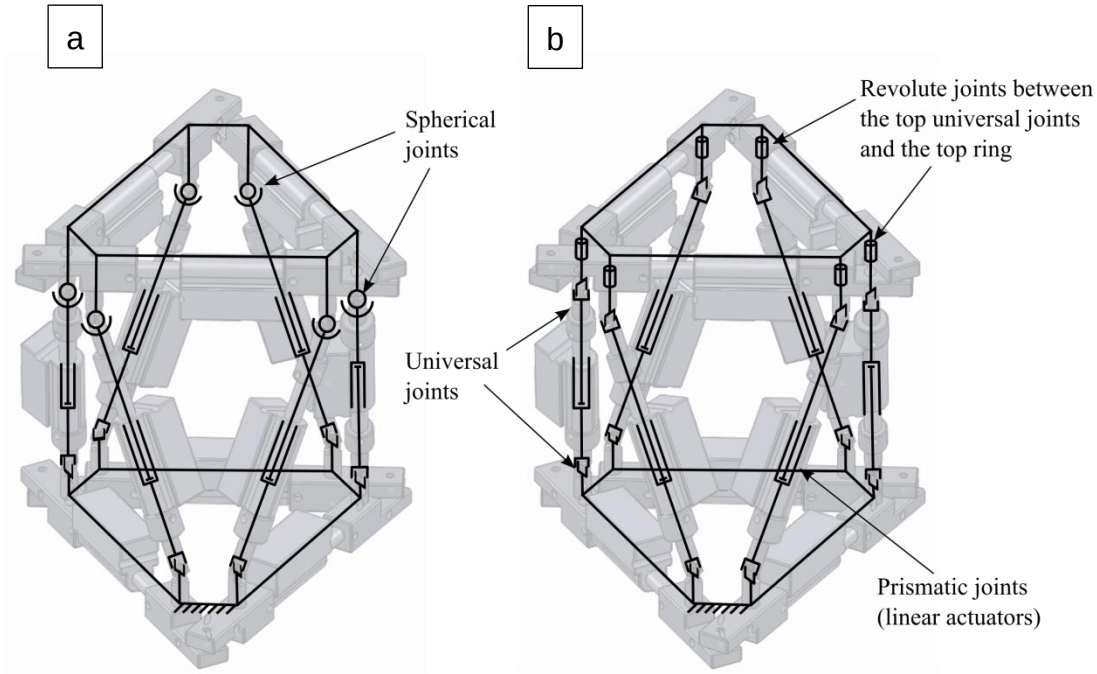


Figure 5.6 Comparison of the original and modified kinematic design.

a) Original kinematic design using spherical and universal joints; b) modified design using only universal joints.

Table 5.1 DOF calculation using universal joints instead of spherical joints.

Number of links (excluding ground link)	19
Total DOF	19 x 6 = 114
Number of 1 DOF joints	12 (6 prismatic, 6 revolute)
Number of 2 DOF joints	12 (universal joints)
Total Constraints	12 x 5 + 12 x 4 = 108
Total DOF	114 – 108 = 6 DOF

5.3.2.2 Actuator-Universal Joint Links

These parts are used to connect both ends of the linear actuators to the universal joints. On one end of the part is a rod that slides into the bore of the universal joint and is fixed in place via a grub screw. At the other end is a drilled hole that allows the mounting point of the linear actuator to be pushed

inside. Another hole drilled across the part allows an M3 screw and nut to hold the linear actuator in place. As each mounting point on the linear actuator has a different size, two different parts had to be made. Figure 5.7 shows the parts along with how they are assembled.

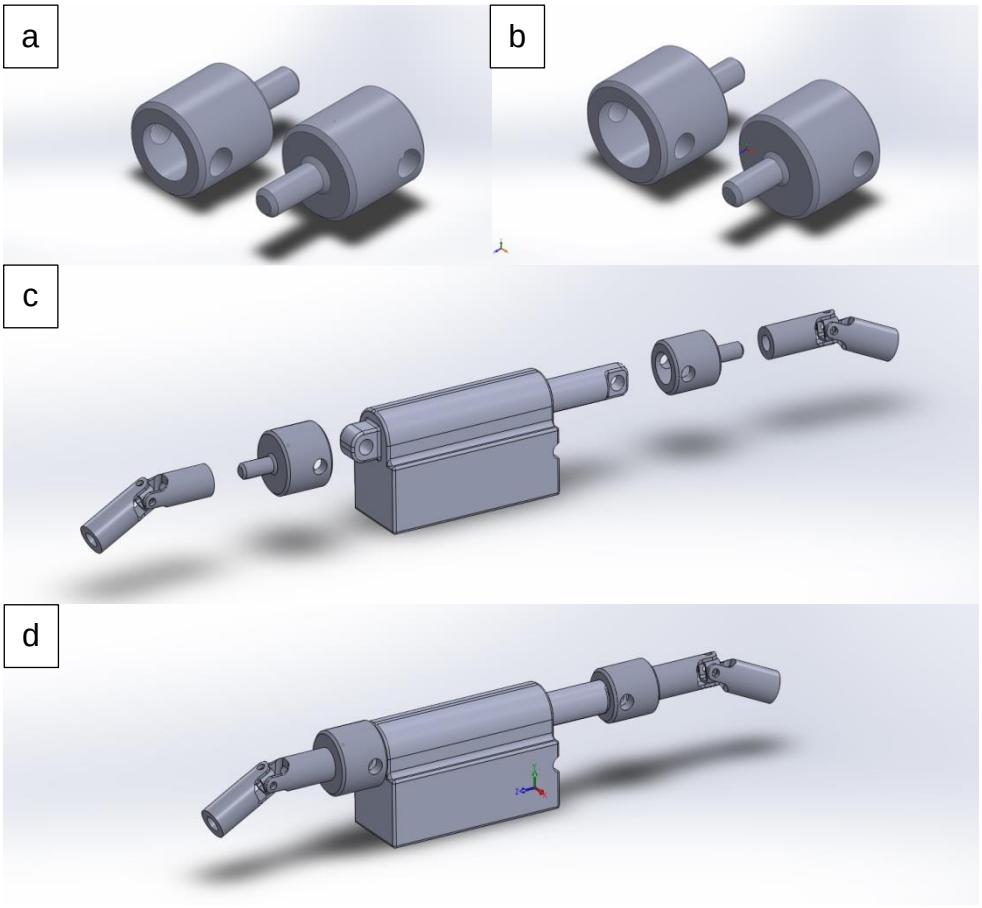


Figure 5.7 Mechanical parts used to connect the Gough-Stewart actuators to the universal joints.

a) Small U-joint adapter; b) large U-joint adapter; c) exploded view of the Gough-Stewart assembly; d) collapsed view showing the Gough-Stewart assembly.

5.3.2.3 Ring Connecting Parts

The parts that connect the linear actuators together to form the ring serve three functions:

- They connect the actuators together in a triangular shape with revolute joints at each corner.
- They allow the Gough-Stewart actuators to be connected via their universal joints.
- They allow the support discs to be mounted at each corner.

The designed parts fulfil all of these requirements while being mechanically simple. As with the Actuator-Universal joint links, one end of the part

consists of a drilled hole that allows the linear actuator mounting point to be pushed in and fixed in place with a nut and bolt. A series of holes drilled into various locations at the part allow the universal actuators to be mounted underneath the ring via an M3 screw, and allow revolute joints to be formed at the corners using another M3 screw as a hinge. Figure 5.8 shows the parts along with how they are assembled.

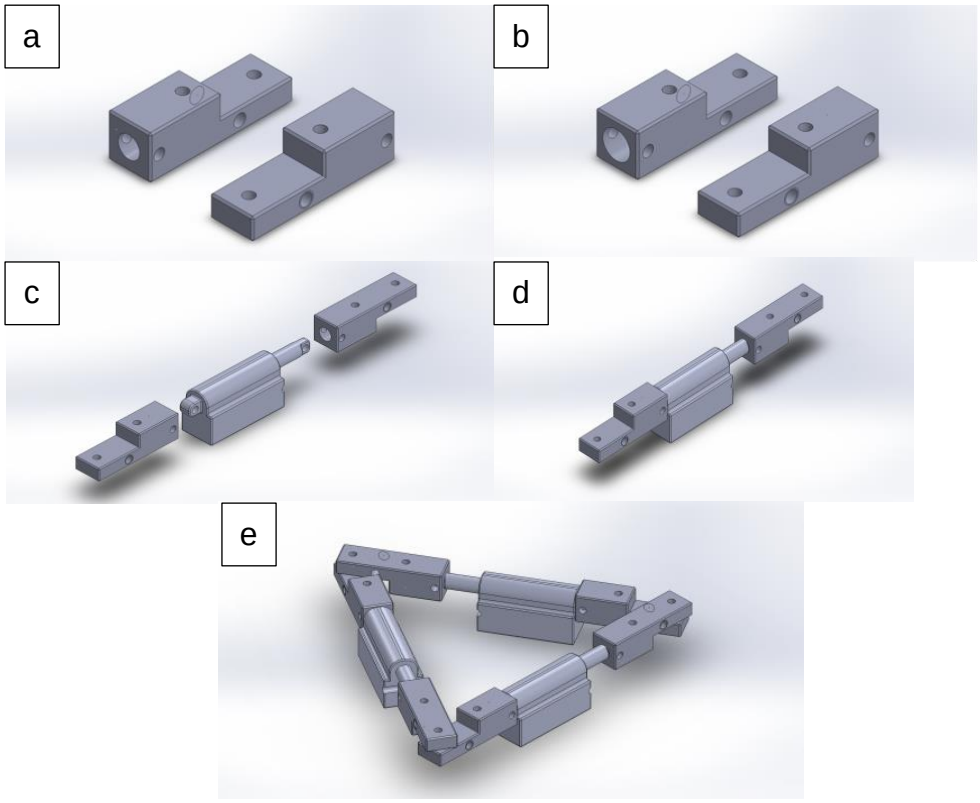


Figure 5.8 Mechanical parts used to form the top and bottom rings.

a) Small ring joint; b) large ring joint; c) ring side assembly exploded; d) ring side assembly collapsed; e) full ring assembly.

5.3.3 Support Discs

The purpose of the support discs is to allow the robot to navigate bends in the pipe without the Gough-Stewart actuators colliding with the pipe wall. As described previously in section 4.3.2, if the robot is designed with the linear actuators positioned at the outermost edges, it will not be able to navigate bends in the pipe as the actuators will collide with the pipe wall. The simplest solution involved moving the linear actuators in towards the centre of the robot, and this is the solution implemented in this design. By adding a set of support structures to the robot's exterior, the robot's overall size is increased, which effectively moves the linear actuators in towards the centre of the robot.

When initially designing the support discs, it was decided to use a set of rubber-tipped protruding rods arranged around the rings, as shown in Figure 5.9. However, this design faced two problems. The first was that a suitable, off the shelf part could not be found and manufacturing such a part with the facilities available was not deemed to be trivial. The second was that it was immediately apparent that this design would inflict significant bending moments on the linear actuators as the rings expanded and braced themselves against the pipe wall, as demonstrated in Figure 5.9. This is because the rods would be perpendicular to the linear actuators and thus cannot transmit the forces directly along the actuators' longitudinal axes. Such bending moments could potentially damage the linear actuators as they are not designed to withstand such moments.

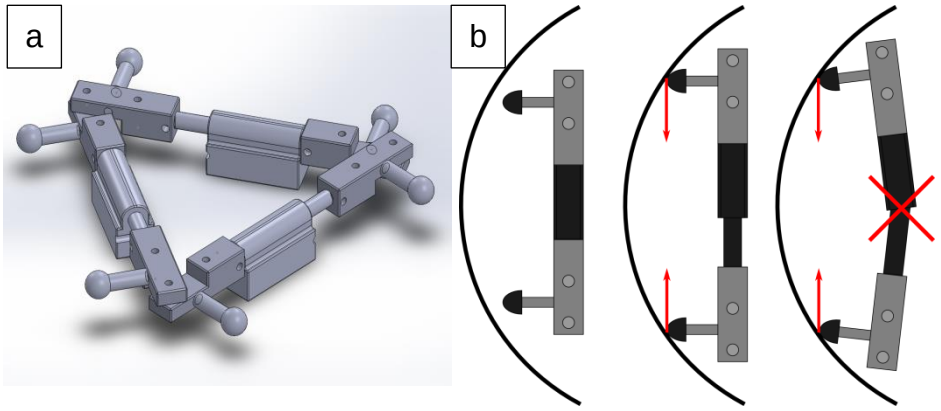


Figure 5.9 Robot ring with rubber stops.

a) Robot ring with rubber stops; b) the use of rubber stops generates a large bending moment that could damage the actuators.

An alternative solution was found, which involved placing a rubber rimmed disc at the corners of each ring, as shown in Figure 5.10. The benefits of this method are that the discs are simple to manufacture using a lathe; the rubber grip around the rings can be accomplished using a conventional O-ring; allowing the discs to spin would give the robot the ability to "self-centre" inside the pipe; and the reaction forces transmitted from the pipe wall to the robot are transmitted directly along the longitudinal axes of the linear actuators, which are designed to withstand such forces. Figure 5.11 shows how the support discs transfer the forces exerted on them along the longitudinal axes of the linear actuators.

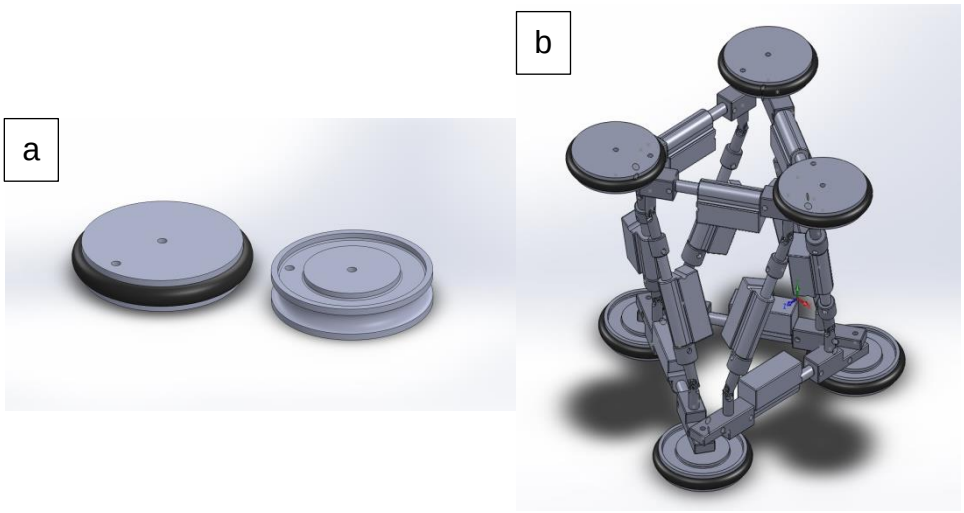


Figure 5.10 Robot support discs.

a) The support discs with and without the rubber O-ring; b) the support discs as mounted on the robot.

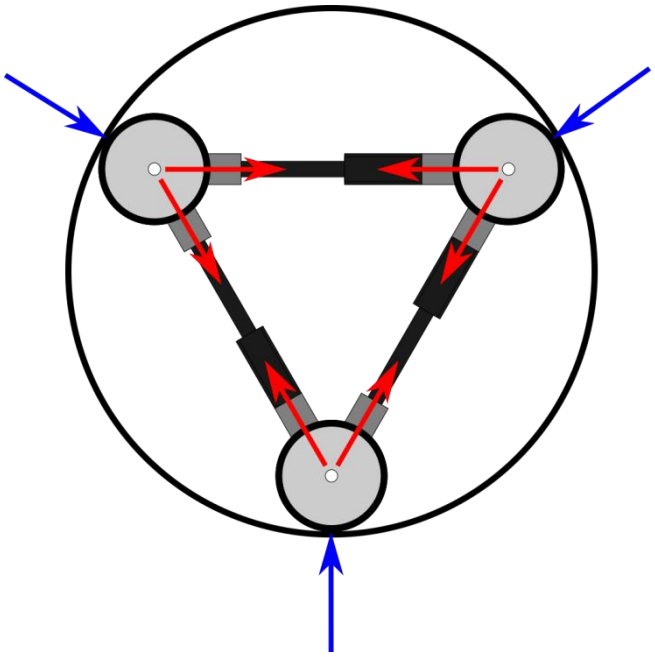


Figure 5.11 Forces acting on the support discs.

The reaction forces from the pipe wall (blue) are transmitted along the longitudinal axes of the linear actuators (red) by the support discs.

Because the discs are mounted slightly above the linear actuators' longitudinal axes, as shown in Figure 5.12, there would still be a bending moment inflicted on the actuators as the rings expand against the pipe wall. However, unlike with the support rods, it is possible to minimise this height without compromising the discs' function of preventing the Gough-Stewart actuators from colliding with the pipe wall, thus minimising the bending moments exerted on the actuators.

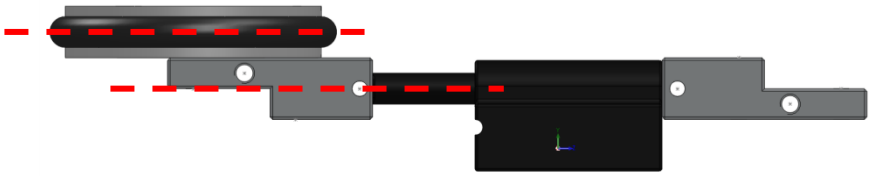


Figure 5.12 Height difference between the support disc and the linear actuator longitudinal axis.

5.3.3.1 Support Disc Dimensions

The dimensions of the discs were determined by specifying the actuator extension at which the robot should touch the pipe wall and then comparing the dimensions of the ring with the dimensions of the pipe and calculating the diameter of support disc required.

It was decided to design the discs so that the robot rings would touch the pipe wall when the linear actuators were at 80% extension. Since the actuators have a stroke length of 20mm, this would allow 4mm of the actuators' travel to be used for compressing the rubber on the discs to generate the required holding force, and would leave 16mm of inwards travel to be used for manipulating the position of the robot. At 80%/16mm extension, the length of each side of the triangle would be 122.5mm, as shown in the 3D model in Figure 5.13.

Figure 5.14 is a geometric drawing showing the dimensions of the robot ring at 80% extension inside the pipe wall. The radius, r, of the support discs can be calculated as follows:

$$r = p - m \tag{17}$$

Where p is the radius of the pipe and m is the distance between the triangle corner and the centroid, which is 2/3 of the triangle's median. The distance m can be calculated from the triangle side, s (determined to be 122.5mm earlier), as follows:

$$m = \frac{2}{3} \times s \frac{\sqrt{3}}{2} = \frac{s\sqrt{3}}{3} \tag{18}$$

$$m = \frac{122.5\sqrt{3}}{3} = 70.7mm \tag{19}$$

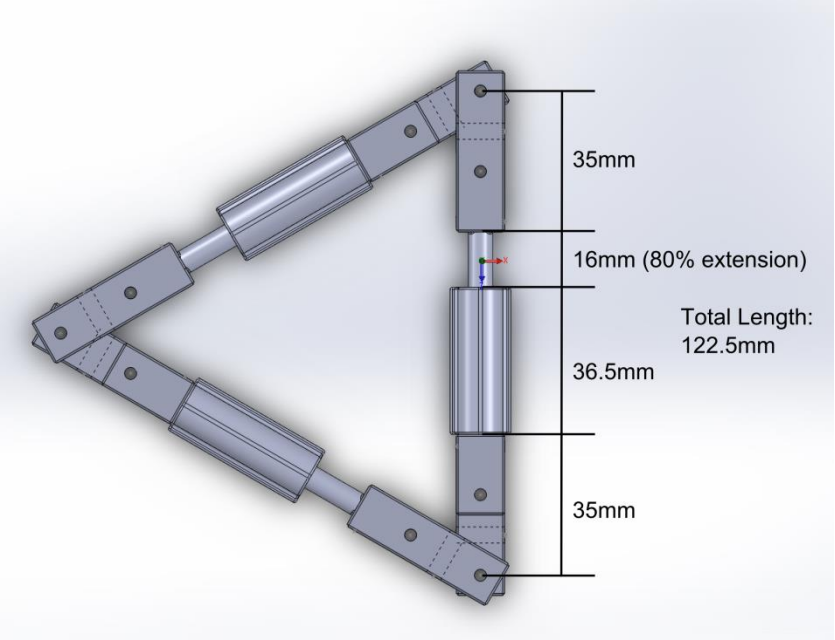


Figure 5.13 Length of the ring sides at 80% extension.

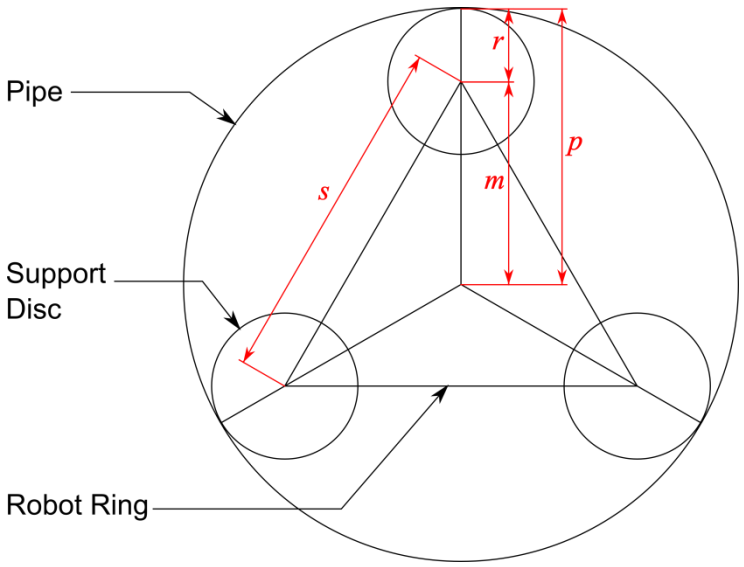


Figure 5.14 Geometric drawing to calculate the dimensions of the support discs.

The radius of the support disc can now be determined by subtracting the distance m from the radius of the pipe, p. The diameter of the pipe section available is 194mm, giving a radius of 97mm. The support disc radius is therefore:

$$r = p - m \tag{20}$$

$$r = 97 - 70.7 = 26.3mm \tag{21}$$

This gives a disc diameter of 52.6mm.

For the actual design, a 50mm diameter disc was turned from nylon, with a suitable groove for an O-ring. The O-ring selected was a 6mm thick ring with a 50mm outer diameter. When placed on to the disc, the total diameter of the disc is approximately 57.6mm (radius 28.8mm).

Based on the actual diameter of the support disc, the extension at which the robot's rings would touch the pipe wall could be calculated by determining the value of the triangle side, s, using the new disc diameter. To do so, equations (18) and (20) need to be rearranged so that s becomes the subject, as follows:

$$r = p - m \therefore m = p - r \tag{22}$$

$$m = \frac{s\sqrt{3}}{3} \tag{23}$$

$$\therefore s = \frac{3m}{\sqrt{3}} = m\sqrt{3} = (p - r)\sqrt{3} \tag{24}$$

$$\therefore s = (97 - 28.8)\sqrt{3} = 118.1mm \tag{25}$$

From Figure 5.13, the fixed lengths of the triangle side are:

$$35 + 35 + 36.5 = 106.5mm \tag{26}$$

Therefore, the actuator extension when the total length is 118.1mm is 11.6mm or 58%

The support discs are mounted at each corner of the rings with the same M3 screw forming the corner revolute joint passing through them to act as an axle.

5.3.4 3D CAD Assembly

The final design was assembled in Solidworks in order to confirm that all the parts will fit together as expected. Figure 5.15 shows various views of the 3D model.

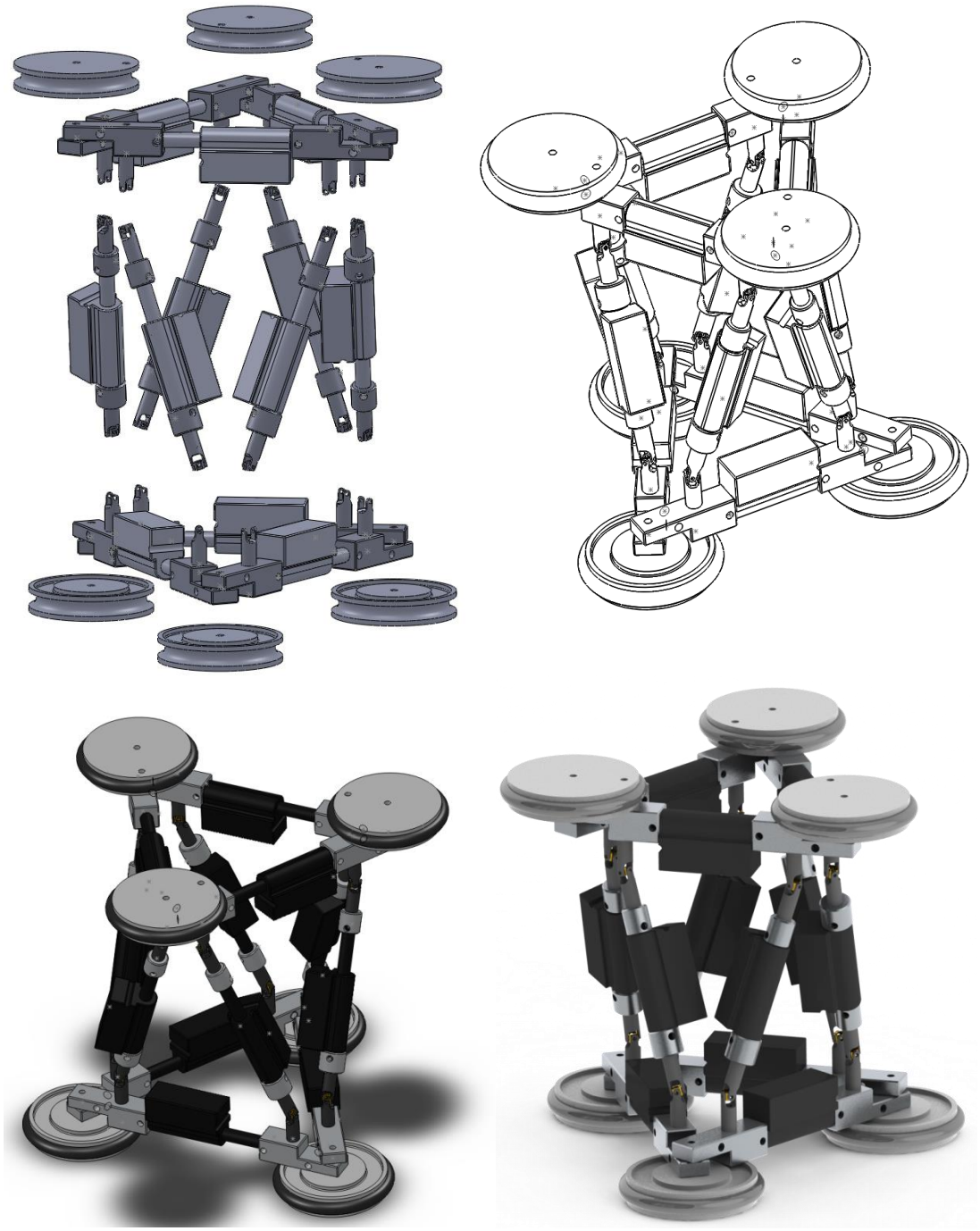


Figure 5.15 3D CAD model of the robot.

5.4 Electronics

The robot's electronic control circuitry needed to be designed to allow the mechanical structure to be computer controlled. When planning the electronic design, the main challenge identified was how to allow all 12 linear actuators to be controlled via a central controller using a manageable wiring scheme. As each linear actuator has 5 wires (2 for power, 3 for the feedback potentiometer) this results in a total of 60 wires that would need to

be interfaced to a central controller. Such a large number of wires was deemed to be impractical. Therefore, rather than all 12 motors being directly connected to a central controller, each motor would be connected to its own local controller, and the local controllers networked together using a bus system. A single power and communications cable can then connect the central controller to the bus and allow for the control of all 12 actuators with minimal wiring, as shown in Figure 5.16.

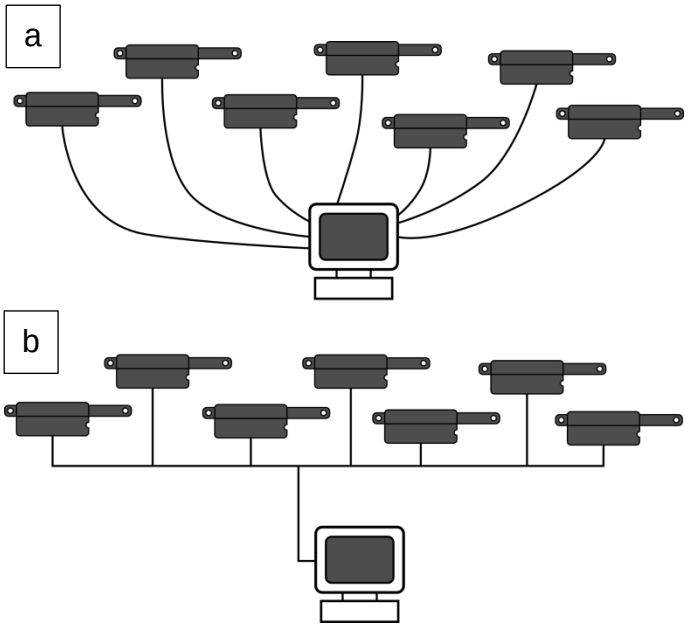


Figure 5.16 Direct wiring of the motor (a) vs bussed wiring (b).

The first step in designing the electronics was to determine the motor controller. After searching for a suitable off-the-shelf controller, it was decided to use the Pololu Jrk 21v3, shown in Figure 5.17. This is a small, USB motor controller with feedback that can be used for closed-loop speed or position control. The controller can accept commands via various different sources, including over a serial port using a custom protocol. The controller is capable of measuring the current consumed by the motor and making that information accessible for reading.

Figure 5.17 Pololu Jrk 21v3 USB Motor Controller with Feedback [120].

There were, however, some limitations in the controller that meant that on its own, it could not fulfil all the requirements. These are:

- The on-board serial port cannot be directly connected in a bus with other controllers.
- The resolution of the current monitoring circuitry is very low at 38mA per bit resulting in only 5 bits to represent the full 220mA range of the PQ12 linear actuators.
- The linear actuators use a proprietary FPC cable and therefore cannot be directly plugged into the terminal block on the Jrk.

In order to overcome these limitations, an add-on board was developed that plugged into the Jrk and provided the following functionality:

- Allows the Jrk controllers to be networked over an RS485 bus, increasing communication robustness.
- The units can be daisy-chained together, requiring only a single stretch of 4-core cable to run through the entire bus.
- Adds a high resolution current monitor for measuring the current consumed by the actuator in both the forward and reverse directions.
- Allows the linear actuators to be easily and directly connected to the controller.

A block diagram of the add-on board is shown below in Figure 5.18.

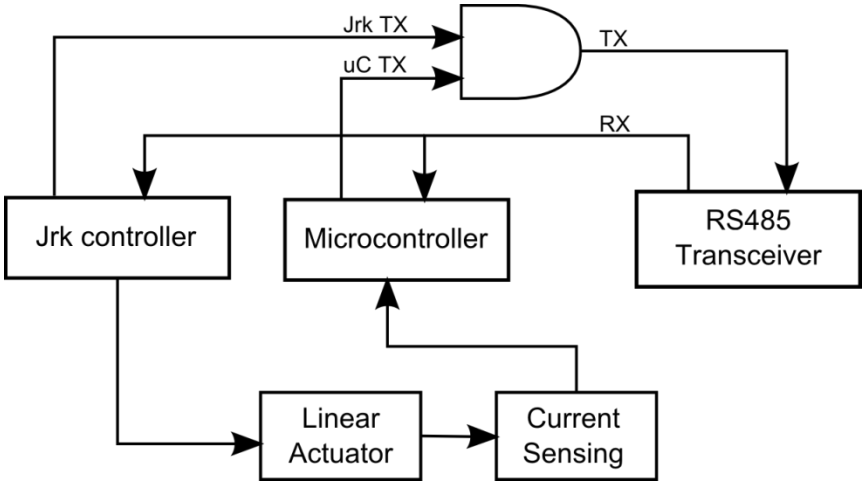


Figure 5.18 Controller add-on board block diagram.

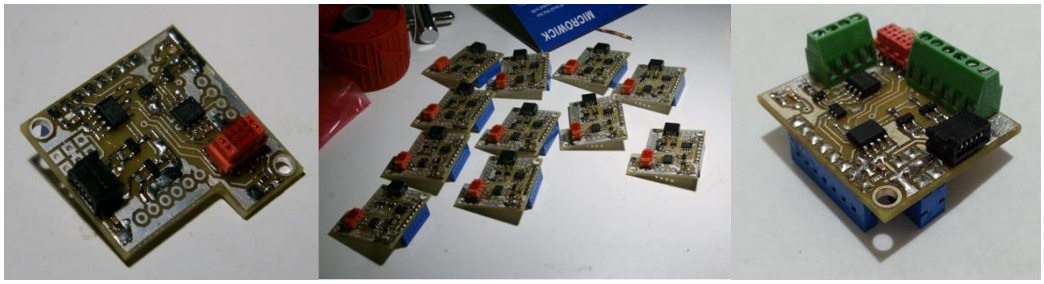


Figure 5.19 Add-on board in various stages of completion.

An SP485EEN RS485 transceiver chip and some logic gates convert the on-board TTL serial signals to RS485 and allow the Jrk controllers to be connected on a bus. A pair of ZXCT1109 high-side current monitors and a sense resistor measure and amplify the current consumed by the motor, allowing it to be read by an on-board PIC12F1822 microcontroller. The firmware on the microcontroller implements the same serial protocol that the Jrk controllers use, and is also connected to the RS485 bus. This allows both the Jrk controller and the microcontroller to be easily accessed via the same communication protocol. An on-board FPC connector allows the linear actuators to be directly connected to the add-on board. Figure 5.20 shows the completed electronic control module assembly, showing both the add-on board and Jrk controller, along with how the two boards are connected together.

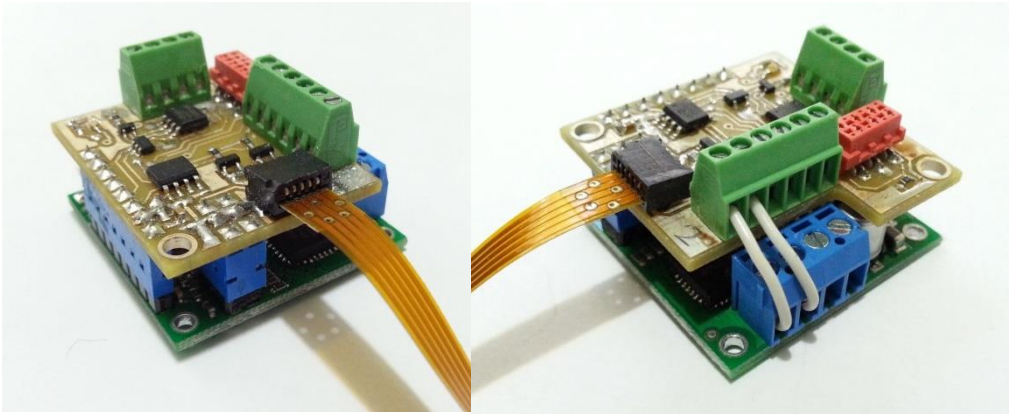


Figure 5.20 Completed electronic control module for the linear actuators.

The green terminal blocks on the add-on board are used to connect the power and RS485 signals. The same connections are replicated on both terminal blocks, allowing the modules to be daisy-chained together as shown below.

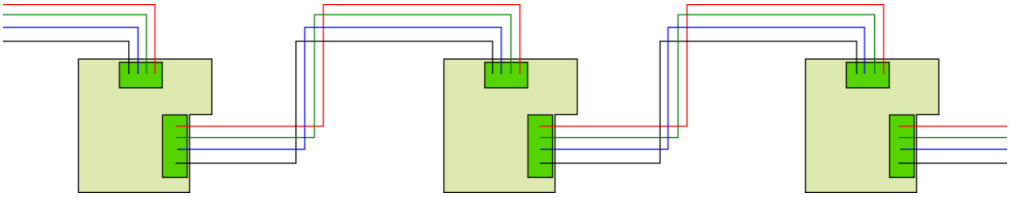


Figure 5.21 Daisy-chaining of the control boards.

The add-on boards are daisy chained via the two terminal blocks on each one.

After assembling the boards and programming the microcontroller with the firmware, the boards needed to be tested in order to verify their operation. The following tests were carried out:

- RS485 communication with the Jrk.
- RS485 communication with the microcontroller.
- Current monitor functionality.
- Communication test in a network.

Testing communication with the Jrk involved sending commands to it over the RS485 bus, with only one node attached, and verifying that the commands were executed successfully. This could be done by connecting one of the linear actuators to the node and then sending the commands to extend and retract the actuator and visually verifying that the command was being executed. Similarly, both the communication with the microcontroller and the current monitor functionality could be tested by stalling the linear actuator in both the forward and reverse directions and reading the current measured by the microcontroller in both cases and verifying that the result is a value whose magnitude is close to the maximum possible value (due to the motor drawing its maximum current when stalled) and that the sign of the value matches the direction that the actuator is driven towards.

The same set of basic tests described above were also carried out in a network with all 12 devices attached and each device configured to a unique address, to ensure that there were no communication issues.

5.4.1 Mounting

Due to the size of the motor controllers, care needed to be taken when mounting them on the robot to ensure that they did not interfere with the robot's movement. A representative CAD model of the controllers was drawn up in Solidworks and this was used to arrange them on the main robot assembly so as to minimise the chances of interference. It was decided to mount each controller on the outside face of its corresponding actuator in

order to maximise the distance between them. Figure 5.22 shows how the controllers are arranged on the robot.

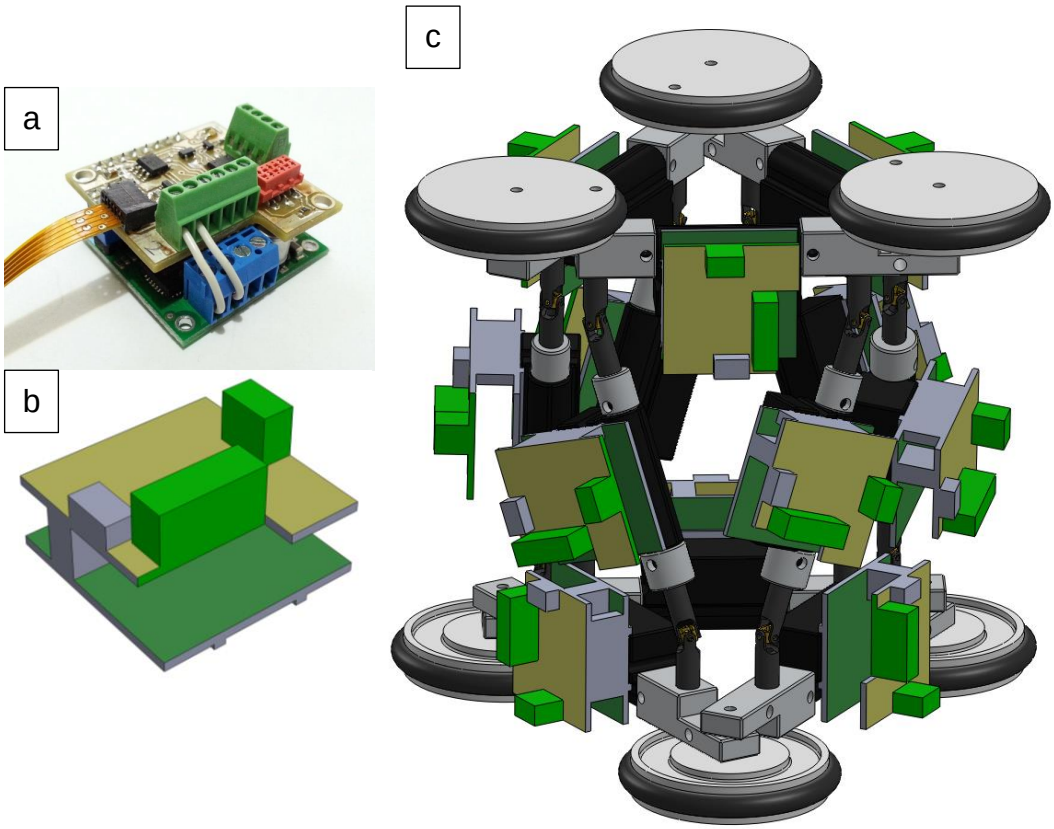


Figure 5.22 3D CAD model of the robot with motor controllers mounted.

(a) Photo of the motor controller; (b) 3D CAD model of the controller; (c) 3D CAD model of the robot assembly showing the mounting positions of the controllers.

5.5 PC Software

In order to control the electronics and robot easily, PC-based software was written that would allow the controllers to be configured and controlled via an easy-to-use GUI. The software was written in Java and is composed of several tabs, where each tab represents a different function, with some of the functions outlined below:

- Basic command test tab
- Motor control tab
- Automated movement and data logging tab
- Detailed motor monitoring tab

Figure 5.23 shows some screenshots of the software.

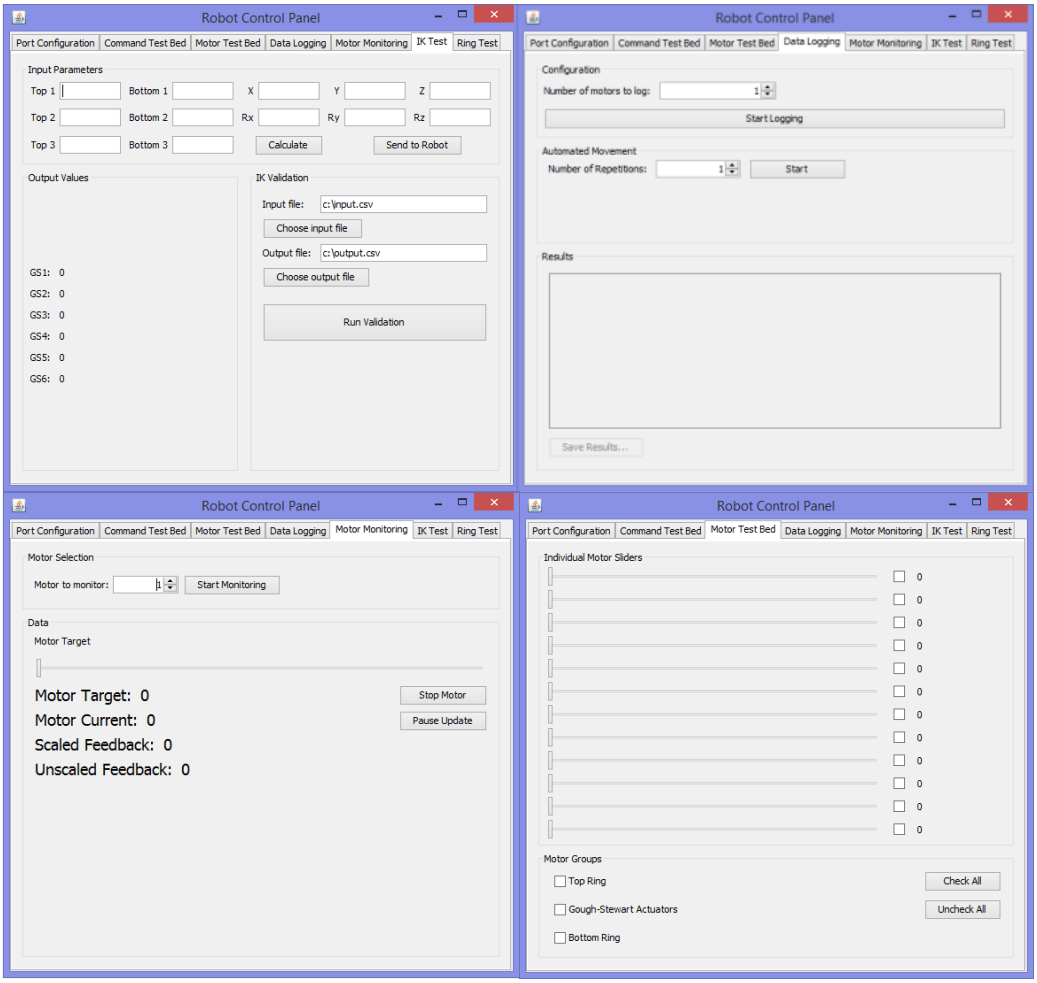


Figure 5.23 Screenshots of the PC control software.

5.6 Final Robot Assembly

With all of the components designed and specified, they could be purchased and manufactured and the robot prototype assembled. Figure 5.24, Figure 5.25, Figure 5.26 and Figure 5.27 show the robot in various stages of assembly, as well as the final prototype.

Figure 5.24 Assembly of the robot's mechanical structure.

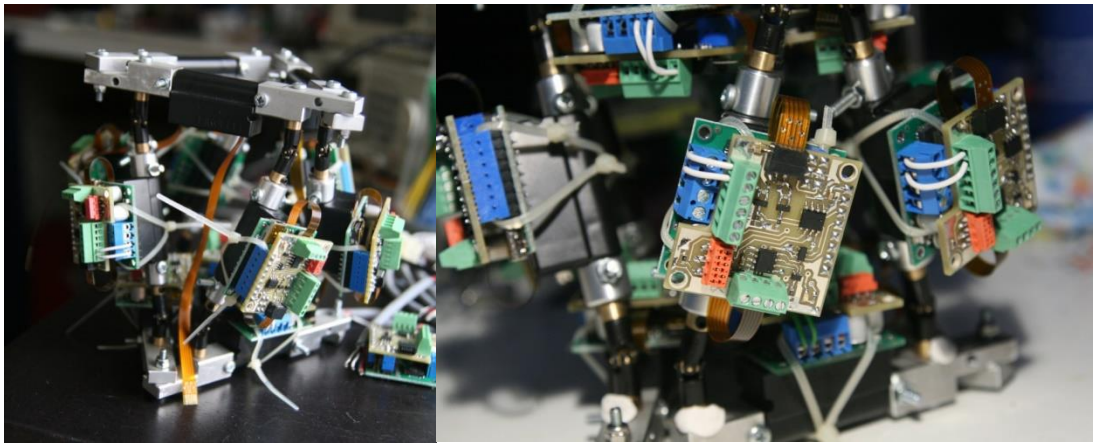


Figure 5.25 Mounting of the robot's control modules.

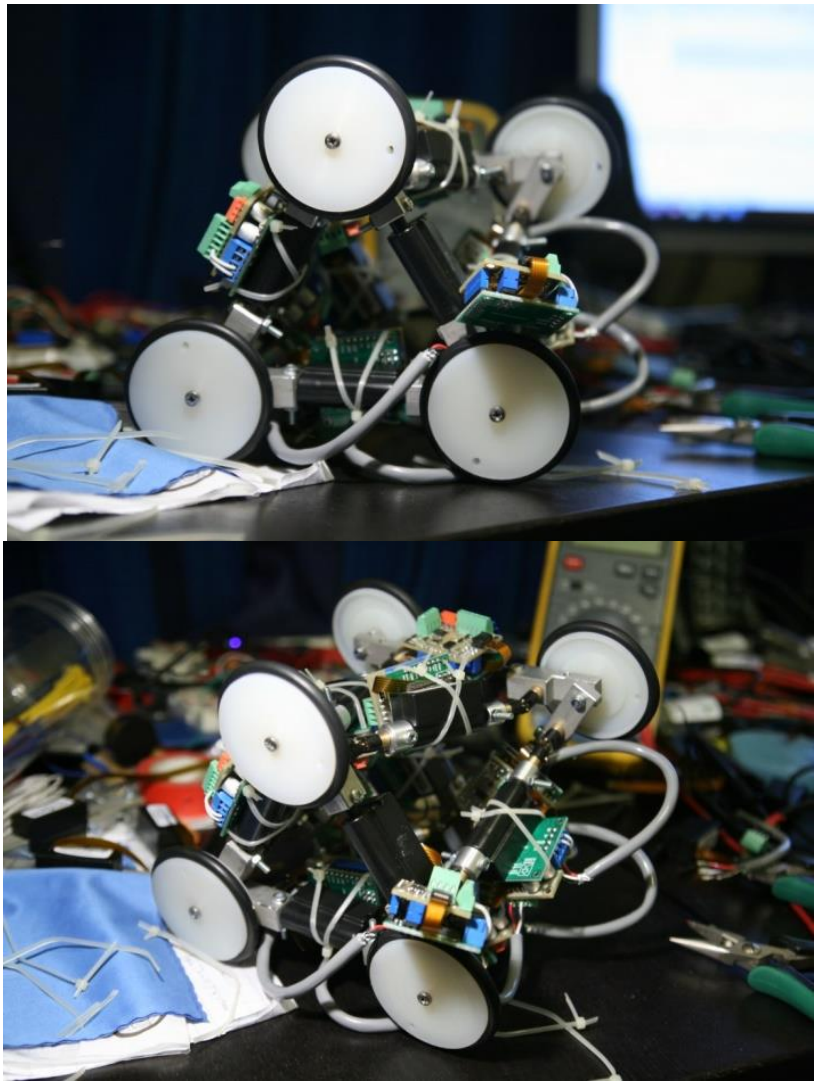


Figure 5.26 Support discs mounted on the robot.

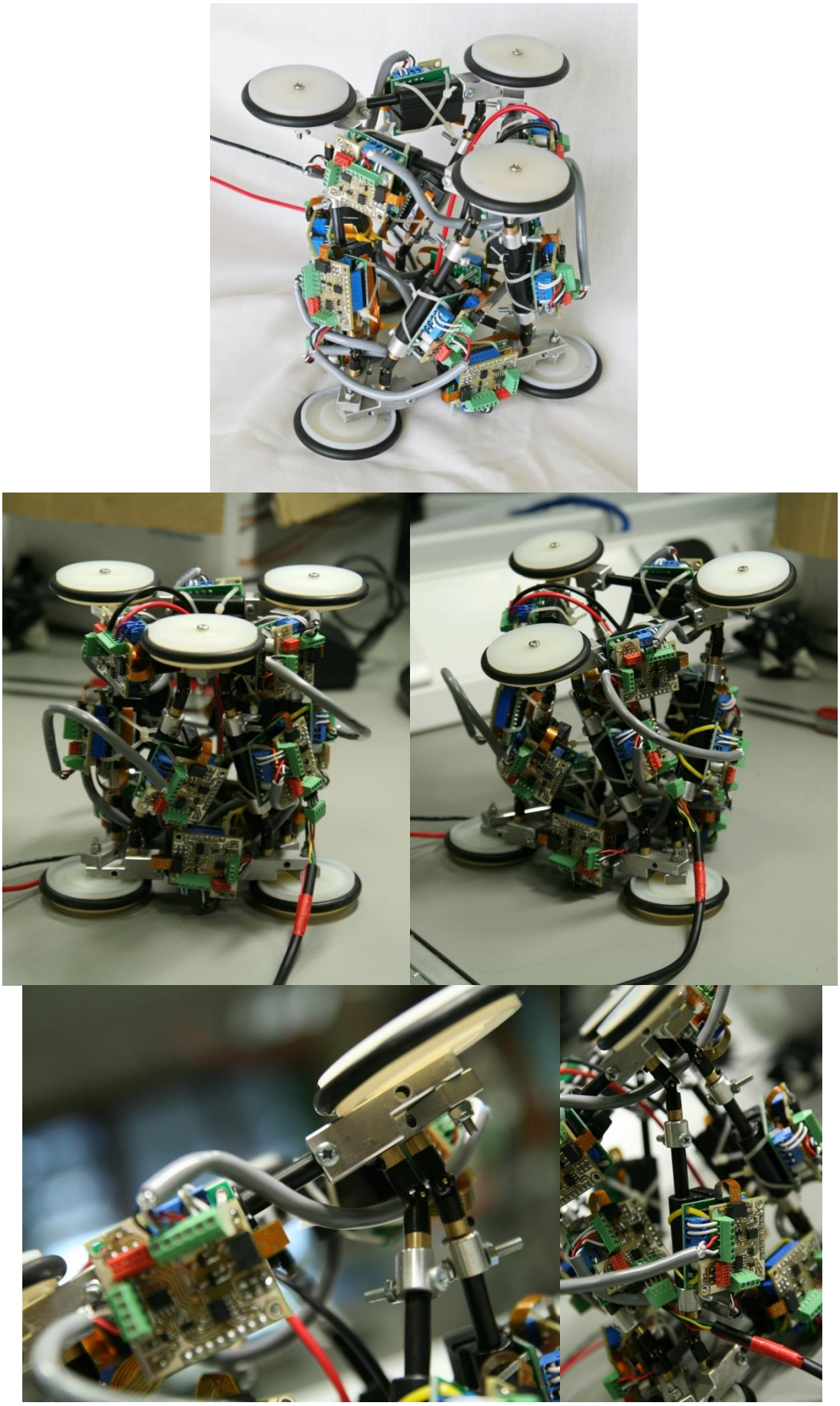


Figure 5.27 The completed robot prototype.

5.7 Actuator Force Update

With the prototype constructed, it's actual weight could now be measured. This turned out to be 920g. The actuator forces that were estimated previously in section 5.3.1.2 could now be updated. The values are summarised in Table 5.2.

Table 5.2 Actuators forces based on the robot's actual weight.

Actuator	Maximum Force
Ring Actuators	6.948N
Gough-Stewart Actuators	0.8233N

With the actuators' maximum force being 20N, they would still be capable of overcoming the forces needed to actuate the robot.

5.8 Summary

In this chapter the implementation of the working prototype from the conceptual design was presented. The implementation consisted of two main sections: the mechanical design and the control electronics.

The first step in the mechanical design involved selecting the linear actuators that would be used in the robot. This was governed by two main factors: the required actuator force and the required working length. The required force was determined by calculating the forces that the actuators would be exposed to based on their location and function in the robot. The required working length was determined by looking at the geometry of the robot and calculating the lengths required for the robot to fit inside the available pipe samples. Once these two factors were determined, a suitable actuator was selected.

The second step in the mechanical design involved the design and selection of the various connecting parts and joints that would be used to connect the linear actuators together and form the robot's structure. Universal joints were chosen to connect the Gough-Stewart actuators with the top and bottom rings, with the robot's DOFs recalculated to ensure that the required number of DOFs would still be met. The interconnecting mechanical parts were then designed by looking at the design of the universal joints and actuators and determining how these could be connected together via parts that would be simple to manufacture.

The final step in the mechanical design involved the design of the robot's support discs. The need for such a feature was outlined and the pros and cons of two different implementation methods were discussed, culminating in the decision to use discs rather than protruding rods as the use of rods could easily damage the linear actuators. The size of the support discs was then determined by examining the geometry of the robot inside the pipe and determining the disc size required for them to touch the pipe wall when the actuators were at a particular extension.

The design focus was then moved to the control electronics. The first step was to determine how the linear actuators, motor controllers and a central control computer (a PC) would be connected together, and it was decided that a bussed wiring system would be the most practical method.

An off the shelf motor controller was chosen to drive the linear actuators, but several shortcomings were identified that could not allow it to be used directly. These were: the inability to communicate in a bussed fashion; the inability to measure the actuator current with sufficient resolution; and the inability to easily connect the linear actuators to the controller. In order to overcome these limitations an add-on board was designed that could plug into the motor controller and provide these features.

The final part to the design of the working prototype was the control software running on a PC. This is responsible for providing a user interface that would allow the robot to be easily monitored and controlled.

Chapter 6
Theoretical Analysis and Simulation

6.1 Introduction

The aim of this chapter is to present the theoretical work that was performed around the robot design in order to predict and understand its performance, properties and limitations, as well as the simulation work that was carried out in order to verify the theoretical models and prove their theoretical validity before they were tested experimentally.

This chapter explores three important properties of the robot: the mathematical model that governs its geometry (i.e. inverse kinematics), the limitations on its movement and the effects of fluid flow on the robot. Each one of these is an important criterion in the development of a robot capable of operation in active pipelines. The inverse kinematics are important as they form the basis of the robot's control system which will propel it through the pipe. Understanding the robot's movement limitations and how these link with the robot's geometry allows the capabilities of the robot in navigating pipe networks to be assessed and aids in the understanding of how the mechanical design can be adjusted to accommodate different pipe networks. Finally, understanding the effects of the fluid on the robot is necessary in order to assess the feasibility of it working in an active pipeline and can help guide future design processes on where to make modifications to minimise drag.

The chapter is split into three parts, with each part presenting one of the properties mentioned above, its theoretical analysis and simulation validation.

6.2 Inverse Kinematics

Understanding the link between the physical space occupied by the robot and the actuators that form it is crucial when designing any robotic system. This is because the control system needs to have an understanding of how the actuators that it directly controls affect the physical shape of the robot and the space it occupies in its working environment, so that it can correctly move the robot within that environment and manipulate its surroundings. In robotics, the study of the relationship between a robot's actuators and the position and motion of its end effector is called Kinematics, which can further be split into two subsections: Forward Kinematics (using the actuators as

inputs to determine the end effector location and motion) and Inverse Kinematics (using a desired end effector position and motion to determine the required actuator inputs).

For this robot, the actuators can be split into three main categories: the 3 top ring actuators, the 3 bottom ring actuators and the 6 Gough-Stewart (GS) actuators. These are shown in Figure 6.1.

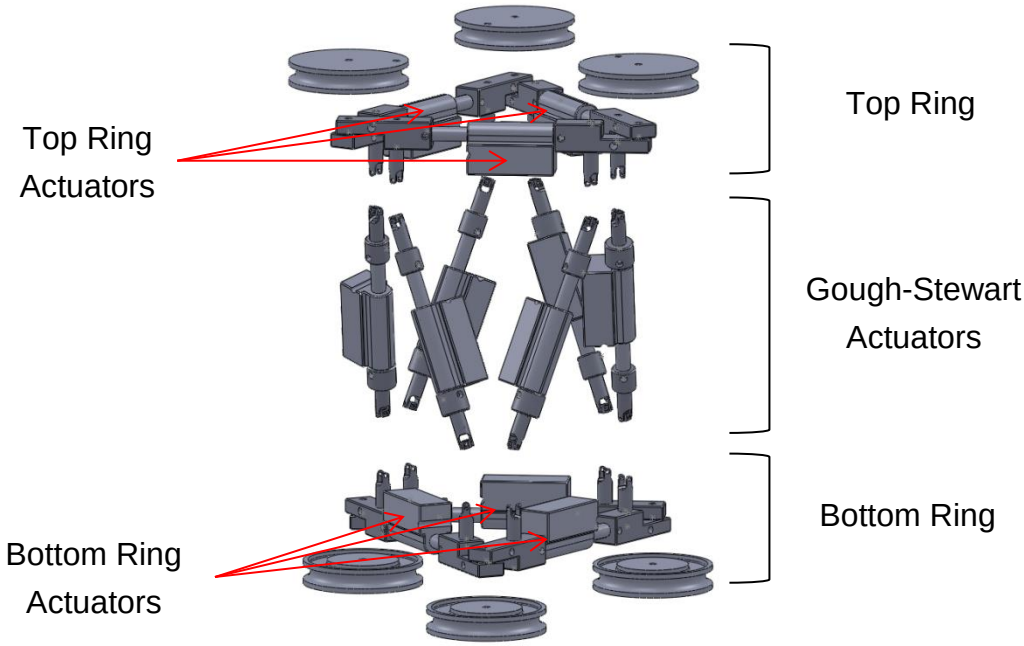


Figure 6.1 Robot nomenclature.

The top and bottom rings are primarily used to grip the pipe wall and hold the robot in place. Due to the small number of actuators and their simple geometry, it was expected that the mode of operation for these actuators would either be all fully retracted (in order to release from the pipe wall) or all extended to the same length (in order to grip the pipe wall). This simple mode of operation allows them to be directly controlled by the control system. The position and orientation of the two rings relative to each other is governed by the extensions of the 6 Gough-Stewart actuators. These are the primary actuators used to propel the robot through the pipe network, and the relationship between their extensions and the robot's target position and orientation needed to be found. The inverse kinematics equations therefore needed to be derived. These would rely on the target position and orientation being fed to the equations and, along with the existing extensions of the ring actuators, would determine what Gough-Stewart extensions are needed in order to achieve those targets.

The derivation was performed by analysing the robot's geometry and solving the vector equations that relate the position and orientation of the two rings

via the location and orientation of the linear actuators. Once the equations were derived, they were integrated into the robot's PC control software to allow them to be used to directly control the robot prototype. Following that, in order to verify their validity, a simulation model of the robot was built in software and then the results of the two were compared. This process is outlined below.

6.2.1 Nomenclature

The first step involved defining some of the basic geometric entities on the robot, specifically, the robot's origin, it's axis system and the geometric points and lines that would be used in the derivation. Figure 6.2 shows a vector representation of the robot structure. The following definitions were made:

- Points $T_{1..6}$ and $B_{1..6}$ represent the connections between the top and bottom rings and their respective universal joints.
- The counterpart points $T'_{1..6}$ and $B'_{1..6}$ are the projections of $T_{1..6}$ and $B_{1..6}$, respectively, on to a plane that is parallel to the original ring plane but is coincident with the universal joint hinges.
- Points O_T and O_B are the local origins of the top and bottom rings, respectively. The origins are defined as the centroids of the triangle of linear actuators that form each ring.
- O_B is considered to be the global origin of the robot and its axes are used as the global axis system, whilst O_T is the target position.
- The axis definitions are as follows:
    - The x-axis points from the ring origin to the point P_1, where P is either B or T depending on whether the ring being discussed is the bottom or top ring, respectively.
    - The y-axis points to the "left", in the general direction of the points P_5 and P_6.
    - The z-axis points "up" towards the top of the robot.
- At the rest position with all actuators retracted, O_T is at angle of 60° about the global z-axis.

For simplicity, the support discs have been ignored as they can be represented by simple offsets along the z-axes of the local origins.

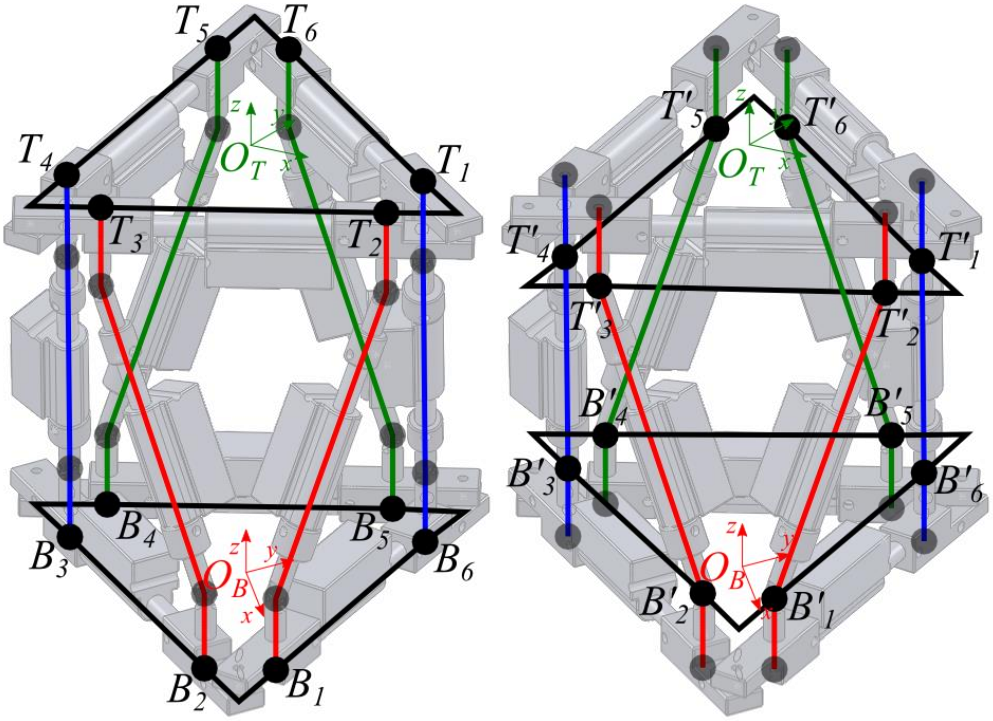


Figure 6.2 Simplified vector representation of the robot structure.

The main points are shown on the left, and their counterpart points on the right.

6.2.2 Derivation

The vector representing any of the six Gough-Stewart linear actuators can be expressed as $\overrightarrow{B'_u T'_v}$, where $u, v \rightarrow 1..6$ and the combination of u and v is one of the following pairs: (1,2), (2,3), (3,4), (4,5), (5,6) and (6,1). The modulus of this vector represents the sum of the linear actuator's total length (including extension) and the fixed length of the connecting joints. In order to determine its modulus, the positions B'_u and T'_v relative to the robot origin need to be determined. Figure 6.3 shows how these points relate to the geometry of the ring.

The points $P_{1..6}$ represent either $T_{1..6}$ or $B_{1..6}$ depending on whether the top or bottom ring is being examined. The lengths $d_{1..6}$ and the angles $\alpha_{1..6,1..6}$ can be calculated from the extensions of the linear actuators that form the ring and the dimensions of the links connecting them together. The equations that govern them can be derived by examining the geometry of the rings in detail, as shown in Figure 6.4. Table 6.1 lists the length and angle definitions used to derive the equations.

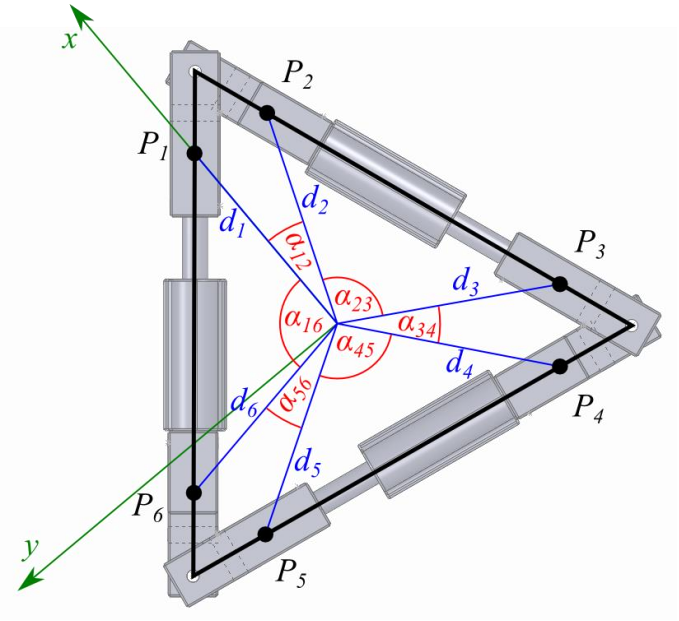


Figure 6.3 Geometry of the robot rings.

Table 6.1 Ring geometry definitions.

Variable	**Description**
Input variables	
L_{xy}	The lengths between the two universal joints on one side. This is a function of the current linear actuator extension, the size of the actuator body itself and the size of the joints connecting the linear actuators together.
t	The length between a universal joint and the closest corner revolute joint. This is a fixed value determined by the size of the joints connecting the linear actuators together.
Intermediate variables	
S_{xy}	The lengths of the triangle sides.
u_{xy}	The distances between two adjacent universal joints.
m_{xy}	The distances between the triangle corners and the triangle centroid ($\frac{2}{3}$ the length of each median).
θ_x	The angles between each triangle side and a median.
Output variables	
d_x	The lengths between each universal joint and the triangle centroid.
α_{xy}	The angles between the d_x segments.

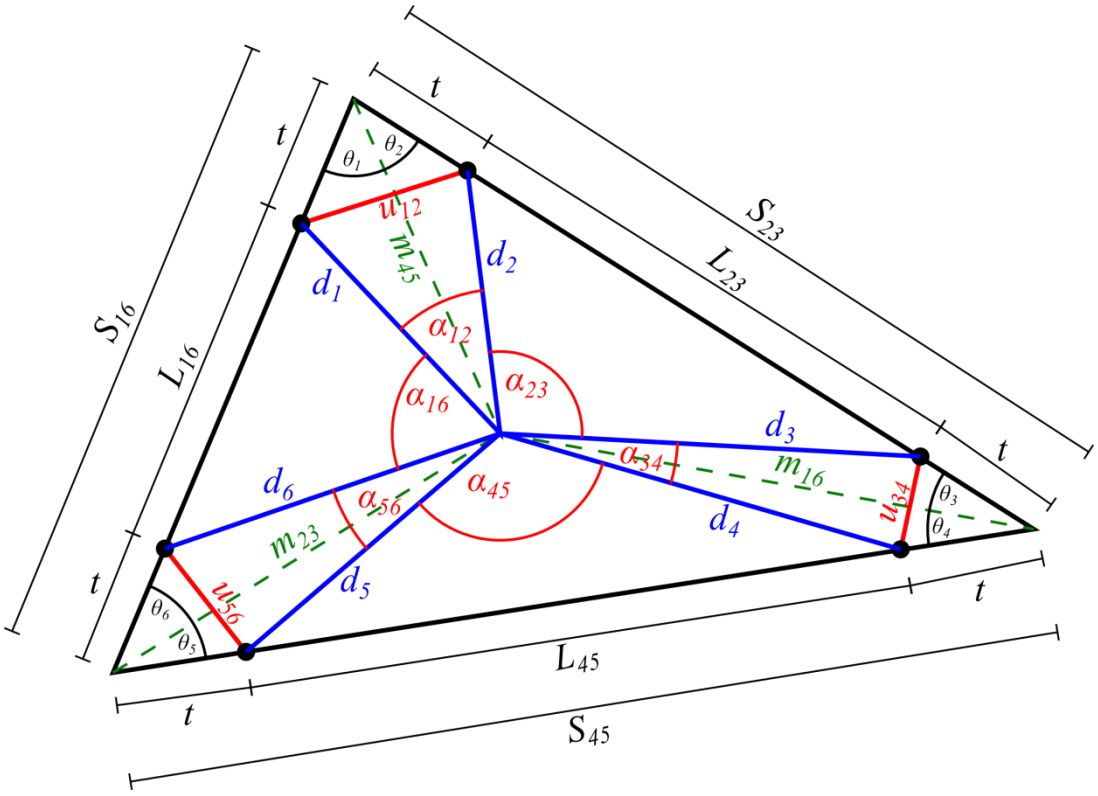


Figure 6.4 Detailed geometric representation of the ring.

The S_{xy} lengths are simply calculated by adding the corresponding L_{xy} and t lengths, as follows:

$$S_{16} = L_{16} + 2t \tag{27}$$

$$S_{23} = L_{23} + 2t \tag{28}$$

$$S_{45} = L_{45} + 2t \tag{29}$$

The m_{xy} lengths are then calculated using the standard formulas for triangle median length:

$$m_{16} = \frac{2}{3}\left(\frac{1}{2}\sqrt{2S_{23}^2 + 2S_{45}^2 - S_{16}^2}\right) = \frac{1}{3}\sqrt{2S_{23}^2 + 2S_{45}^2 - S_{16}^2} \tag{30}$$

$$m_{23} = \frac{1}{3}\sqrt{2S_{16}^2 + 2S_{45}^2 - S_{23}^2} \tag{31}$$

$$m_{45} = \frac{1}{3}\sqrt{2S_{16}^2 + 2S_{23}^2 - S_{45}^2} \tag{32}$$

With the S_{xy} and m_{xy} lengths determined, it is possible to calculate the θ_x angles by examining the sub-triangle formed by each triangle side and two

medians. In Figure 6.5, the triangle formed by S_{16}, m_{23} and m_{45} is shown along with the corresponding corner angle:

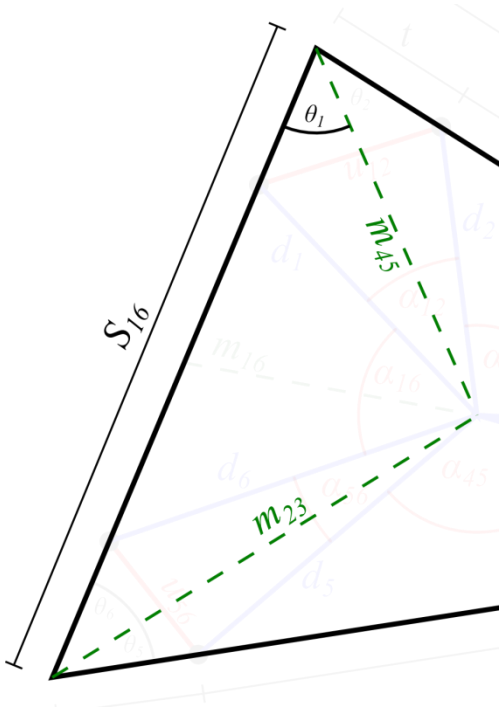


Figure 6.5 Geometry used to calculate θ_1.

Using the cosine rule:

$$m_{23}^2 = S_{16}^2 + m_{45}^2 - 2S_{16}m_{45}\cos\theta_1 \tag{33}$$

Rearranging this equation results in:

$$\theta_1 = \cos^{-1}\left(\frac{S_{16}^2 + m_{45}^2 - m_{23}^2}{2S_{16}m_{45}}\right) \tag{34}$$

Similarly, for the remaining angles:

$$\theta_2 = \cos^{-1}\left(\frac{S_{23}^2 + m_{45}^2 - m_{16}^2}{2S_{23}m_{45}}\right) \tag{35}$$

$$\theta_3 = \cos^{-1}\left(\frac{S_{23}^2 + m_{16}^2 - m_{45}^2}{2S_{23}m_{16}}\right) \tag{36}$$

$$\theta_4 = \cos^{-1}\left(\frac{S_{45}^2 + m_{16}^2 - m_{23}^2}{2S_{45}m_{16}}\right) \tag{37}$$

$$\theta_5 = \cos^{-1}\left(\frac{S_{45}^2 + m_{23}^2 - m_{16}^2}{2S_{45}m_{23}}\right) \tag{38}$$

$$\theta_6 = \cos^{-1}\left(\frac{S_{16}^2 + m_{23}^2 - m_{45}^2}{2S_{16}m_{23}}\right) \tag{39}$$

With the θ_x angles calculated, it becomes possible to determine the d_x lengths by examining the triangle formed by the t, d_x and m_{xy} lengths, an example of which is shown in Figure 6.6.

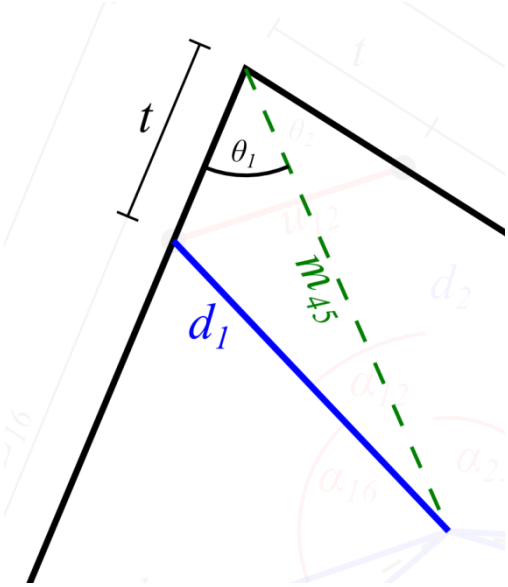


Figure 6.6 Geometry used to calculate d_1.

Using the cosine rule:

$$d_1^2 = t^2 + m_{45}^2 - 2tm_{45}\cos\theta_1 \quad (40)$$

$$d_1 = \sqrt{t^2 + m_{45}^2 - 2tm_{45}\cos\theta_1} \quad (41)$$

Similarly for the remaining d_x lengths:

$$d_2 = \sqrt{t^2 + m_{45}^2 - 2tm_{45}\cos\theta_2} \quad (42)$$

$$d_3 = \sqrt{t^2 + m_{16}^2 - 2tm_{16}\cos\theta_3} \quad (43)$$

$$d_4 = \sqrt{t^2 + m_{16}^2 - 2tm_{16}\cos\theta_4} \quad (44)$$

$$d_5 = \sqrt{t^2 + m_{23}^2 - 2tm_{23}\cos\theta_5} \quad (45)$$

$$d_6 = \sqrt{t^2 + m_{23}^2 - 2tm_{23}\cos\theta_6} \quad (46)$$

Before calculating the α_{xy} angles, the u_{xy} lengths need to be determined first. This is done by considering the triangle formed by two adjacent t lengths and their corresponding u_{xy} length, as shown in Figure 6.7:

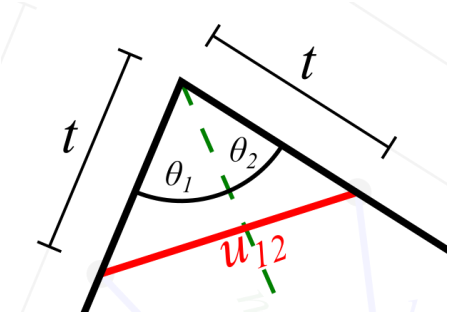


Figure 6.7 Geometry used to calculate u_{12}.

Using the cosine rule:

$$u_{12}^2 = 2t^2 - 2t^2\cos(\theta_1 + \theta_2) \quad (47)$$

$$u_{12} = \sqrt{2t^2 - 2t^2\cos(\theta_1 + \theta_2)} \quad (48)$$

Similarly for the other u_{xy} lengths:

$$u_{34} = \sqrt{2t^2 - 2t^2\cos(\theta_3 + \theta_4)} \quad (49)$$

$$u_{56} = \sqrt{2t^2 - 2t^2\cos(\theta_5 + \theta_6)} \quad (50)$$

The α_{xy} angles can now be calculated by considering the triangle formed by each u_{xy} length and its corresponding d_x lengths as shown in Figure 6.8:

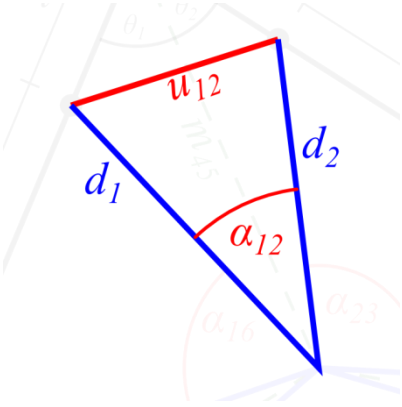


Figure 6.8 Geometry used to calculate α_{12}.

Using the cosine rule:

$$u_{12}^2 = d_1^2 + d_2^2 - 2d_1d_2\cos\alpha_{12} \quad (51)$$

Rearranging this equation results in:

$$\alpha_{12} = \cos^{-1}\left(\frac{d_1^2 + d_2^2 - u_{12}^2}{2d_1d_2}\right) \quad (52)$$

Similarly for the other α_{xy} angles:

$$\alpha_{23} = \cos^{-1}\left(\frac{d_2^2 + d_3^2 - L_{23}^2}{2d_2d_3}\right) \quad (53)$$

$$\alpha_{34} = \cos^{-1}\left(\frac{d_3^2 + d_4^2 - u_{34}^2}{2d_3d_4}\right) \quad (54)$$

$$\alpha_{45} = \cos^{-1}\left(\frac{d_4^2 + d_5^2 - L_{45}^2}{2d_4d_5}\right) \quad (55)$$

$$\alpha_{56} = \cos^{-1}\left(\frac{d_5^2 + d_6^2 - u_{56}^2}{2d_5d_6}\right) \quad (56)$$

$$\alpha_{16} = \cos^{-1}\left(\frac{d_1^2 + d_6^2 - L_{16}^2}{2d_1d_6}\right) \quad (57)$$

Note that for angles α_{23}, α_{45} and α_{16} their corresponding L_{xy} length was used instead of u_{xy}. With the d_x and the α_{xy} angles calculated, the vectors $\overrightarrow{P_{1..6}}$ relative to the ring's local origin can now be calculated using the equation below, assuming the axis system shown in Figure 6.3:

$$\overrightarrow{P_x} = \begin{bmatrix} d_x \cos\left(\sum_{i=1}^{x-1} \alpha_{i,i+1}\right) \\ -d_x \sin\left(\sum_{i=1}^{x-1} \alpha_{i,i+1}\right) \end{bmatrix}, \text{where } \overrightarrow{P_1} = \begin{bmatrix} d_1 \\ 0 \end{bmatrix} \quad (58)$$

The counterpart points for each ring are calculated by translating the original points along the ring's local z-axis for a distance s, where s is the distance between a universal joint's hinge and the point at which it connects to a ring. This is a fixed value for all the universal joints. For the top ring, this translation is in the –z direction, whereas for the bottom ring, it is in the +z direction, as shown below:

$$\overrightarrow{P'_{x\,Bottom}} = \overrightarrow{P_{x\,Bottom}} + \begin{bmatrix} 0 \\ 0 \\ s \end{bmatrix}, \text{and} \quad (59)$$

$$\overrightarrow{P'_{x\,Top}} = \overrightarrow{P_{x\,Top}} + \begin{bmatrix} 0 \\ 0 \\ -s \end{bmatrix} \quad (60)$$

The same translation is performed on O_T and O_B to calculate their counterparts O'_T and O'_B.

A further step is required for the vectors of the top ring and its origin, as they need to be expressed relative to the robot's global origin rather than that of the top ring. This is done by multiplying each vector by a transformation matrix, M, which represents the target orientation of the top ring relative to the bottom one, then applying a translation, which represents the location of the top ring relative to the bottom one:

$$\overrightarrow{T'_v} = \left(M \cdot \overrightarrow{T'_{v_{Local}}}\right) + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \quad (61)$$

$$\overrightarrow{O'_T} = \left(M \cdot \overrightarrow{O'_{T_{Local}}}\right) + \begin{bmatrix}\Delta x \\ \Delta y \\ \Delta z\end{bmatrix} \quad (62)$$

Once the positions of B'_u, T'_v, O'_B and O'_T have been determined, their values can be used to calculate the $\overrightarrow{B'_u T'_v}$ vector for each linear actuator. The modulus of $\overrightarrow{B'_u T'_v}$ can then be used to calculate the required extension by subtracting the fixed length of the linear actuator and the connecting parts. The extension can then be sent to the robot's control system.

Once the equations were derived, they were integrated into the PC control software, with a user interface that allowed the top and bottom ring actuator extensions and the target position and orientation to be specified. The software would then calculate the required Gough-Stewart actuator lengths and could send those directly to the robot. This allowed for some basic visual testing of the software implementation to ensure that it was generally orienting the robot in the right position and that other aspects, such as axis orientation, were correct. A screenshot of the IK user interface is shown below.

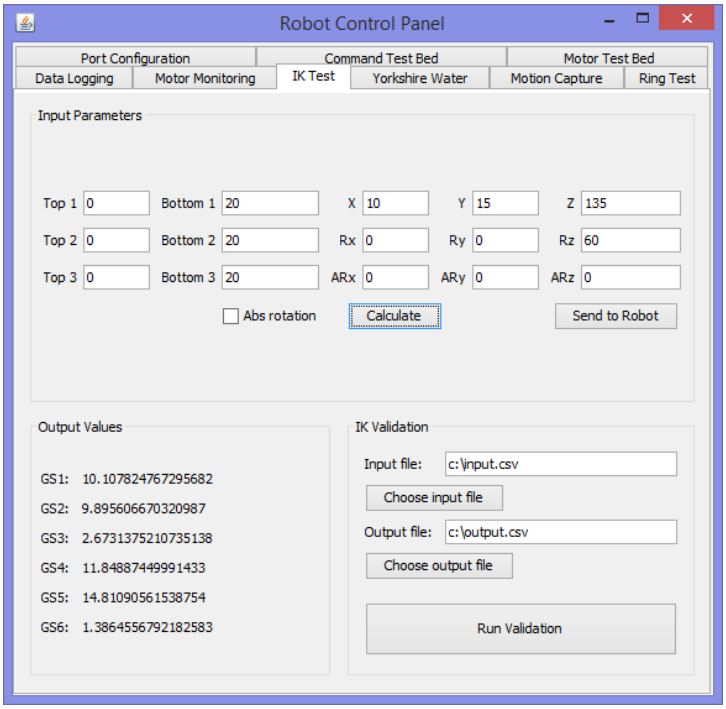


Figure 6.9 Screenshot of the PC software IK calculation feature.

6.3 Forward Kinematics

After the inverse kinematics were integrated into the control system, some basic visual testing was carried out using the robot prototype and the software. However, this only covered a few target positions and was not a practical way of testing the complete range of possible positions the robot

could take, hence the need for an automated, computerised verification method.

It was decided to do this by determining the forward kinematics model for the robot, which take the actuator extensions and work out the target position and orientation. Sample actuator extensions could be input into the forward kinematics model, resulting in an end effector position and orientation. These end effector values could then be taken and fed into the previously derived inverse kinematics equations as the target location and orientation. Those equations would then be used to calculate the actuator extensions that would achieve those targets. These calculated actuator extensions should match the original extensions input into the forward kinematics equations.

The problem of determining the forward kinematics for a Gough-Stewart platform is a topic that continues to attract research interest. Although the forward kinematics problem for Gough-Stewart platforms is solved, the solution method is non-trivial, with up to 40 direct kinematics solutions possible for each set of inputs [121]–[123]. For this application, since the actual equations themselves were not required (unlike the inverse kinematics), it was decided to use simulation software in order to model the robot's forward kinematics.

6.3.1 Adams Simulation

It was decided to use the Adams simulation package from MSC Software to build the model of the robot. This package can be used to build mechanical models, attach joints and specify kinematic constraints and then simulate the motion of the model under those constraints.

The use of Adams as a preliminary means of verifying the IK equations was deemed to be valid because it would use a different method to arrive at the same solution. The IK equations were derived algebraically by visually examining the robot's geometry and from that deducing the relationship between the various points on the robot. The Adams simulation works by defining the robot's links and joints in the software, and then using the software's numerical solver to numerically solve the forward kinematics problem and correctly position the robot. Therefore, by using two very different methods to arrive at the solution, the probability of the IK equations being valid is increased.

First, a kinematically accurate model of the robot needed to be constructed in Adams. In order to do that, a simplified 3D model of the robot was created

in Solidworks with the same degrees of freedom and dimensions as the original design. This removed the unnecessary geometric elements from the model (such as the detailed shape of the linear actuators) whilst keeping the kinematic structure intact, simplifying the simulation. The model was then imported into Adams and the joints and constraints defined to create a kinematically equivalent model. The only kinematic difference between the two models is that the actual robot uses two universal joints and a revolute joint for each Gough-Stewart actuator, whereas the simulation model uses a universal joint and a spherical joint. As discussed previously in 5.3.2.1 these two arrangements are equivalent. Figure 6.10 compares the original and simplified models. Figure 6.11 shows how the Adams model was constructed.

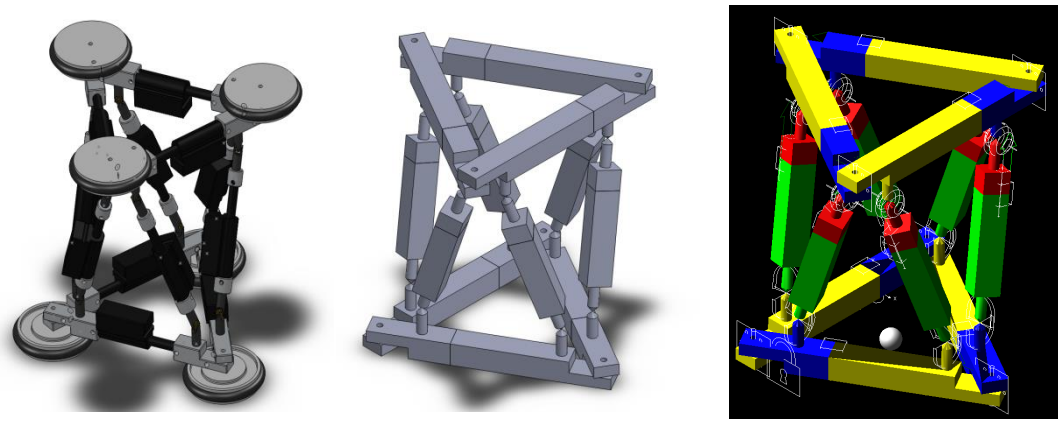


Figure 6.10 Left to right: the original robot model, simplified model, simplified model in Adams.

After the basic model was constructed, the robot's axis system needed to be defined in the simulation so that any measurements can be taken relative to the robot's own coordinate system and thus be directly comparable with the data generated by the IK equations. This involved two main steps: defining the position of the top and bottom origins and defining their orientation relative to the robot.

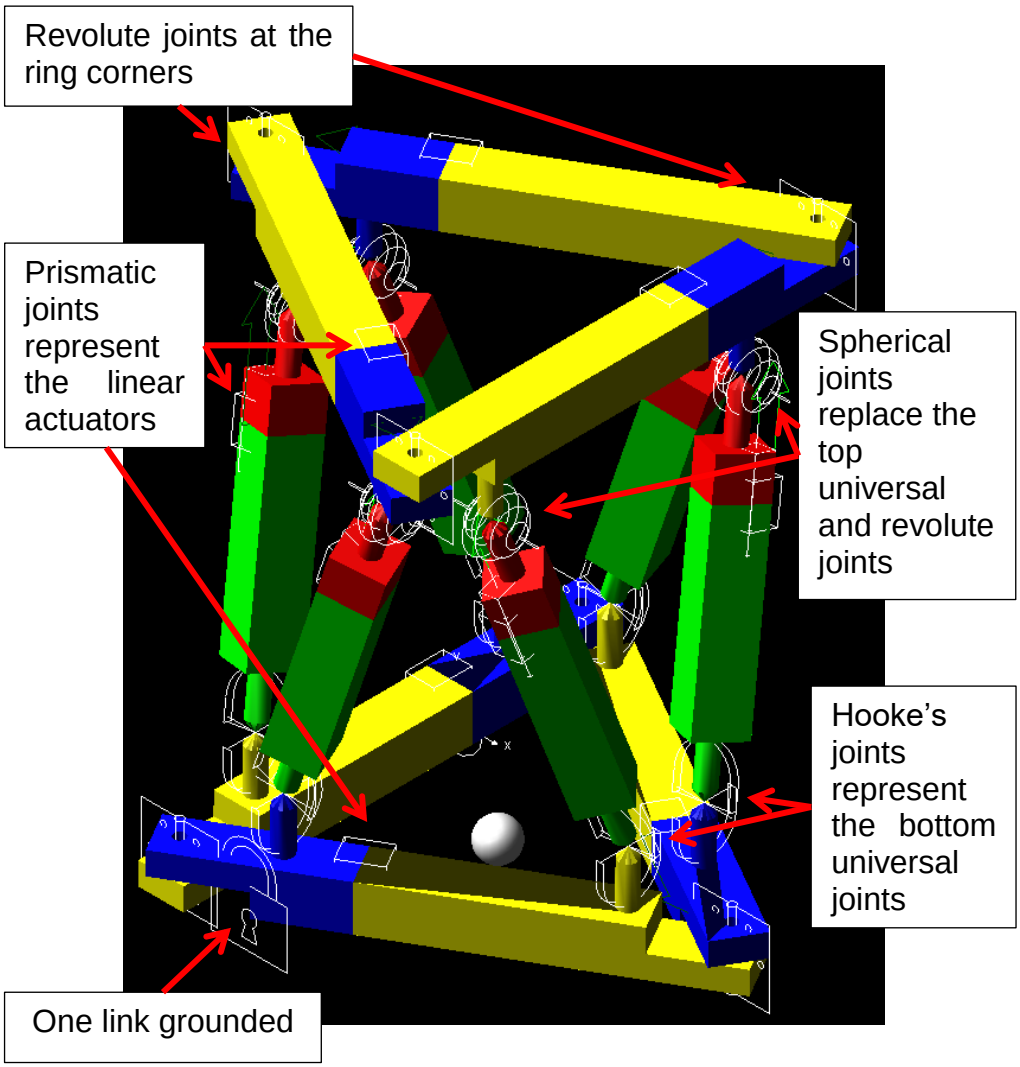


Figure 6.11 Annotated breakdown of the Adams model construction.

6.3.2 Origin Position Definition

As defined in section 6.2.1, the top and bottom ring origins are at the centroids of their respective rings, which can be calculated by averaging the coordinates of the three corners of each ring. Therefore a constraint needed to be created that would constrain the positions of the origin markers according to these rules. As this is a custom constraint and cannot be represented by a standard mechanical joint (such as revolute, prismatic, spherical etc.), a General Constraint (GCON) needed to be created. GCONs allow constraints to be built using mathematical equations, thus giving great flexibility in how to define the behaviour of the constraint. GCONs are defined as an expression that the simulation equates to 0 by adjusting the variables that make up the expression. Therefore, if the following relationship needs to be defined,

$$x = y + z \tag{63}$$

then it is created in Adams as:

$$0 = y + z - x \tag{64}$$

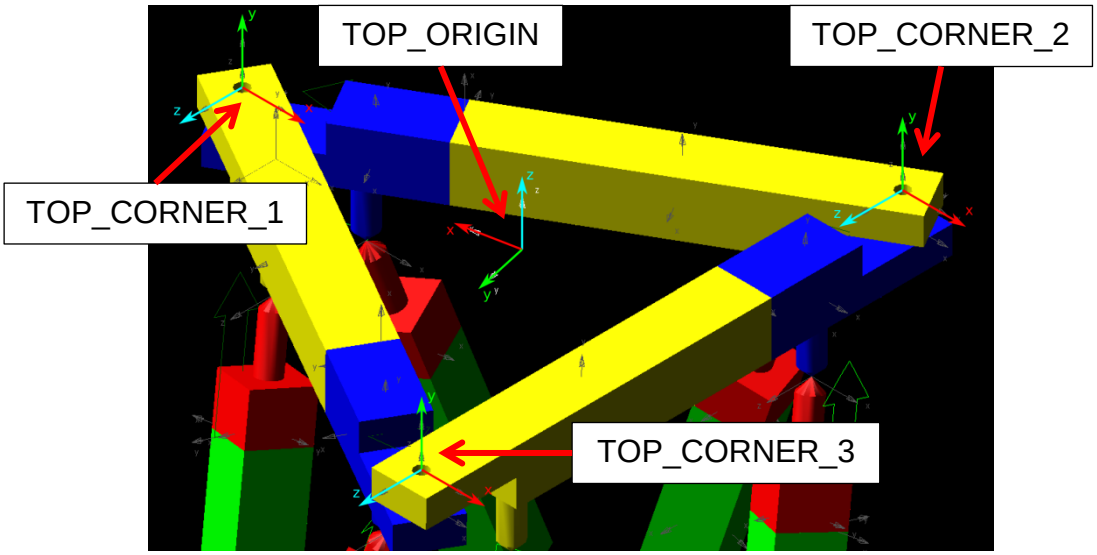


Figure 6.12 Locations of the markers used to constrain the origin position for the top ring.

Table 6.2 GCONs used to constrain the position of the top and bottom origin markers.

Top Origin X	DX(TOP_ORIGIN) - AO((DX(TOP_CORNER_3) + DX(TOP_CORNER_2) + DX(TOP_CORNER_1))/3)
Top Origin Y	DY(TOP_ORIGIN) - AO((DY(TOP_CORNER_3) + DY(TOP_CORNER_2) + DY(TOP_CORNER_1))/3)
Top Origin Z	DZ(TOP_ORIGIN) - AO((DZ(TOP_CORNER_3) + DZ(TOP_CORNER_2) + DZ(TOP_CORNER_1))/3)
Bottom Origin X	DX(BOTTOM_ORIGIN) - ((DX(BOTTOM_CORNER_1) + DX(BOTTOM_CORNER_2) + DX(BOTTOM_CORNER_3))/3)
Bottom Origin Z	DZ(BOTTOM_ORIGIN) - ((DZ(BOTTOM_CORNER_1) + DZ(BOTTOM_CORNER_2) + DZ(BOTTOM_CORNER_3))/3)
Bottom Origin Y	Fixed at 0

In order to produce this constraint, markers were attached to the model at the corners of each ring, and then a set of GCONs were created in Adams that used the positions of those markers to determine the position of the

origin marker. Figure 6.12 shows where the additional markers were created for the top ring (a similar setup was also created for the bottom ring). Table 6.2 shows the GCONs used to constrain the positions of the two origins. The y-value of the bottom origin marker (the simulation software uses the y-axis to represent “up”) was fixed at a constant value as the model was designed so that the bottom ring would only move in the horizontal plane and would not be subject to any vertical motion.

The constraints were tested by applying random extensions to all the actuators while outputting the positions of the origin markers and the corner markers. The positions of the top and bottom origin markers were then compared to the average positions of the corner markers to ensure that they coincided. Figure 6.13 shows the robot in various positions and how the top and bottom origin markers are maintained at the correct position.

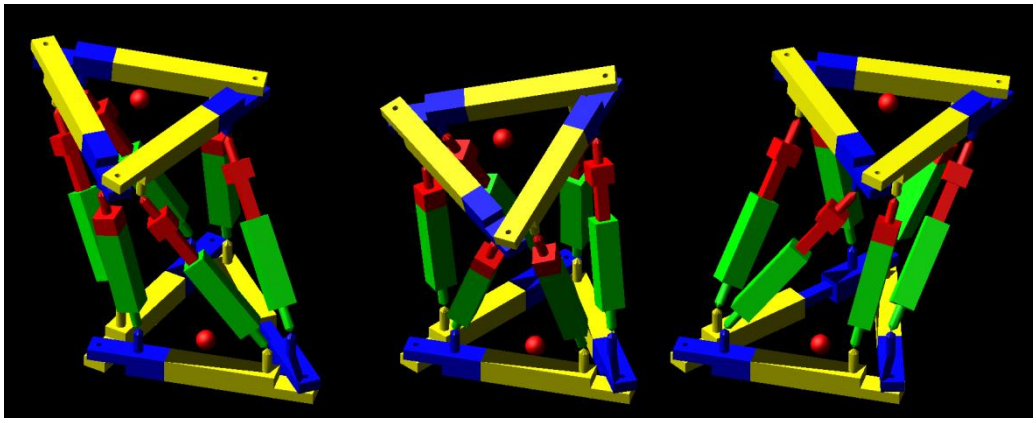


Figure 6.13 Screenshots showing how the top and bottom origin positions (shown as red spheres) are correctly maintained at various robot configurations.

Using these GCONs, the positions of the origin markers are correctly updated in real time by the simulation software, taking into account the extensions of the various actuators.

6.3.3 Origin Orientation Definition

The next step involved constraining the orientation of the markers so that they conform to the robot’s axis system, as defined in 6.2.1. For this case, the top and bottom origins were treated separately, as it was much simpler to create the GCONs to constrain the orientation of the bottom origin due to the fact that the bottom ring always remained horizontal.

6.3.3.1 Bottom Ring Orientation Definition

In order to create the GCONs that constrain the orientation of the bottom ring, the transformation from the simulation global axis system to the robot’s

axis system needed to be determined. Using the functions available as part of the simulation suite, this transformation could be expressed using three separate rotations in the following order: yaw (rotation in z), pitch (rotation in y) and roll (rotation in x). Figure 6.15 shows the effects of the three rotations on the marker's orientation.

The three steps are as follows:

- Rotate by +180° about the z-axis.
- Rotate by +α° about the y-axis to point the x-axis at the point B_1.
- Rotate by +90° about the x-axis to bring the y-axis on the horizontal plane and to point the z-axis upwards.

From the above steps, the angle α is unknown and needs to be determined in relation to the robot's geometry. In order to determine the angle α, the point B1, defined previously in section 6.2.1, was added to the robot model and the geometric representation shown in Figure 6.14 was used to determine the angle.

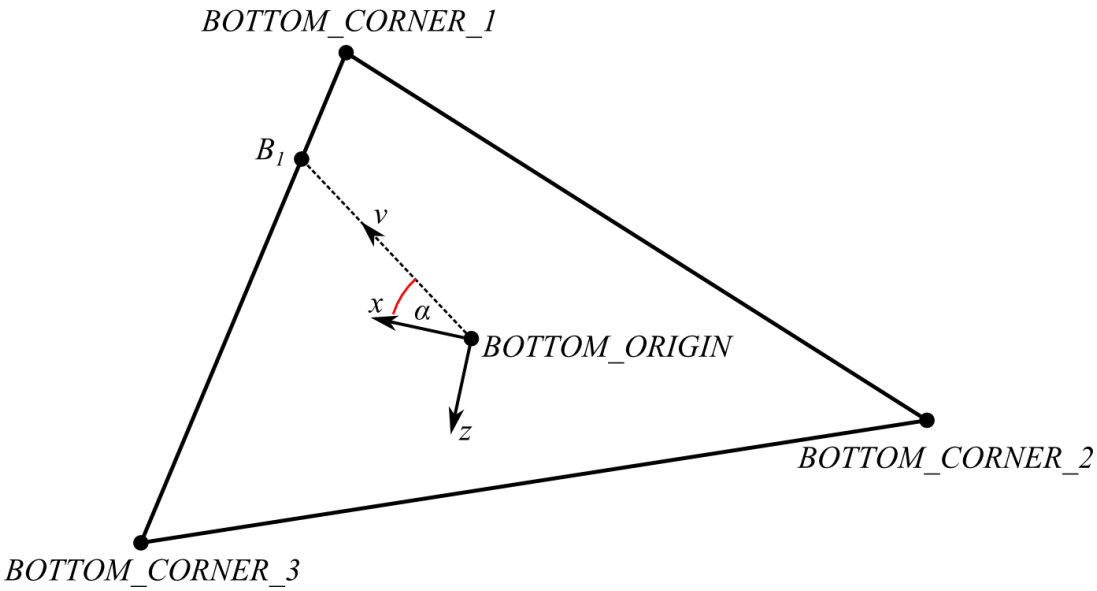


Figure 6.14 Geometric representation of the bottom ring to determine the angle α.

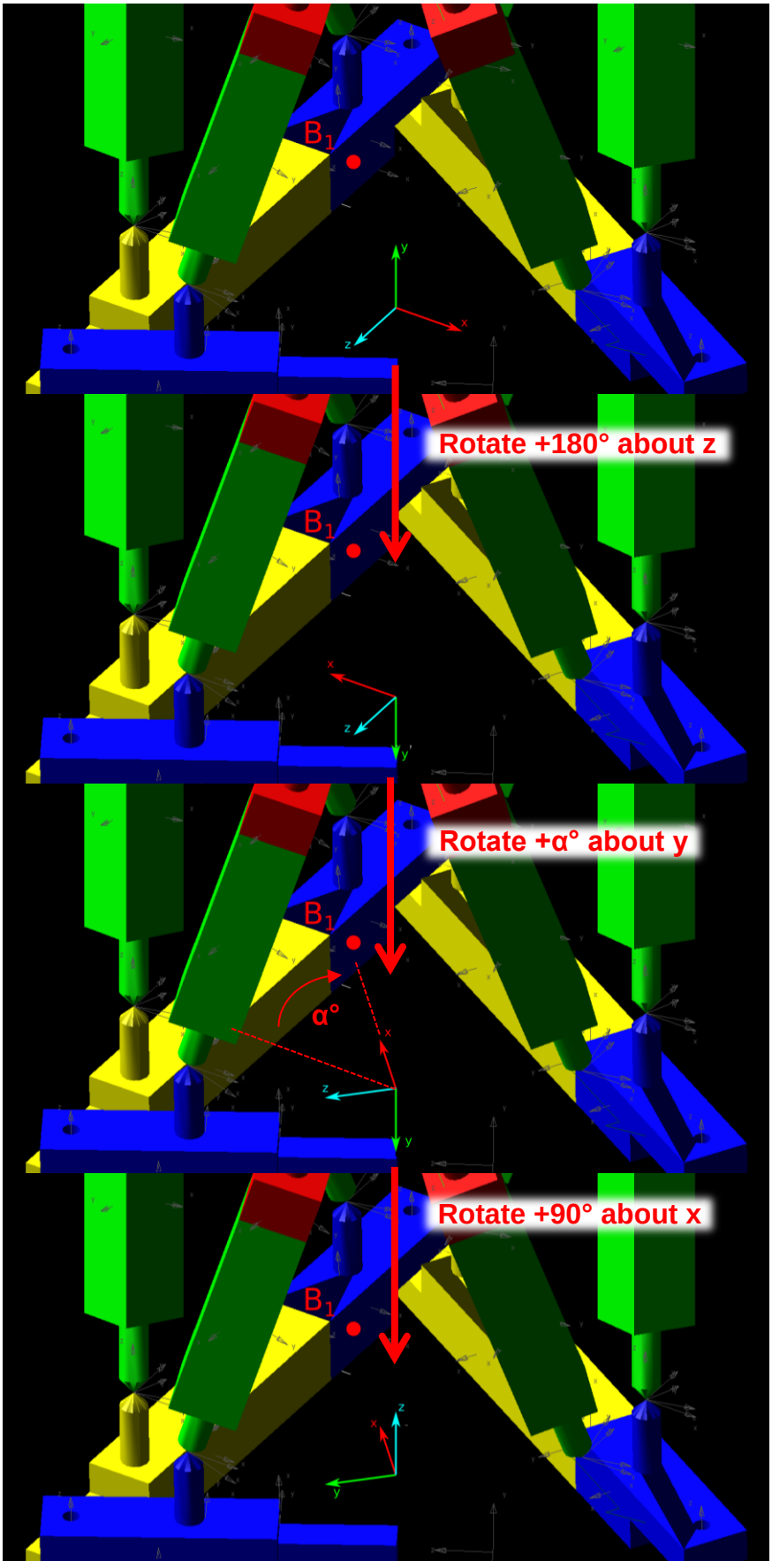


Figure 6.15 The three rotation steps applied on the bottom origin.

Note: Part of the robot's ring has been hidden for clarity.

The angle α, is the angle between the current x-axis of the origin marker and the vector between the bottom origin and the point B1, which is denoted by the vector v. This angle can be determined by using the dot product as shown in equations (65)-(70). The result of this equation was then implemented as a set of GCONs for the marker's yaw, pitch and roll and tested visually. Figure 6.16 shows how the bottom marker now maintained the correct orientation and position and thus could represent the robot's global origin and axis system.

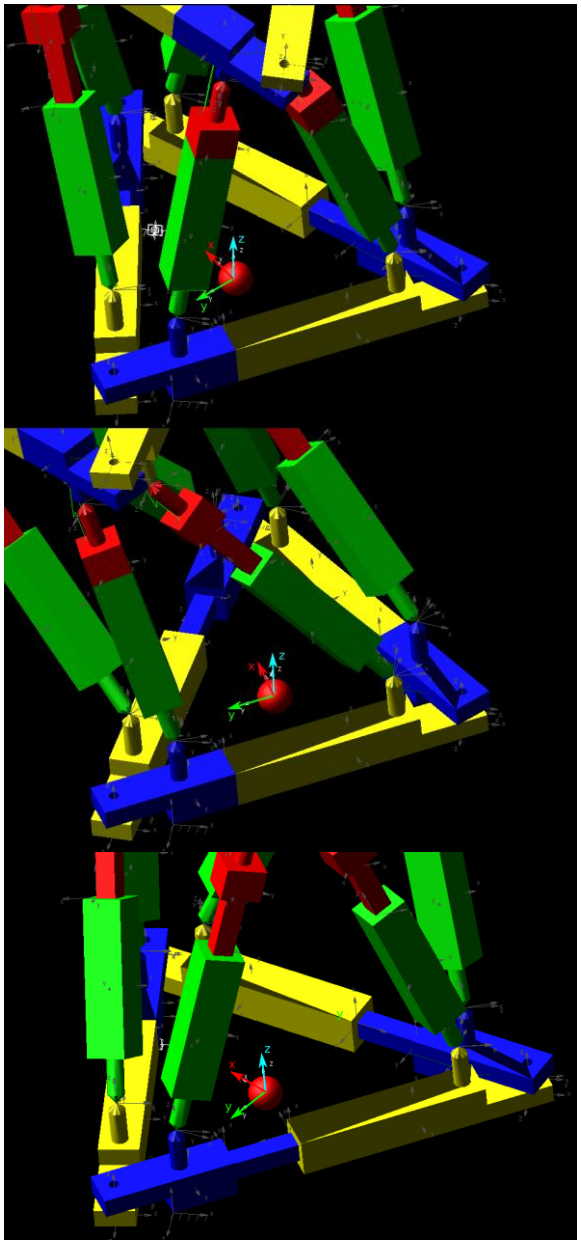


Figure 6.16 Screenshots showing the orientation and position of the bottom marker constrained to represent the robot's global axis system.

$$v = \overrightarrow{B_1} - \overrightarrow{BOTTOM\ ORIGIN} = \begin{bmatrix} B_{1_x} - ORIGIN_x \\ B_{1_y} - ORIGIN_y \\ B_{1_z} - ORIGIN_z \end{bmatrix} \quad (65)$$

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (66)$$

$$x \cdot v = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} B_{1_x} - ORIGIN_x \\ B_{1_y} - ORIGIN_y \\ B_{1_z} - ORIGIN_z \end{bmatrix} = B_{1_x} - ORIGIN_x \quad (67)$$

$$x \cdot v = \|x\|\|v\| \cos\alpha = \|v\| \cos\alpha \quad (68)$$

$$\therefore \|v\| \cos\alpha = B_{1_x} - ORIGIN_x \quad (69)$$

$$\therefore \cos\alpha = \frac{B_{1_x} - ORIGIN_x}{\|v\|} \quad (70)$$

6.3.3.2 Top Ring Orientation Definition

In order to constrain the top ring, a similar method to the one used to constrain the bottom origin was initially attempted. However, this proved to be a far more difficult task, primarily due to the fact that, unlike the bottom ring, the top ring does not remain horizontal during the simulation. Also, the built in functions that are available for use when creating GCONs (run-time functions) are less than those that can be used when constructing the model at design time (design-time functions) and are generally less sophisticated. An example of this is the ORI_IN_PLANE function, which can only be used at design time. This function allows the axis system of a marker to be completely defined in one call by specifying the direction of two of the vectors. Unfortunately, this function is not available during run-time and as such, the manual method outlined in the last section has to be used.

In order to work around this, it was decided to simplify the problem somewhat. It was realised that the orientation of the top origin's x-y plane is primarily dictated by the extensions of the top ring actuators. Therefore, if those actuators were to remain fixed during the simulation, the orientation problem could be simplified. If the marker were to be attached to one of the top actuators and thus always remain fixed relative to it, then as that actuator moved and rotated along with the top ring, the marker would move and rotate accordingly and remain in the correct position and orientation relative to that actuator.

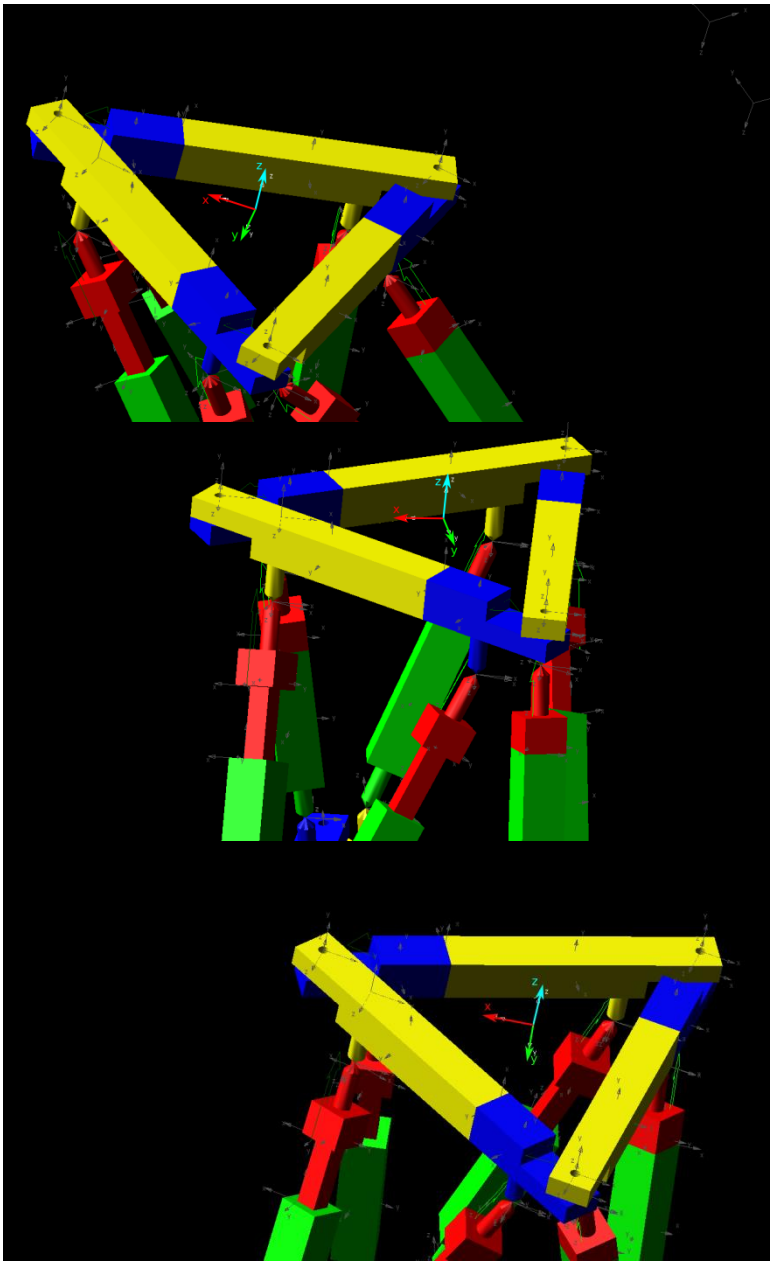


Figure 6.17 Demonstration showing how the top origin constraints correctly orient the top origin marker relative to the top plane.

Therefore, the top marker was constrained as follows:

1. The marker was created and then attached to one of the top ring actuators.
2. The marker was positioned at the centroid of the top ring at design time.
3. The ORI_IN_PLANE function was used to correctly define the orientation of the marker at design time.
4. The GCONs constraining the top origin position were disabled as this was now handled by steps 2 and 3.

5. During simulations, the top ring actuators remained at their same fixed extension at design time.

Using the above method, the top origin could be correctly defined, on the condition that the top actuators remain at the same extensions as they were at design-time. Figure 6.17 shows how the top origin remains at the correct position at various robot configurations.

6.3.4 Simulation

At this point, the model is complete and can be used to verify the IK equations. This was done by inputting a wide range of actuator extensions into the model and letting the simulation solve the forward kinematics problem and orient the robot. The extensions were generated using custom-written software and were designed to produce all actuator combinations where the actuators are either fully retracted or fully extended. Table 6.3 shows a sample of the input data.

Table 6.3 Sample linear actuator input data used in the Adams simulation.

Actuator 1	Actuator 2	Actuator 3	Actuator 4	Actuator 5
0	0	0	0	20
0	0	0	20	0
0	0	0	20	20
0	0	20	0	0
0	0	20	0	20

The data was then imported into Adams as a set of splines, which were then applied to the linear actuators as motions, with the value of the spline representing a displacement of the linear actuator. The simulation software interpolates in between each value, resulting in a smooth motion and far more extension combinations than would have been generated just from the imported data. Figure 6.18, Figure 6.19 and Figure 6.20 show the graphical representation of one of the splines, the interpolation between the points and how it is applied to an actuator as a motion.

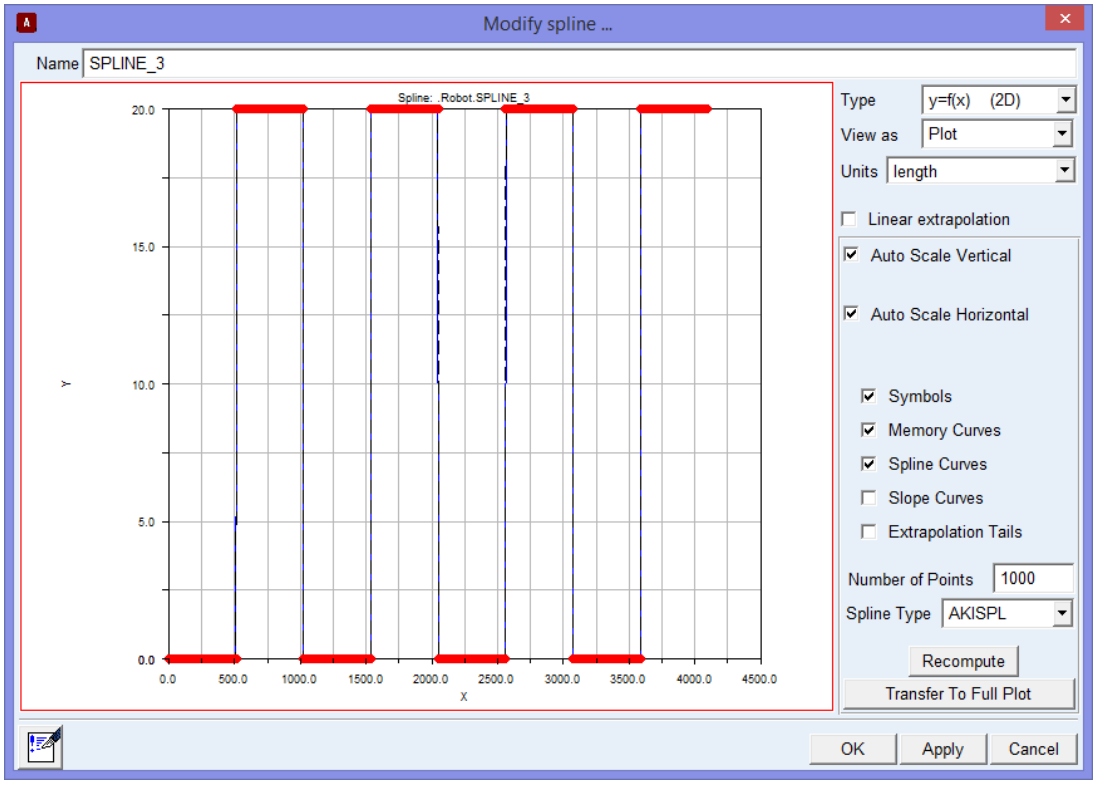


Figure 6.18 One of the splines imported into Adams.

The vertical scale represents the actuator extension in millimetres, whereas the horizontal scale represents the simulation time in seconds.

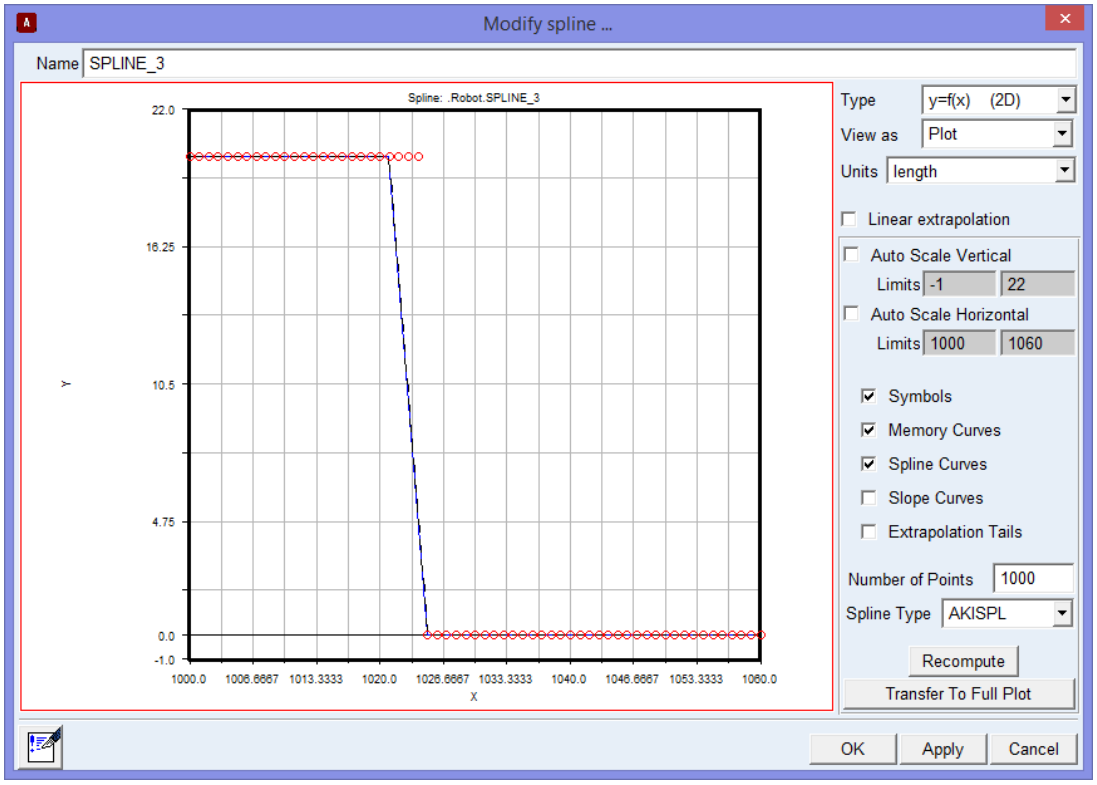


Figure 6.19 Interpolation between points on the spline.

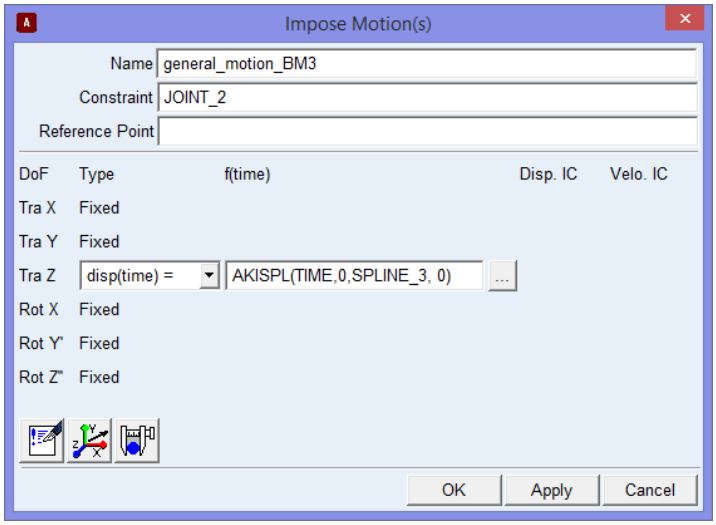


Figure 6.20 The splines being applied to the actuators as motions.

The simulation was then run for the entire time duration covered by the input data. The location and orientation of the top origin marker relative to the bottom origin marker were recorded and exported as tabular data.

The PC software was then modified to accept the tabular data and then for each row, extract the target position and orientation values and the top and bottom ring extensions, use the IK equations to determine the extensions of the Gough-Stewart actuators and then output those into the same file. The updated file was then loaded into a spreadsheet and the difference between the Gough-Stewart extensions input to the Adams simulation and the extensions calculated by the IK equations compared. The statistics shown in Table 6.4 were then calculated.

Table 6.4 Simulation validation statistical summary.

Number of data points	40961
Average extension deviation	1µm
Max extension deviation	53µm

As can be seen from the data, the IK equations match the simulation results very well. The deviations are most likely due to the limited accuracy and resolution of the numerical simulation.

As the simulation also generated a lot of positional information, it was decided to perform some analysis on that in order to understand the movement limits of the robot. A summary of the results is shown in Table 6.5.

Table 6.5 Summary of the robot's position and orientation extents.

Target X value range	-52.6 to 50.0
Target Y value range	-52.8 to 50.4
Target Z value range	120.8 to 151.4
Target Yaw value range	6.20° to 116.9°
Target Pitch value range	-21.9° to 22.2°
Target Roll value range	-21.9° to 21.6°

As can be seen from Table 6.5, the robot's movement range is symmetrical, as would be expected. However, that data alone does not paint an accurate picture of the robot's movement range, as the robot is unable to reach all of the extremities of each axis at the same time. In order to visualise this, a 3D plot was created of the target positions measured during the simulation in order to view the robot's movement envelope. This is shown in Figure 6.21.

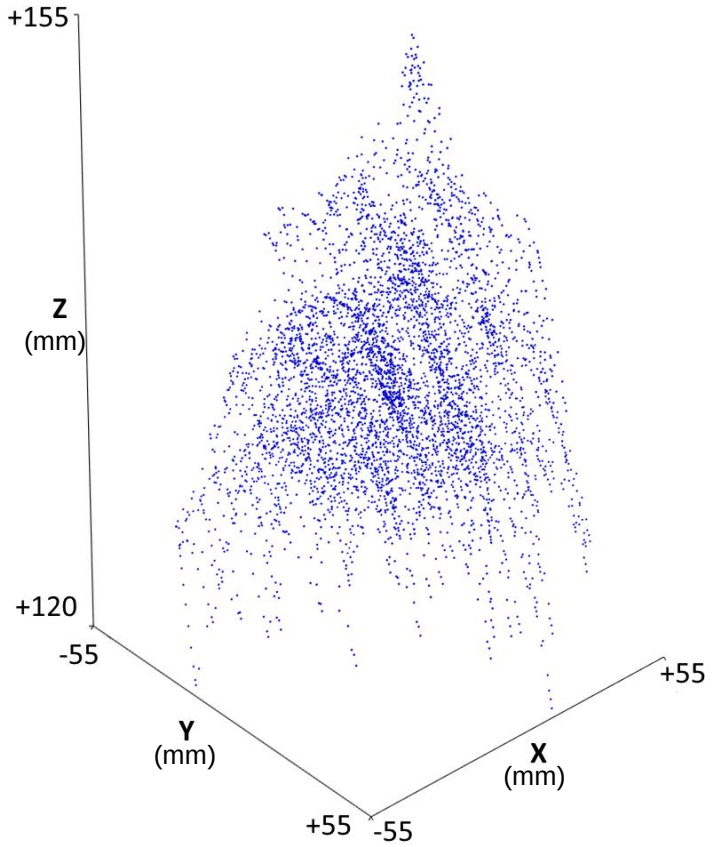


Figure 6.21 3D scatter plot of the simulated target positions.

As the visualisation shows, the extreme values for each axis only occur when the other axes are at their minimum or zero positions. The robot

appears to have the most fidelity in the central region of the visualisation where the points are clustered close together, indicating that the robot is able to position itself more freely and is able to reach the same target position from slightly different configurations. Figure 6.22 shows a density plot of the target data, highlighting the areas that the robot was able to reach from multiple positions. Areas in red indicate that those regions can be accessed via many configurations, whereas areas in blue can only be accessed via very few.

Based on these visualisations, a primary working area can be defined as a trapezoidal area that is a subset of the entire working envelope, inside which the robot has the most movement flexibility. Figure 6.23 shows the definition of this area and its properties are summarised in Table 6.6.

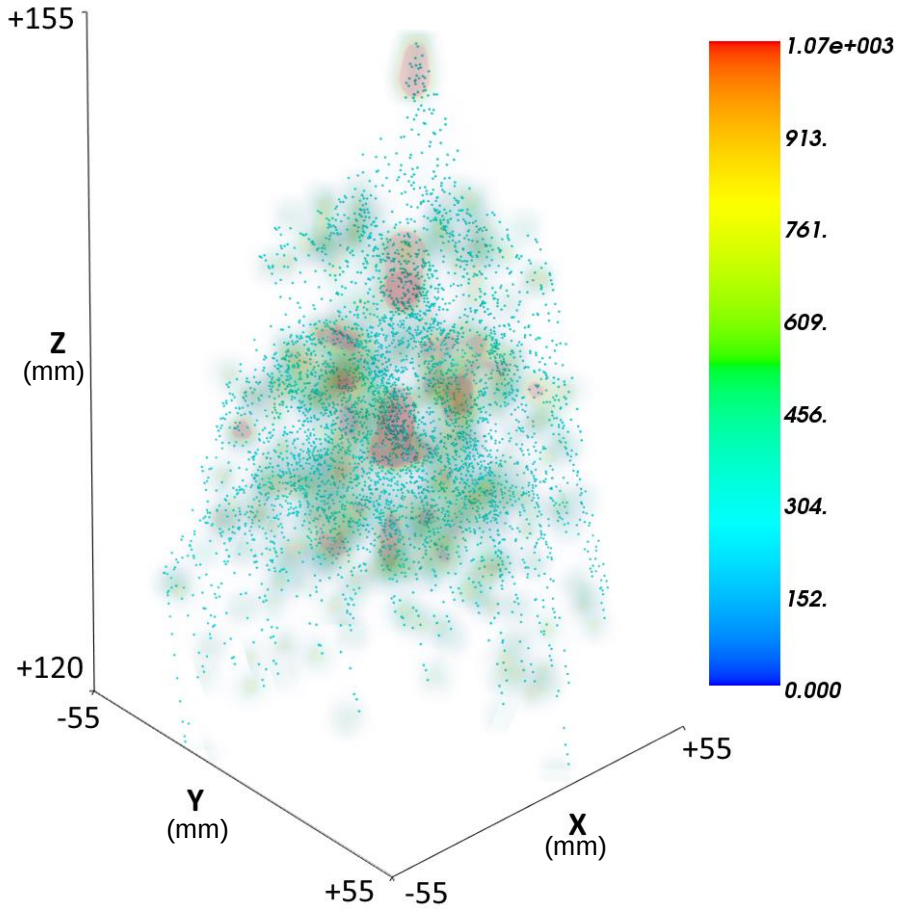


Figure 6.22 Density map of the robot's target positions, highlighting the areas that could be reached from multiple configurations.

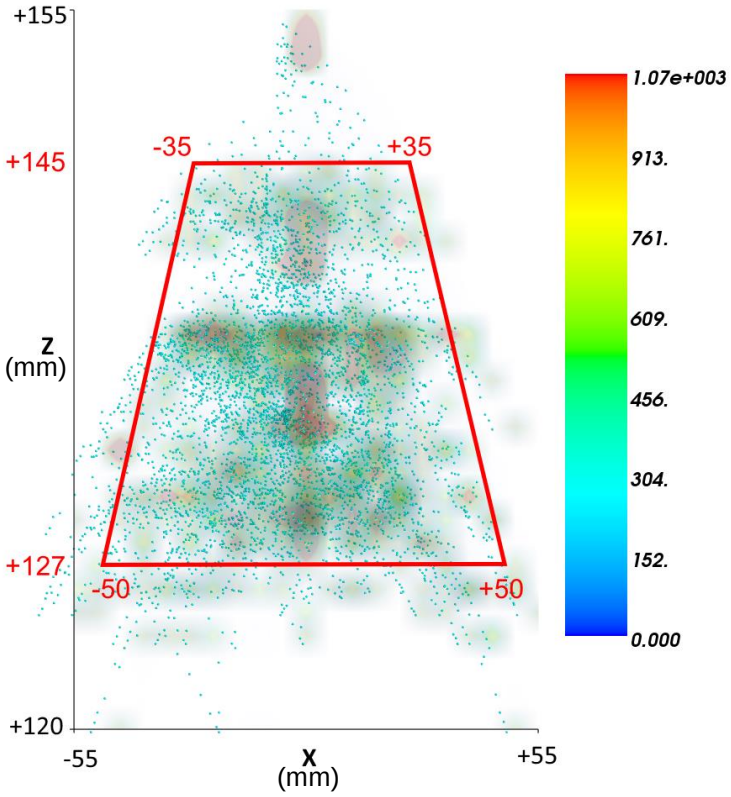


Figure 6.23 Primary working space of the robot.

Table 6.6 Definition of the robot's primary working area, in terms of the valid range of x, y and z values.

Z Height	X working range	Y working range
127mm	±50mm	±50mm
145mm	±35mm	±35mm

In summary, the simulation helped to increase the confidence that the derived inverse kinematics equations are an accurate representation of the robot's geometry and can be used by the control system to control the robot's actuators. The results of the simulation were also useful in visualising the robot's working envelope and its limitations.

6.4 Bend Radius Determination

In this section, the process used to determine the robot's minimum bend radius is presented. This includes the theoretical calculations, followed by the simulation and analysis of the results.

In order to demonstrate the robot's ability to navigate bends, a swept bend section of pipe needed to be purchased or manufactured. However, it was important that the bend radius of the pipe be greater than the minimum bend

radius that the robot prototype is capable of navigating. The bend radius of a pipe bend is defined as the radius of the circle that forms the inside bend of the pipe. Figure 6.24 shows how the bend radius is measured in a pipe bend.

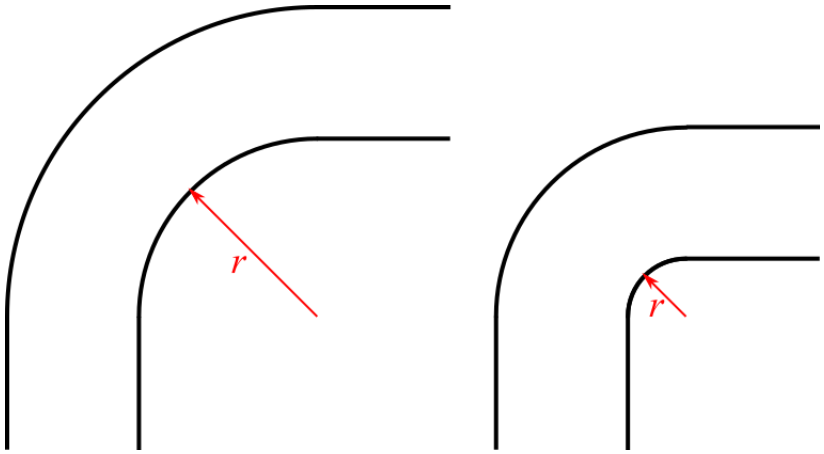


Figure 6.24 The definition of a bend radius on a pipe, showing examples of two bend radii.

As can be seen from the figure, the larger the bend radius, the gentler the bend and vice versa. It is apparent that if the robot is capable of navigating a particular bend radius, it will also be capable of navigating all bend radii larger than that. The smaller the bend radius that the robot is capable of navigating, the larger the range of pipe bends that it can move through, giving it more mobility.

The robot's minimum possible bend radius is limited by its geometry. This limitation is governed by the following factors:

- The ratio of the robot's width and height.
- The maximum angle that the top ring can form relative to the bottom ring.

The ratio of the robot's width and height is important as it determines whether a robot will get "stuck" in a pipe as it is trying to navigate the bend. Figure 6.25 demonstrates how a robot with a large height to width ratio would be unable to navigate a bend, whereas a robot with a smaller height to width ratio would.

Similarly, the maximum angle that the top ring can form limits the sharpness of bend that the robot can navigate. If the robot is able to form a large angle, this allows it to navigate sharp bends, whereas if it can only generate small angles, it will either have to navigate the bend in small steps, or will not be capable of navigating it at all. Figure 6.26 demonstrates this.

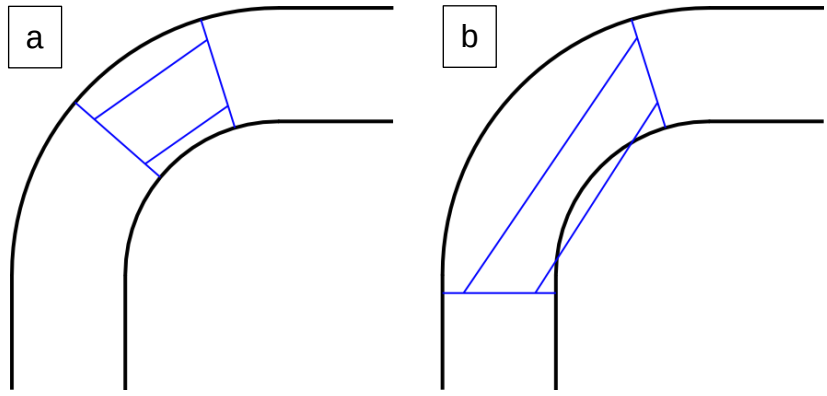


Figure 6.25 A robot with a large height to width ratio (b) has more difficulty navigating the same bend than a robot with a smaller ratio (a).

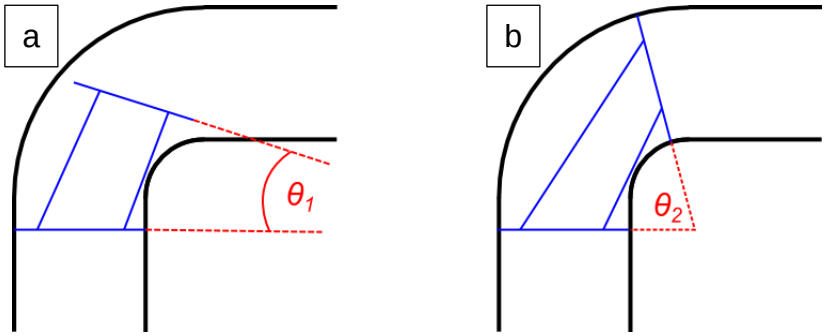


Figure 6.26 A robot with a larger maximum angle (b) can pass through tighter bends easier than one with a smaller maximum angle (a).

6.4.1 Bend Radius Estimation

In order to determine the robot's minimum bend radius, the geometry of the robot needs to be examined in order to calculate the tightest bend that the robot can fit in. Figure 6.27 shows the form that the robot takes inside a bend.

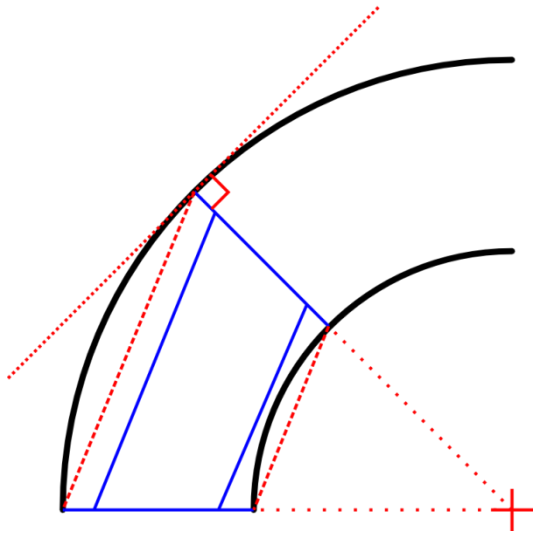


Figure 6.27 The robot's form inside a bend.

One constraint was made when considering the robot's motion inside the bend: when the robot's top and bottom rings make contact with the pipe wall in order to grip, they would always be perpendicular to the tangent of the pipe wall at that point. This ensures that the holding force generated by the top and bottom rings is used entirely to grip the pipe wall and no other force component is generated that could cause the rings to slip.

Due to this constraint, the top and bottom rings will always lie along a radius of the pipe and the outline of the robot forms a 2D isosceles trapezoid, with the two parallel edges representing the extensions of the Gough-Stewart actuators. The difference in length between the two parallel edges of the trapezium are what determine the angle between the top and bottom rings. The larger the difference, the smaller the angle and vice versa, as shown in Figure 6.28.

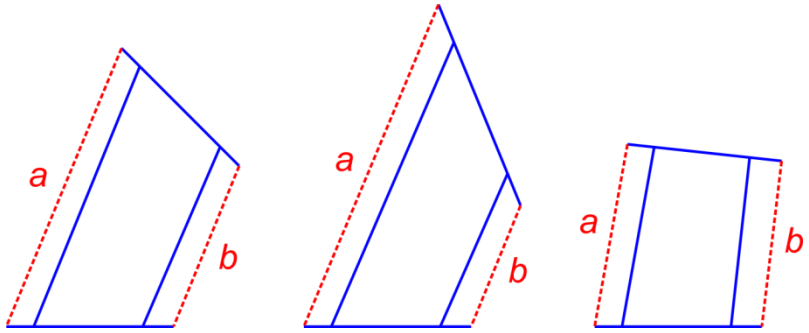


Figure 6.28 Diagram showing how the difference in the sides a and b affect the angle between the top and bottom platforms.

By inspection, it can be seen that in order to produce the largest angle (and hence the smallest bend radius), one of the sides needs to be at its maximum length, whereas the other needs to be at its minimum. Similarly, the sizes of the top and bottom rings also affect the angle. The larger they are, the smaller the angle and vice versa. In order to estimate the smallest bend radius it was decided to use the Adams model to measure the robot's height when the Gough-Stewart actuators are all fully extended and again when they are all fully retracted. This was done with the top and bottom rings at maximum extension to determine a bend radius that the robot could navigate for all pipe diameters. The theoretical maximum and minimum values for the lengths a and b in Figure 6.28 are summarised in Table 6.7.

Table 6.7 Robot heights when GS actuators are fully extended or retracted.

GS Actuator Condition	Robot Height
All fully retracted	125.6mm
All fully extended	147.5mm
Top/Bottom ring diameters	123.0mm

Based on these measurements, an estimate of the minimum bend radius can be calculated as follows:

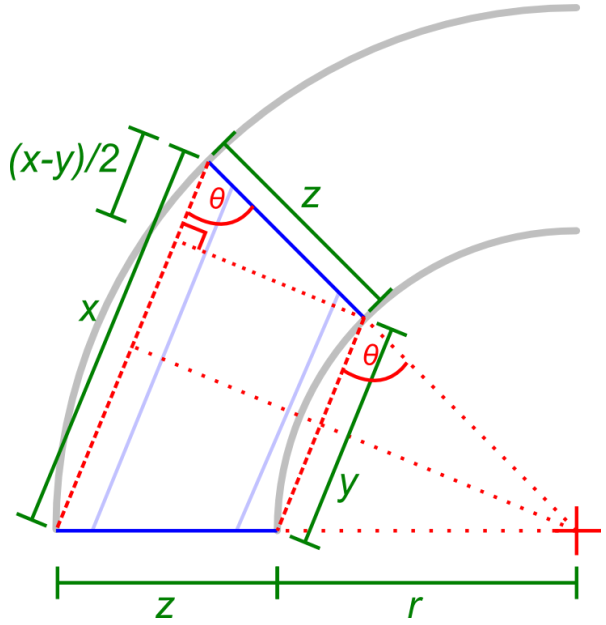


Figure 6.29 Geometrical representation of the robot inside a bend.

$$\cos(\theta) = \frac{x-y}{2z} = \frac{y}{2r} \tag{71}$$

$$\therefore r = \frac{yz}{x-y} \tag{72}$$

The values for the variables are defined below.

Table 6.8 Bend radius calculation variable definitions.

Variable	Description	Value
x	Robot outer height	147.5
y	Robot inner height	125.6
z	Ring diameter	123.0

Based on these variables, the minimum bend radius, r, is estimated to be 705.4mm.

6.4.2 Calculation via Simulation

The previously described method of estimating the pipe bend radius may not give a very representative value due to it greatly simplifying the geometry of the robot. The results of the Adams simulation presented in Table 6.5, for example, indicate that the robot is able to generate z-heights smaller than 125.6mm and larger than 147.5mm. The complex nature of the robot's geometry, the interlinking of the 6 Gough-Stewart actuators and their mutual effect on each other mean that it is not possible to consider the extensions of a subset of the actuators in isolation as they affect the behaviour of the other actuators and vice versa, and therefore the results of the estimation method described earlier were not expected to be very representative. In order to determine the minimum bend radius more accurately, it was decided to analyse the large amount of positional data that was generated previously using the Adams simulation.

For each position generated by the simulation, the corner positions of the top and bottom ring were extracted. Each set of corner positions represents the plane of the corresponding ring (Figure 6.30a). The following algorithm is used to determine the bend radius of each position:

1. The normal vectors for each of the top and bottom ring planes ($\widehat{n_t}, \widehat{n_b}$) are determined by calculating the cross product of two vectors that form each plane (Figure 6.30a).
    a. In order to determine whether the two planes are parallel or not, the cross product of the two normals needs to be calculated. If the magnitude of the result is 0, then the top and bottom rings are parallel and this position is discarded. Otherwise, the algorithm continues on to the next step.
2. The angle between the planes can be calculated by determining the angle between the two plane normals $\widehat{n_t}$ and $\widehat{n_b}$. This is done by calculating the dot product of the two normals.
3. The intersection line between the two planes then needs to be calculated. This is composed of two components: the direction vector, $\hat{v}$, and a point on the line, p (Figure 6.30b).
    a. The direction vector $\hat{v}$ is the result of the cross product of the two plane normals $\widehat{n_t}$ and $\widehat{n_b}$.

   b. In order to find a point on the line, the plane equations of the two planes are derived and are equated together to find a point common to both.
   c. The results of the previous two steps define the intersection line of the two planes in 3D space.
4. The distance between the top and bottom ring centres and the intersection line (d_t, d_b) now need to be determined (Figure 6.30c).
   a. The ring centres are calculated by averaging the 3 coordinates that make up each ring.
   b. The distance between each centre and the intersection line is calculated as follows (Figure 6.30d):
      i. The vector, w, between the ring centre and the point p on the intersection line is determined.
      ii. This vector is then projected on to the intersection line using the dot product, resulting in u.
      iii. The projected vector u is then subtracted from the original projector vector w. The length of the resultant vector is the perpendicular distance from the ring centre to the intersection line.
5. If this robot position represents a valid position inside a curved pipe, then the distance between each of the top and bottom ring centres and the intersection line (d_t and d_b) should be equal. This allows all invalid positions to be discarded by keeping only those robot positions where the difference between the two distances is less than a particular threshold (Figure 6.30e).
6. The bend radius is calculated by subtracting the ring centre-intersection line distance from the ring radius.

By processing all the positions generated by the simulation using the above algorithm, only the robot positions representing a valid position inside a bent pipe remain.

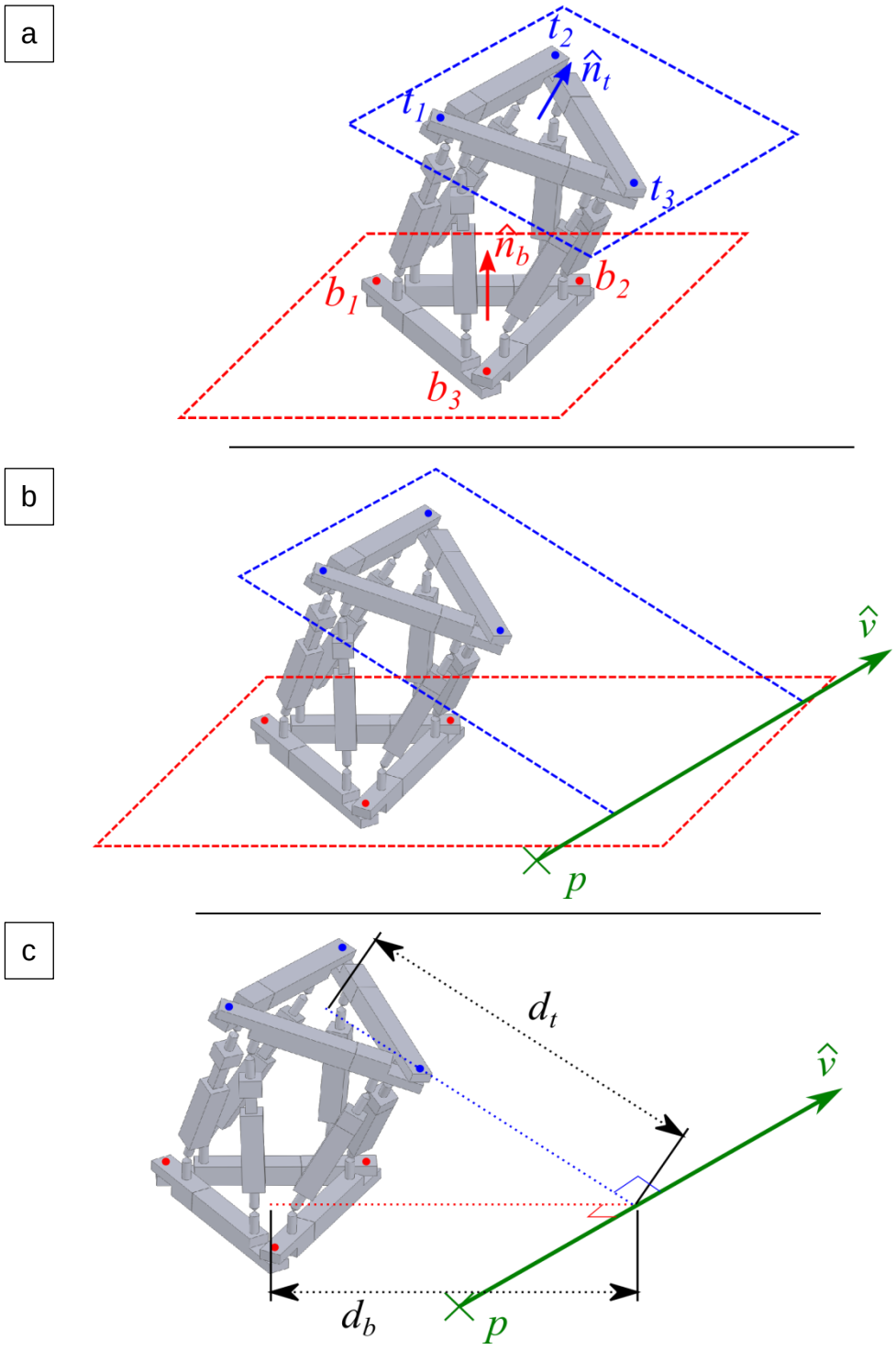


Figure 6.30 continued on the next page.

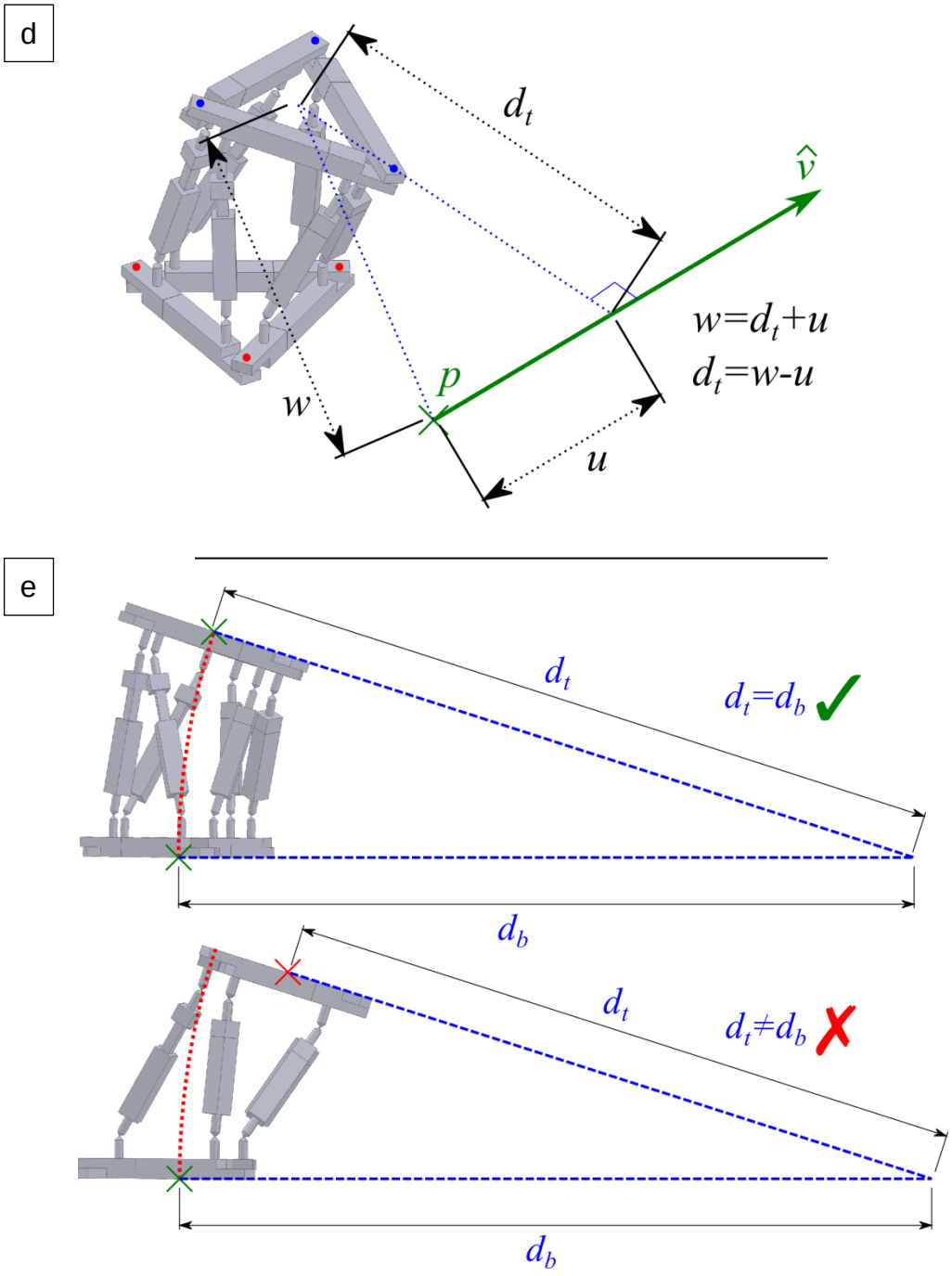


Figure 6.30 Various diagrams to illustrate the algorithm for determining valid bent positions.

This algorithm was applied to the simulation data, filtering out all positions where the top and bottom differences were greater than 1mm. Once the data was processed, a graph of the bend radius vs the angle between the two planes was plotted. The graph is shown in Figure 6.31.

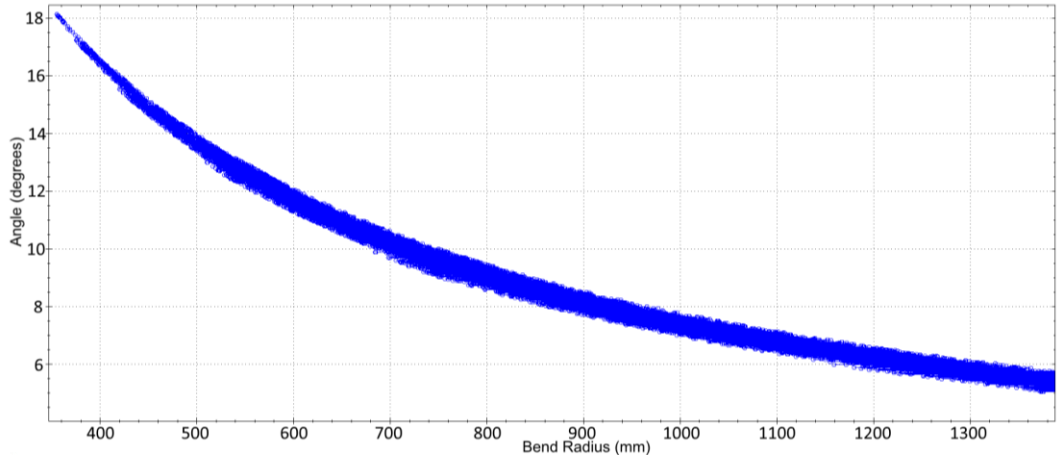


Figure 6.31 Graph of bend radius vs top/bottom ring plane angles.

As can be seen from the figure, the range of possible bend radii ranges starts from approximately 350mm. However, what is interesting to note from the graph is that the range of angles at that bend radius is very small, indicating that there are very few positions that the robot can take that will properly fit in the pipe. As the bend radius increases the range of angles gradually increases, reaching a maximum of approximately 1 degree around 700-800mm radius. The range then starts to very slowly decrease as the bend radius increases further.

The decrease in angle range as the bend radius becomes smaller is to be expected as the robot will be expected to have less manoeuvrability in a very tight bend than in a gentler one. It is expected that the smaller the angle range, the smaller the step size that robot would be able to take in order to navigate that bend. The very gradual decrease in range as the bend radius continues to increase was not expected and may be due to the limited resolution of the simulation. As can be seen from the plots shown in Figure 6.21 and Figure 6.22 the spread of the data points is not always uniform. Therefore it is possible that with more data points, the gradual decrease in angle range may be different.

Regardless, the results of the simulation analysis show the range of bend radii that the robot would be capable of navigating as well as an indication of its performance at each radius.

6.5 Fluid Force Analysis

One of the important design criteria of the robot is its ability to withstand the force of the fluid flowing past it and to be able to maintain its position inside the pipe and move against the flow. This ability stems from two main sources:

- Maximising the frictional force that the robot can generate using its ring actuators.
- Minimising the drag force generated by the flow of fluid past the robot.

The optimisation of each of those criteria can be considered a project in and of itself. Optimising the frictional force involves studying the conditions of the inside surface of the pipe and then using the principles of tribology and materials science to fashion gripping material that maximises the frictional force. Optimising the drag force involves studying the dynamics of the fluid as it flows past the robot and then using that information to optimise the mechanical design in order to minimise it.

For this project, these two topics were not explored in their full detail as the prototype was not to be tested in a live water situation. However, several simulations were carried out in order to estimate the forces exerted on the robot due to the fluid flow and thus provide an indication of the feasibility of the robot operating in a live water pipe. The construction of the simulation models and the analysis of the simulation results are detailed in this section.

6.5.1 Model Construction

In order to assess the performance of the robot inside a pipe network, several simulations were built that placed the robot in various pipe configurations in order to examine how the forces exerted on the robot varied depending on the pipe network. The following scenarios were examined:

1. Robot inside a straight pipe.
2. Robot in various positions while navigating a bend.
3. Robot in various positions while navigating a junction.
4. Robot inside a straight pipe with a butterfly valve set to various positions

Scenarios 1-3 demonstrate the most common scenarios the robot is expected to encounter in a pipeline. The butterfly valve scenario explores the case where the flow of fluid is significantly disturbed, and whether that disturbance has a significant effect on the drag force.

The simulation was modelled as a fluid-solid interaction simulation. This requires that both the fluid and robot volumes are modelled and then assembled in their correct relative positions. The boundary conditions could then be defined as well as the fluid-solid interface.

6.5.1.1 Robot Volume

A simplified model of the robot was created in Solidworks that had the same primary dimensions as the actual robot, but with many of the unnecessary details removed (such as the detailed shape of the linear actuators), so as to keep the simulation simple. Figure 6.32 shows both the simplified and original models side by side.

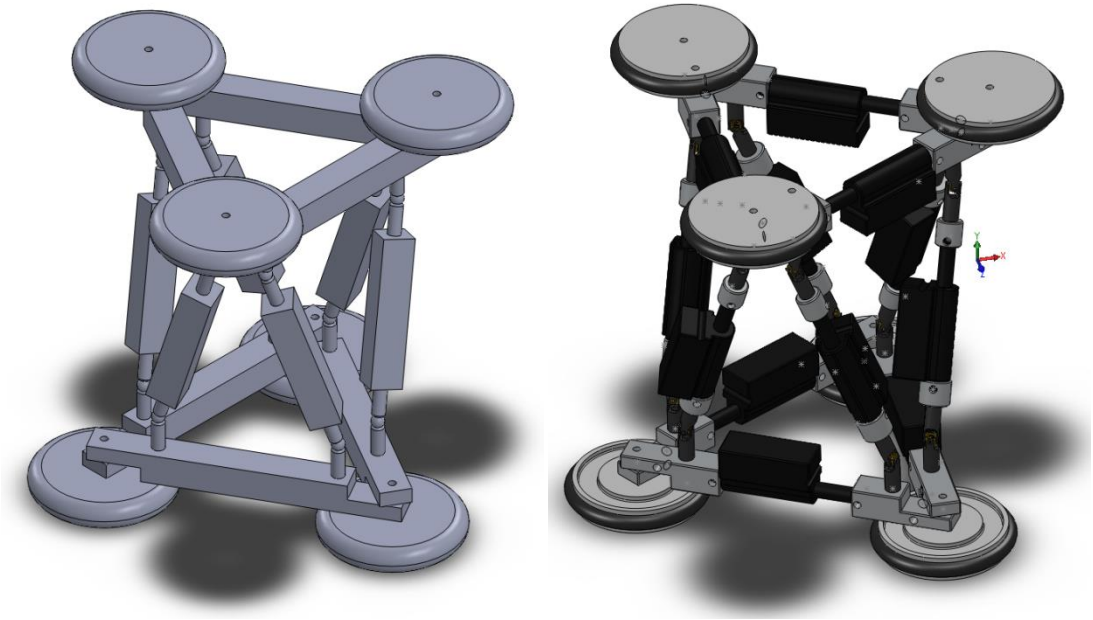


Figure 6.32 Comparison of the simplified robot model used in the CFD simulation (left) and the original robot model (right).

6.5.1.2 Fluid Volume

Models that represented the volume of fluid inside the pipe needed to be created. The creation of these models involved two main steps: to model the internal shape of the pipe, and then to also carve out the physical space occupied by the robot and any other obstacles inside the pipe. This generates the actual physical space occupied by the fluid, bound by the confines of the pipe walls and excluded from the physical space occupied by the robot and any other obstacles. Figure 6.33 shows an example of the pipe volume for one of the simulation scenarios, where the physical space occupied by the robot and a butterfly valve is carved out from the internal space of the pipe.

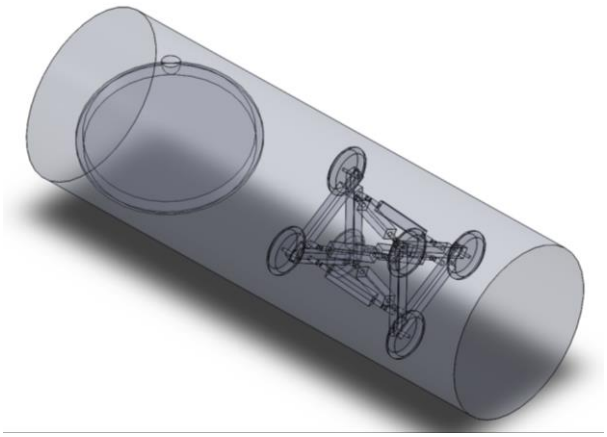
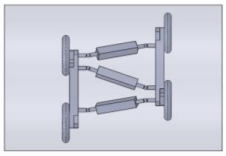


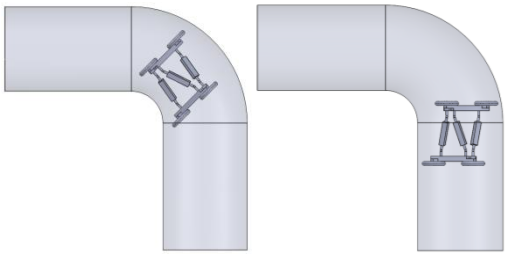
Figure 6.33 3D model showing the volume occupied by the fluid, excluding the physical space of any obstacles.

6.5.2 Simulation Scenarios

Table 6.9 shows the models that were constructed and simulated. For all these models, the boundary conditions defined in Table 6.10 were applied.

Table 6.9 Contructed simulation models.

Robot in straight pipe	

Robot in bend:
• Robot entering bend
  ○ Inlet near robot
  ○ Outlet near robot
• Robot in middle of bend
	

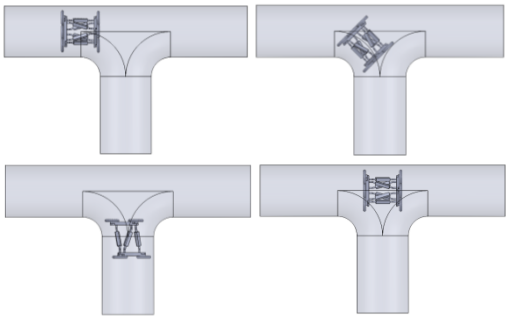
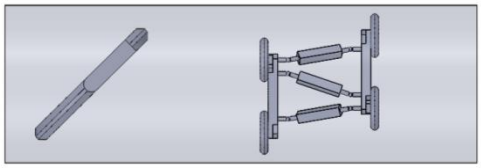
Robot in junction • Inlet at main section on the left. • Inlet at branch.	
Robot in straight pipe with butterfly valve at various positions: • 22.5° • 45° • 67.5°	

Table 6.10 Boundary conditions applied for all CFD simulations.

Pipe Diameter	200mm
Inlet Pressure	8bar
Outlet mass flow rate	0.8m/s at 200mm pipe = 25.1L/s, which for water is 25.1kg/s For the T-junction, the mass flow rate is split evenly between the two outlets.

6.5.3 Turbulence Model

The first step in preparing for the simulation was to choose the turbulence model that would be used in it. In order to do that, the Reynolds number for the simulation needed to be calculated in order to determine if the fluid flow will be turbulent or laminar. The Reynolds number can be calculated using the following equation:

$$Re = \frac{VD}{v} \tag{73}$$

Where:

- *V* is the velocity (0.8m/s)
- *D* is the effective diameter of the pipe (200mm)
- *v* is the kinematic viscosity of the fluid. (1.004E-6 at 20°)

The Reynolds value is therefore calculated to be 159362. Since this value is greater than 2300, it is considered to be high and therefore the flow will be turbulent. Therefore a suitable turbulence model needed to be selected.

In order to select an appropriate model, a literature search was carried out in order to find existing work that simulated a similar scenario and validated the simulation using experimental data. Several papers were found that explored the forces and flows around complex mechanical valves inside pipes in three dimensions [124]–[126]. It is believed that the work carried out in those papers can be applied here due to the following similarities:

- All scenarios deal with simulating non-simple geometries.
- The range of fluid pressures, flows and pipe diameters encompass the values being used in this scenario.

For all papers, the k-epsilon turbulence model was used and was found to generally provide good correlation with the experimental results. It was noted in [126] that when the valve was almost fully closed, the simulation failed to

correlate well with the experimental data. Because the robot does not restrict the majority of the pipe's internal space for this application, this observation does not apply here.

Based on this, it was decided to also use the k-epsilon turbulence model for these simulations.

As per the recommendations of the CFX User Guide, the turbulence intensity at the inlet is set to be Medium (Intensity = 5%), due to the fact that the inlet turbulence is unknown.

6.5.4 Mesh Independence Study

After the physical simulation environment was constructed and the turbulence model determined, the appropriate mesh size needed to be determined. This is done by running a simulation using several different mesh sizes and comparing the results produced by each mesh. As the mesh size gets smaller, its effect on the simulation also reduces to the point where changes in the mesh size result in negligible changes in the result. This helps to determine the maximum mesh size that can produce acceptable results.

For this simulation, it was necessary to use an adaptive meshing algorithm that varies the mesh element size depending on the complexity of the geometry. This is because the simulation model combines complex geometry and a large size. The complex geometry requires a very small element size in order to accurately capture the geometry features. However, using the same size throughout the entire model would result in a very large number of elements. This would make running the simulation impractical. Therefore, an adaptive meshing algorithm was used. An example of the mesh generated by the algorithm can be seen in Figure 6.34. The algorithm allows the minimum element size to be specified and it was this variable that was changed for the mesh independence study.

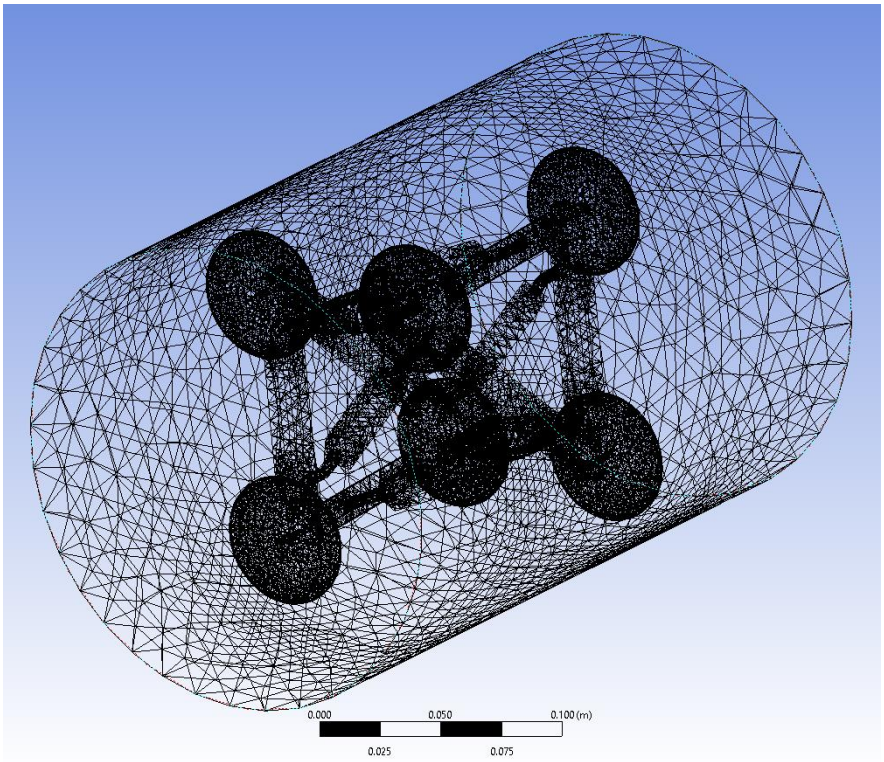


Figure 6.34 Mesh generated by the adaptive meshing algorithm.

The algorithm varies the mesh element size depending on the geometry complexity. The simple geometry of the pipe volume can be satisfactorily modelled using a large element size, whereas the complex geometry of the robot requires a much smaller element size to represent it correctly.

Since the minimum mesh size is dependent on the geometry complexity, the mesh independence study needs to be carried out on the model with the most complex geometry. Because the robot has the most complex geometry and is used in every simulation model, any of the models could be used for the study.

The mesh independence study was carried out on the scenario where the robot is situated inside a straight, unobstructed pipe. This was simulated using the boundary conditions described in Table 6.10, but using the six different minimum mesh sizes listed in Table 6.11. For each minimum mesh size, the total force exerted on the robot along the pipe's longitudinal axis (Z-axis) was recorded, along with graphs of the pressure distributions inside the fluid along horizontal and vertical lines passing through the centre of the pipe, as shown in Figure 6.35.

Table 6.11 Minimum mesh size vs the force exerted on the robot.

Mesh Size (mm)	Force exerted along Z-axis (N)
0.05	9.5181
0.2	9.50485
0.6	9.46078
1.5	9.48328
2.5	12.5029

Graphs of the mesh size vs force and pressure distributions for each mesh size are shown in Figure 6.36, Figure 6.37 and Figure 6.38 respectively.

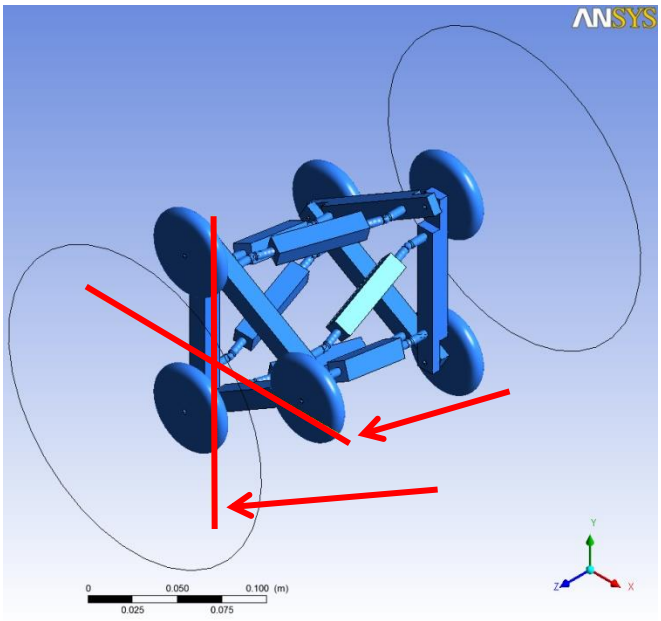


Figure 6.35 Screenshot showing the lines along which the pressure distribution was measured.

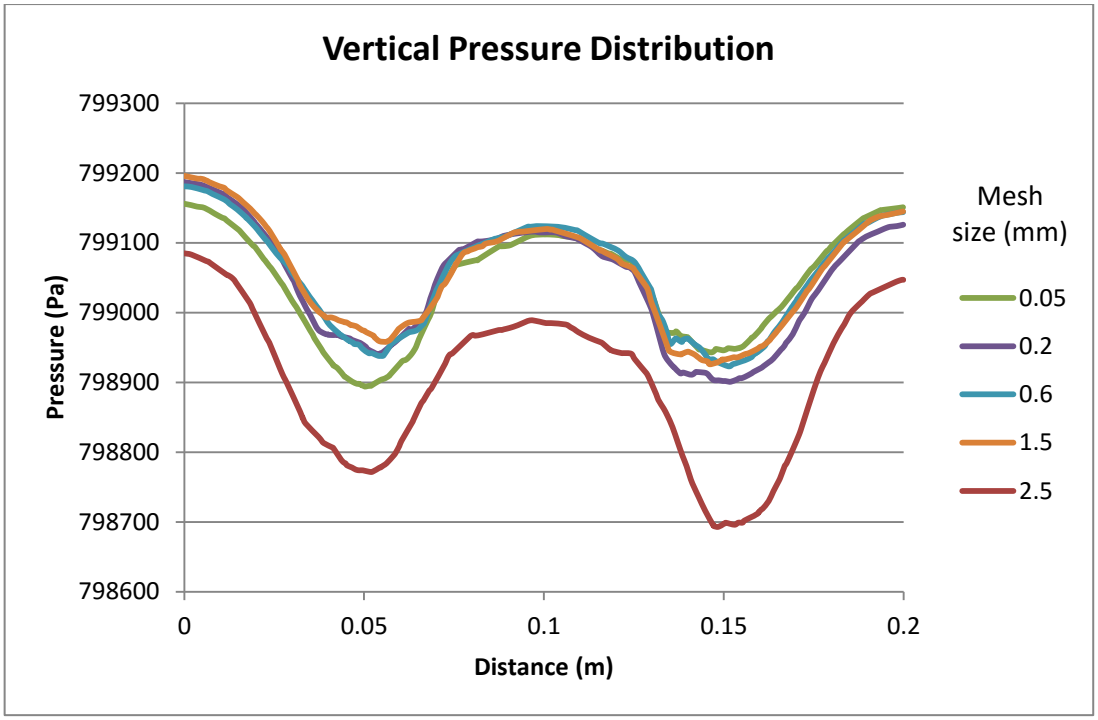


Figure 6.36 Pressure distribution along the vertical line for different mesh sizes.

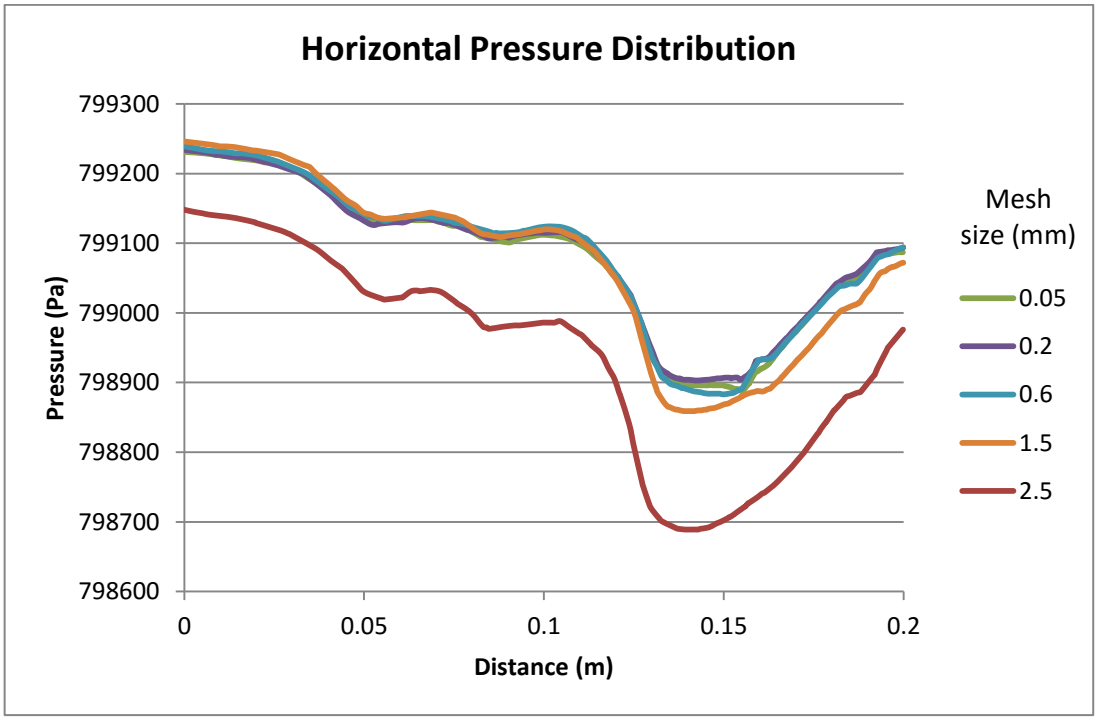


Figure 6.37 Pressure distribution along the horizontal line for different mesh sizes.

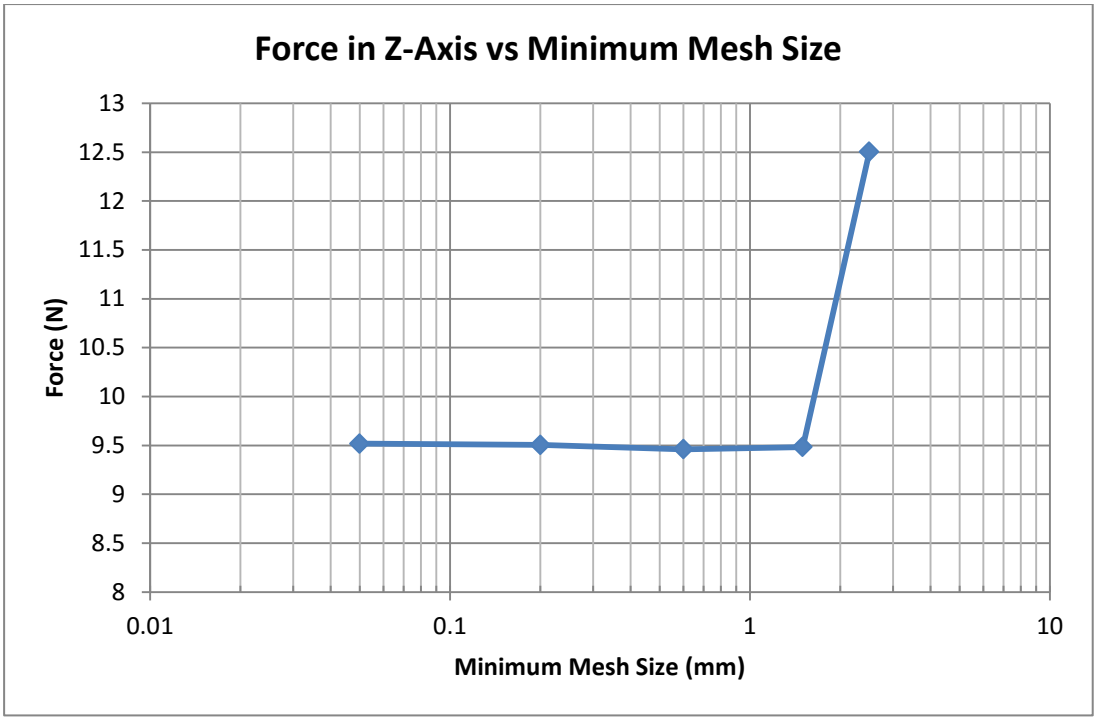


Figure 6.38 Variation in force along the Z axis with various mesh sizes.

As can be seen from Figure 6.38, the results converge when the minimum mesh size is 1.5mm or smaller. Discarding the results of the 2.5mm mesh, a summary of the maximum deviations compared to the average value for each graph is shown in Table 6.12.

Table 6.12 Summary of deviations for the mesh independence study.

Graph	Average Value	Maximum Deviation	Percentage Difference
Vertical Distribution	799032Pa	164Pa	0.0205%
Horizontal Distribution	799029Pa	214Pa	0.0268%
Force in Z	9.492N	0.03097N	0.326%

It is evident from the graphs that changing the minimum mesh size to values smaller than 1.5mm does not produce a significant change in the results. Regarding the force in the Z axis, at most, the results deviated by less than 0.5% from the average force calculated across all mesh sizes. A 0.5% potential error in the force calculation was deemed to be more than adequate for the purposes of determining the amount of force exerted on the

robot due to the fluid in order assess its feasibility. Therefore, it was decided to use the 1.5mm mesh size in all the remaining simulations.

Table 6.13 Summary of the forces exerted on the robot as measured by simulation.

Configuration	X (N)	Y (N)	Z (N)	Magnitude (N)
Straight pipe	0.08299	0.02800	9.483	9.484
Bend, robot entering near inlet	-0.4229	6.072e-4	-9.138	9.147
Bend, robot entering bend, near outlet	0.6069	0.04069	9.750	9.769
Bend, robot in middle of bend	-6.080	0.01643	-6.187	8.674
Tee junction, robot entering bend, inlet in main section	10.56	0.02446	0.4566	10.57
Tee junction, robot entering bend, inlet in branch	-3.048	-6.217e-3	-0.07917	3.049
Tee junction, robot in bend, inlet in main section	2.839	-0.06445	2.123	3.546
Tee junction, robot in bend, inlet in branch	-1.426	-0.1785	-1.483	2.065
Tee junction, robot in branch, inlet in main section	0.1480	7.454e-3	2.170	2.175
Tee junction, robot in branch, inlet in branch	0.1176	0.1689	-6.668	6.671
Tee junction, robot in middle, inlet in main section	2.877	-0.01486	0.1316	2.880
Tee junction, robot in middle, inlet in branch	-1.652e-3	0.01330	-0.5476	0.5477
Butterfly valve 22.5	1.390	1.536	8.480	8.730
Butterfly valve 45	0.8574	0.6236	9.707	9.765
Butterfly valve 67.5	0.6669	0.5192	9.501	9.539

6.5.5 Simulation

For each model along with its combinations of inlet and outlet configurations, a steady-state simulation was run. The postprocessor was then used to calculate the components of the total force exerted on the robot. These results are summarised in Table 6.13.

6.5.6 Analysis

The next step involved taking the maximum force value exerted on the robot of 10.57N and to calculate how much force is required by the Gough-Stewart actuators to push against it, and how much force is required by the ring actuators in order to generate the required frictional force that would hold the robot in place against that force. This was done in a similar manner to that described in sections 5.3.1.2.1 and 5.3.1.2.2.

6.5.6.1 Gough-Stewart Actuators

As described previously in section 5.3.1.2.2, the Gough-Stewart actuators are responsible for pushing the robot through the pipe and thus have to generate enough force to overcome the fluid force acting against the front of the robot. The force is split evenly across all 6 actuators, resulting in 1.762N being exerted on each actuator, and the largest angle they are expected to produce is 24°. Figure 6.39 shows a free body diagram showing a single actuator and the forces acting on it.

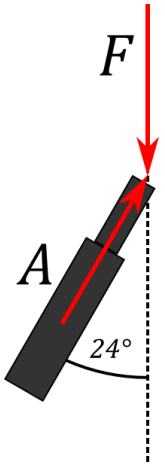


Figure 6.39 Forces acting on a single Gough-Stewart actuator.

The actuator force required to overcome the fluid force is calculated as follows:

$$F = A\cos(24) \tag{74}$$

$$A = \frac{F}{\cos(24)} = \frac{1.762}{\cos(24)} = 1.928\text{N} \tag{75}$$

6.5.6.2 Ring Actuators

The ring actuators are responsible for pushing the support discs against the pipe wall, thus generating a frictional force that holds the robot in place against the flow of water. The required actuator force is calculated in the same way as that shown previously in section 5.3.1.2.1. First, the total force required at each disc is calculated, then the components produced by each actuator are calculated. Figure 6.40 is a free body diagram showing the forces acting at a support disc.

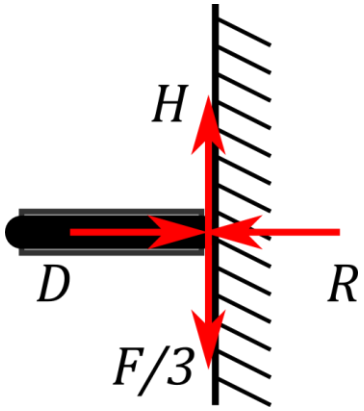


Figure 6.40 Free body diagram showing the forces acting on a single support disc.

The holding force, H, that holds the robot against the weight of the robot, F, is generated by the linear actuator force pushing the disc against the pipe wall, D, and the coefficient of friction between the rubber on the support disc and the pipe wall, μ, as governed by the following equations:

$$D = R \tag{76}$$

$$\frac{F}{3} = H = \mu D \tag{77}$$

As done previously, the coefficient of friction was taken to be 0.5. The actuator force is calculated as follows:

$$\frac{F}{3} = \mu D \therefore \frac{10.57}{3 \times 0.5} = 7.047\text{N} \tag{78}$$

The total disc force is split between two actuators as follows:

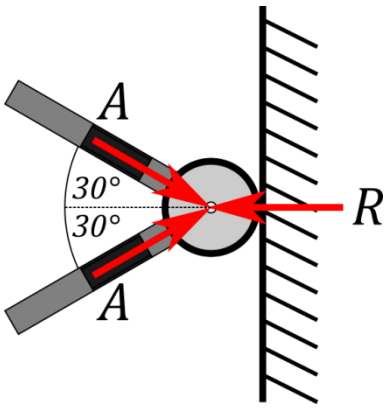


Figure 6.41 Free body diagram showing how the disc force is split amongst two actuators.

$$R = 2A\cos(30) \tag{79}$$

$$A = \frac{R}{2\cos(30)} = \frac{7.047}{2\cos(30)} = 4.069\text{N} \tag{80}$$

The required ring actuator force is therefore 4.069N.

6.5.7 Results

Table 6.14 summarises the effects of the fluid force on each of the ring and Gough-Stewart actuators. The table also includes the force values presented in Table 5.2, which indicate how the weight of the robot is transferred to the different actuators.

Table 6.14 Summary of the forces exerted on the actuators by the fluid and the weight.

Actuator Type	Force exerted due to fluid	Force exerted due to weight	Total force
Ring Actuator	4.069N	6.948N	11.02N
Gough-Stewart Actuator	1.928N	0.8233N	2.751N

As can be seen from the table, the ring actuators are exposed to the most force. This is because unlike the Gough-Stewart actuators, they are not directly working against the fluid force and the force is distributed amongst less actuators. The maximum force exerted on the ring actuators is expected to be 11.02N. When this is compared to the maximum actuator force of 20N of the PQ12 linear actuators chosen to power the prototype in section 5.3.1.3, it becomes evident that the operation of the robot inside an active pipe is very feasible, and that the required forces are within the

capabilities of off-the-shelf parts. The force required by the actuators could be reduced even further by optimising the following design variables:

- Minimising drag by reducing the robot's frontal area.
- Minimising drag by reducing the robot's drag coefficient.
- Reducing the weight of the robot.
- Engineering the support disc grip material to increase the coefficient of friction.

In summary, the results of the fluid force simulation, combined with the weight force calculations done previously, demonstrate that the possibility of running the robot inside an active pipeline is feasible and that off-the-shelf parts have the required power to overcome the resistance forces that the robot is expected to experience.

6.5.8 Level of Confidence

In order to determine whether the CFD simulation is an accurate representation of the forces exerted on the robot, it would need to be verified experimentally. Unfortunately, due to time and resource constraints this was not feasible.

The construction of the CFD simulation aimed to mitigate this by using similar parameters and the same turbulence model used in other published works that have been experimentally verified. Such works also aimed to measure the forces and pressures exerted on complex geometries inserted into pipelines and found that the simulation models were accurate representations. The large difference between the simulated forces on the linear actuators and their maximum load capacities further increases the confidence that, for the purposes of assessing feasibility, the results from the CFD simulation are sufficient.

6.6 Summary

In this chapter, the theoretical work that was carried out on the robot's design was presented. This covered three main aspects: the development of the robot's IK equations used in the control system along with their simulation validation, the determination of the robot's minimum bend radius, and the determination of the fluid forces exerted on the robot.

The inverse kinematics form the heart of the robot's control system as these define how the actuators should extend in order to reach a particular target position and orientation. First, the robot's axis and origin systems were defined, along with the main locations on its geometry. The IK equations

were then derived by examining the robot's geometry and determining how the target position and orientation affect the locations of the main points of the robot's geometry, which subsequently define the extensions of the 6 Gough-Stewart actuators.

In order to provide some preliminary verification for the IK equations, it was decided to use MSC Adams to construct a kinematic model of the robot. By solving the forward kinematics problem in a numerical fashion, the results could be compared to the IK equations, which were derived algebraically. The model was constructed by importing a simplified 3D model of the robot into Adams and then defining all of the joints. The next step involved defining the robot's axis and origin system in the simulation. This was done by defining several General Constraints in the simulation that defined how the location and orientation of the top and bottom origins were related to the rest of the robot's geometry.

The simulation was then used to numerically solve the forward kinematics problem. This was done by inputting a wide range of different extensions into the actuators and then recording the resulting target location and orientation. Each of these results was then fed back into the IK equations, which calculated the extensions of the GS actuators needed to reach those targets. Comparing the actuator lengths input into the simulation and the corresponding ones calculated by the IK equations showed that the equations matched the simulation model, which increased the confidence level in the equations' ability to correctly represent the robot's geometry. The data from the simulation was then used to map out the robot's working envelope by plotting the generated target positions on a 3D density plot and using that to determine the locations where the robot could move freely.

Next, the robot's minimum bend radius was determined. This was done by examining how the robot should position itself inside a bent pipe and then using the minimum and maximum robot height, as determined by the simulation model, to calculate a minimum bend radius. By representing the robot geometry as a isosceles trapezium, a minimum bend radius was estimated.

However, because this method greatly oversimplifies the robot's geometry, there was concern regarding its ability to adequately represent the robot. It was therefore decided to use the large amount of positional data that was generated in the simulation to determine the minimum bend radius. This was done by examining all the positions in software and extracting those that represented the robot inside a bent pipe. For a particular bend radius, more

than one valid position was found, with each position forming a different angle between the top and bottom rings. By plotting the bend radius against the range of angles that the robot was capable of forming, a graph was produced that showed the range of bend radii the robot was capable of navigating, along with an indication of how easily the robot could navigate each radius. Bend radii where the range of angles was large indicated that the robot would navigate those bends more easily.

The final theoretical work was the CFD simulation. As the robot would be designed to operate in an active pipe, the force exerted on it due to the fluid needed to be determined in order to assess the robot's feasibility. Several simulation models were constructed using a simplified model of the robot. These models placed the robot in various different pipe configurations, including straight, bent and tee junctions. The maximum fluid flow and pressure were used in order to get the maximum force on the robot. Once the maximum simulated force was found, its loading effect on the robot's various actuators was calculated. It was found that it was within the capability of off-the-shelf actuators to overcome this load. This showed that it is feasible for the robot design to operate in an active water pipe environment.

Chapter 7
Experimental Validation

7.1 Introduction

The aim of this chapter is to present the experimental work carried out in order to verify the theoretical and simulation work done on the prototype described in the previous chapters. This helps to prove that the robot behaves as calculated and allows the real-world practical limitations of the robot to be observed and quantified.

The experimental work verifies two out of the three theoretical aspects that were presented in Chapter 6: the mathematical models that govern the robot's movement (specifically, the IK equations) and the geometric limitations on the robot's movement (specifically, the minimum bend radius). The last aspect, the force exerted due to flow, was not verified due to resource and time constraints.

This chapter is split into two parts, with each part describing the experimental procedure, analysis and conclusions drawn from the experimental work performed on the two aspects mentioned above.

7.2 IK Equation Validation

Validating the IK equations is important as they form the primary mathematical model that governs how the robot moves and its physical position. The robot's control system relies on the IK equations to accurately represent the robot's geometry when it uses them to command the robot along a particular motion path.

In order to validate the equations, the theoretical robot position and orientation needed to be compared to the actual position and orientation when the results of the IK equations are sent to the robot's motor controllers. This was done by using a motion capture system to measure certain points on the robot and compare those to the their corresponding calculated positions.

However, before carrying out this experiment, the motor controllers themselves needed to be calibrated so that they can accurately position their corresponding linear actuators. Therefore, the actuator feedback calibration is presented first.

7.2.1 Linear Actuator Feedback Calibration

The aim of this experiment was to determine the relationship between the physical extension of the actuator and the resistance of the feedback potentiometer. The linear actuators come with an integrated feedback potentiometer, where the resistor is fixed inside the body of the actuator and the wiper is attached to the actuator's moving shaft. Extension and retraction of the shaft moves the wiper along the resistor's track, thus linking the extension of the actuator to the wiper resistance. Figure 7.1 shows a disassembled actuator with the feedback potentiometer mechanism visible.

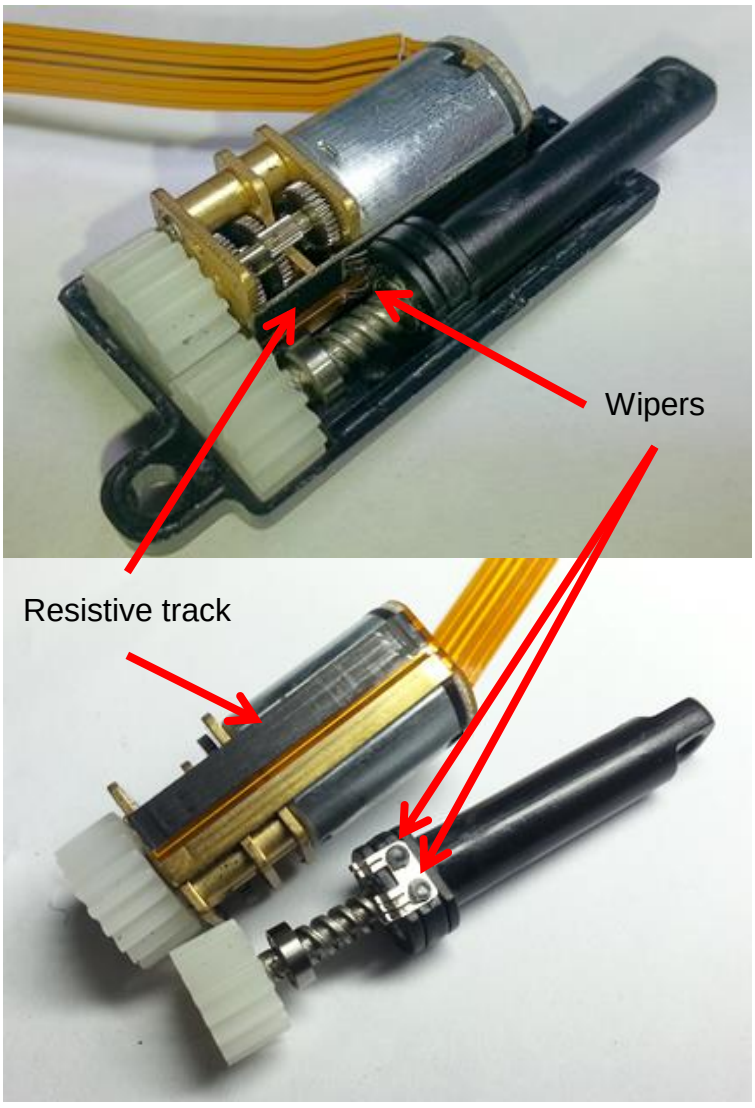


Figure 7.1 Disassembled linear actuator showing the potentiometer feedback.

The calibration was performed by extending the actuators to various lengths and recording the length and feedback value each time, giving a calibration curve that could then be programmed into the control software.

7.2.1.1 Preparation

In order to perform the calibration, two requirements needed to be met:

- A method of measuring the position of the wiper along the resistive track.
- A method of measuring the physical extension of the actuator.

In order to measure the wiper position, the actuator was connected directly to the motor controller, which provides a fixed 5V supply to the potentiometer and uses a 12-bit ADC to measure the wiper voltage, converting that to a number between 0-4095. The benefits of using this method to measure the wiper position, as opposed to measuring the wiper resistance directly are as follows:

- The resulting figure is completely independent of the actual resistance of the feedback potentiometer (which can vary from actuator to actuator), as it is based on the ratio of the wiper resistance and the total resistance. Thus, a value of 2048 will always represent a wiper at half the resistor value, regardless of whether the total resistance is 1k, 10k or 100k.
- Using the output from the motor controller ADC allows the characteristics of the ADC itself to be calibrated, thus relating actuator extension to the characteristics of the feedback resistor and the ADC circuitry.

In order to measure the extension of the linear actuator, the actuator was mounted vertically on a flat surface, and a height gauge used to accurately measure the total height of the actuator, as shown in Figure 7.2.

In order to measure the actual extension, the total height of the actuator when fully retracted is measured first, giving an offset. This offset is then subtracted from any subsequent measurements to give the actual extension.

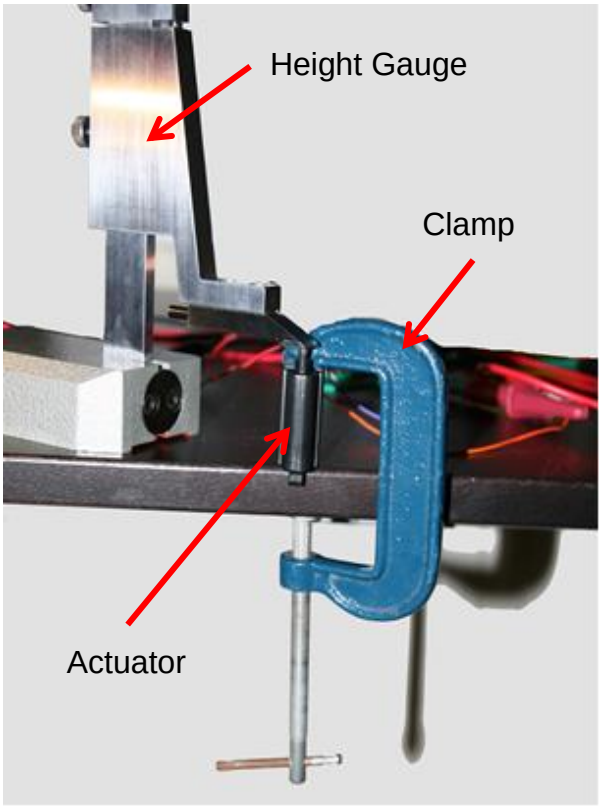


Figure 7.2 Photo showing the actuator extension calibration setup.

7.2.1.1.1 Feedback Scaling

Before performing the experiment, each motor is assigned to a dedicated motor controller board. Then, the motor controller's feedback scaling factor needs to be determined. The motor controller ADC measures the feedback voltage and converts it to a value between 0 and 4095. This value is then further scaled so that a value of 0 represents full retraction and 4095 represents full extension. This scaled feedback value is what is used by the controller's PID to determine if the motor has reached its target location. The scaling calibration is done via configuration software that is provided by the manufacturer (Figure 7.3). The process involves fully retracting the motor and then fully extended it so that the maximum and minimum feedback voltages can be determined and a suitable scaling factor calculated.

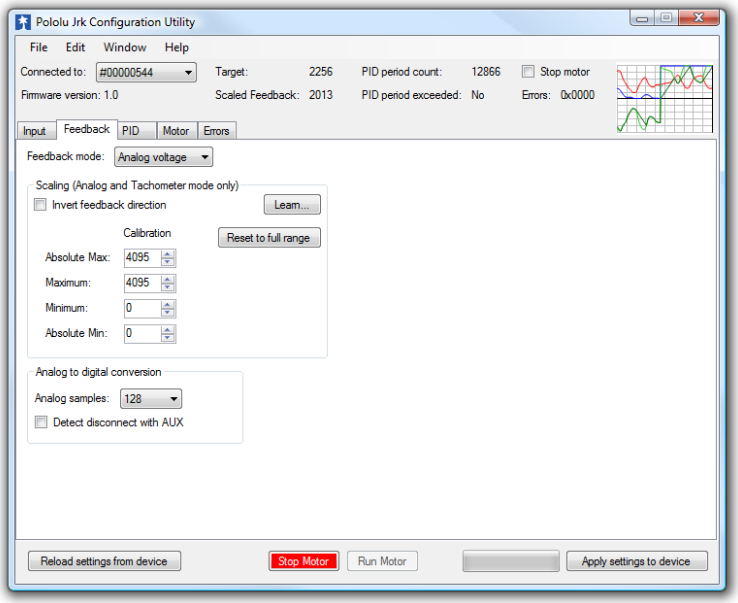


Figure 7.3 Motor controller configuration software for scaling actuator feedback [120].

7.2.1.2 Method

Once the feedback scaling has been performed, the motor is mounted on a flat surface and fully retracted. A height gauge is then used to take the initial height offset, which is recorded.

The 20mm working range of the actuator was split into 5, giving 6 points of measurement: 0mm, 4mm, 8mm, 12mm, 16mm and 20mm. For each point, the following steps were taken:

1. Set the height gauge to the target height required (the height does not need to exactly match the theoretical target height).
2. Slowly extend the linear actuator until it reaches the height gauge.
3. Remove power from the actuator to prevent the motor controller from making any small adjustments to the extension.
4. Use the height gauge to re-measure the total height of the actuator. Subtract the measured height from the offset to get the extension.
5. Record that extension along with the scaled feedback value, as reported by the control software.
6. Repeat for the other points.

Special care was taken when measuring the very first and very last points to ensure that the feedback value was neither 0 nor 4095. This was so that a measurement would not be taken when the feedback value was saturated. At those points, the actuator is either slightly extended or retracted until the feedback value is no longer in saturation.

The whole process was repeated three times and then tabulated.

7.2.1.3 Results

For each actuator, a graph of extension vs scaled feedback was plotted based on the tabulated data. The graphs for each actuator can be seen in Figure 7.4, Figure 7.5 and Figure 7.6.

Once the data was tabulated, the spreadsheet software could then be used to calculate the equation for the linear correlation line for each motor, relating the extension to the expected feedback value. The feedback values were then fed back into the correlation equations to get a theoretical extension and this was compared with the actual extension in order to assess the accuracy of the calibration equations. Table 7.1 shows the calibration equation for each motor along with the extension deviations.

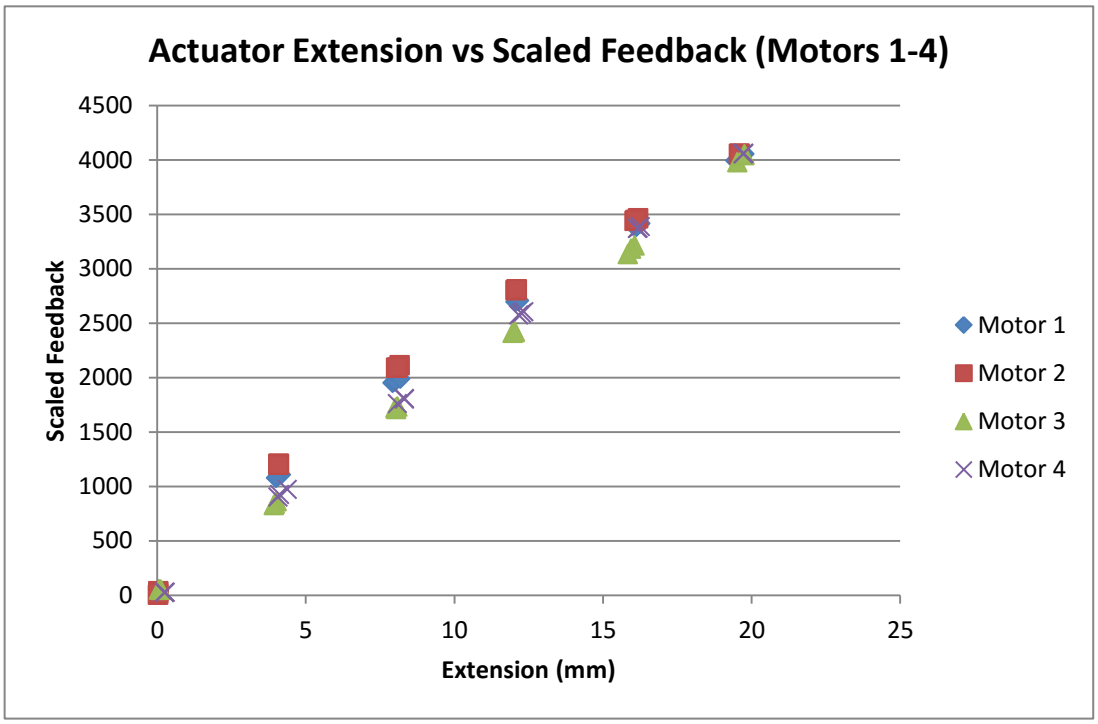


Figure 7.4 Actuator Extension vs Scaled Feedback (Motors 1-4).

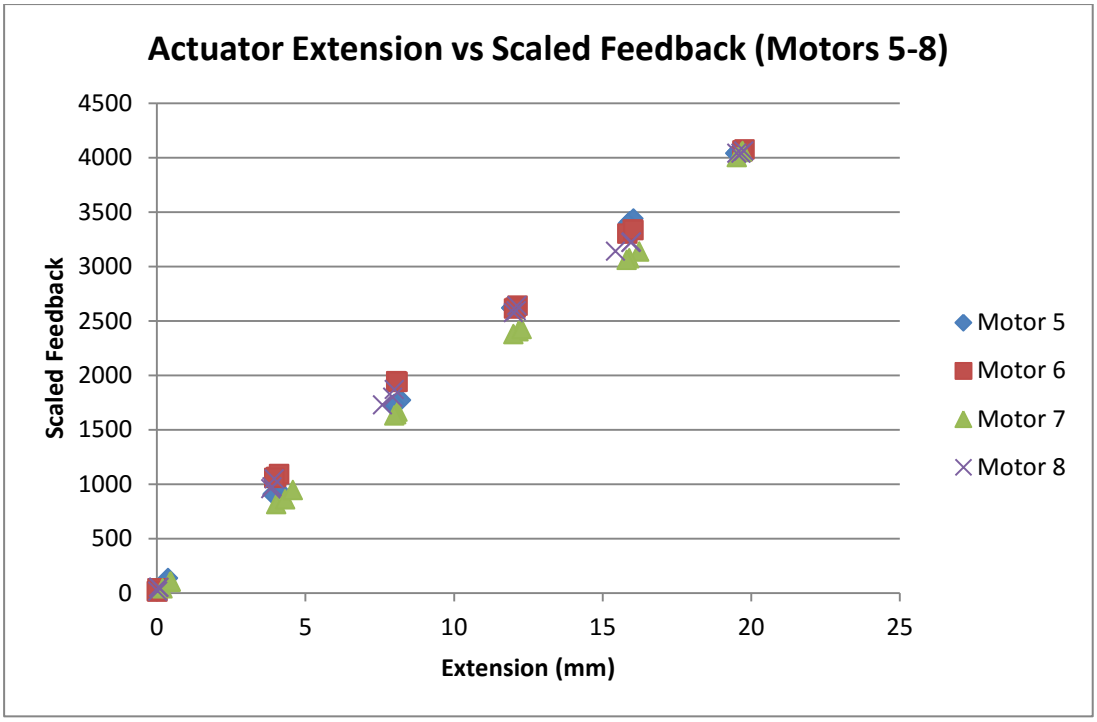


Figure 7.5 Actuator Extension vs Scaled Feedback (Motors 5-8).

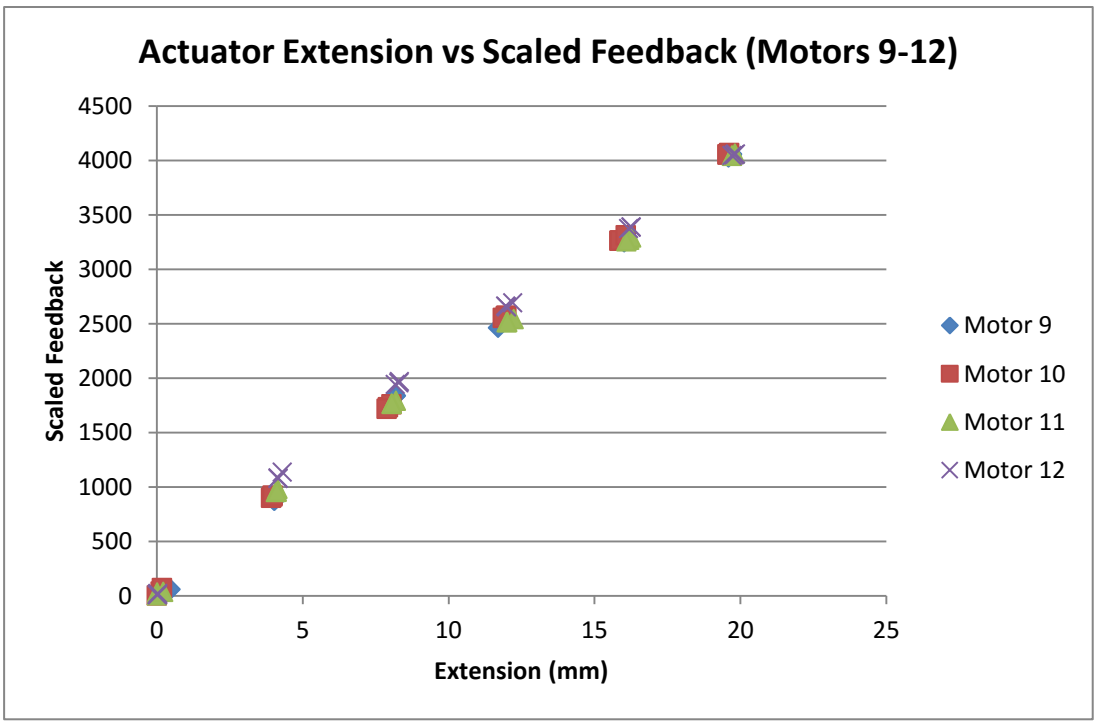


Figure 7.6 Actuator Extension vs Scaled Feedback (Motors 9-12).

Table 7.1 Calibration equations for the linear actuators and their corresponding percentage error values.

Motor	Extension, x, to feedback, f	Extension Deviations (mm)		
		Min	Max	Average
1	$f = 199.6x + 205.4$	0.1798	0.9087	0.5194
2	$f = 200.7x + 264.7$	0.2390	1.319	0.7388
3	$f = 200.1x + 46.03$	0.002869	0.4129	0.1837
4	$f = 205.9x + 38.91$	0.0007576	0.3324	0.1812
5	$f = 207.3x + 76.28$	0.001544	0.3641	0.1983
6	$f = 199.9x + 178.6$	0.1431	0.8484	0.4409
7	$f = 200.6x - 1.603$	0.007393	0.5989	0.2408
8	$f = 199.1x + 143.5$	0.04052	0.6904	0.3735
9	$f = 204.3x + 41.89$	0.02809	0.5990	0.2562
10	$f = 204.2x + 74.79$	0.06338	0.3467	0.1996
11	$f = 200.7x + 82.96$	0.005810	0.4639	0.2250
12	$f = 200.5x + 171.5$	0.2074	0.8505	0.4878
		Overall average:		0.3371

The extension deviations were plotted against the actual extensions, to assess how the accuracy of the calibration changed with extension. This is shown in Figure 7.7.

In summary, the calibration resulted in an average positional accuracy of +/- 0.3371mm and a guaranteed positional accuracy across all actuators of +/- 1.319mm (taking the worst case scenario with actuator 2). It was clear from Figure 7.7 that there was no correlation between the accuracy of the calibration and the extension of the actuator.

With the actuators and controllers calibrated, the calibration equations could be programmed into the PC control software and used to control the actuator extensions.

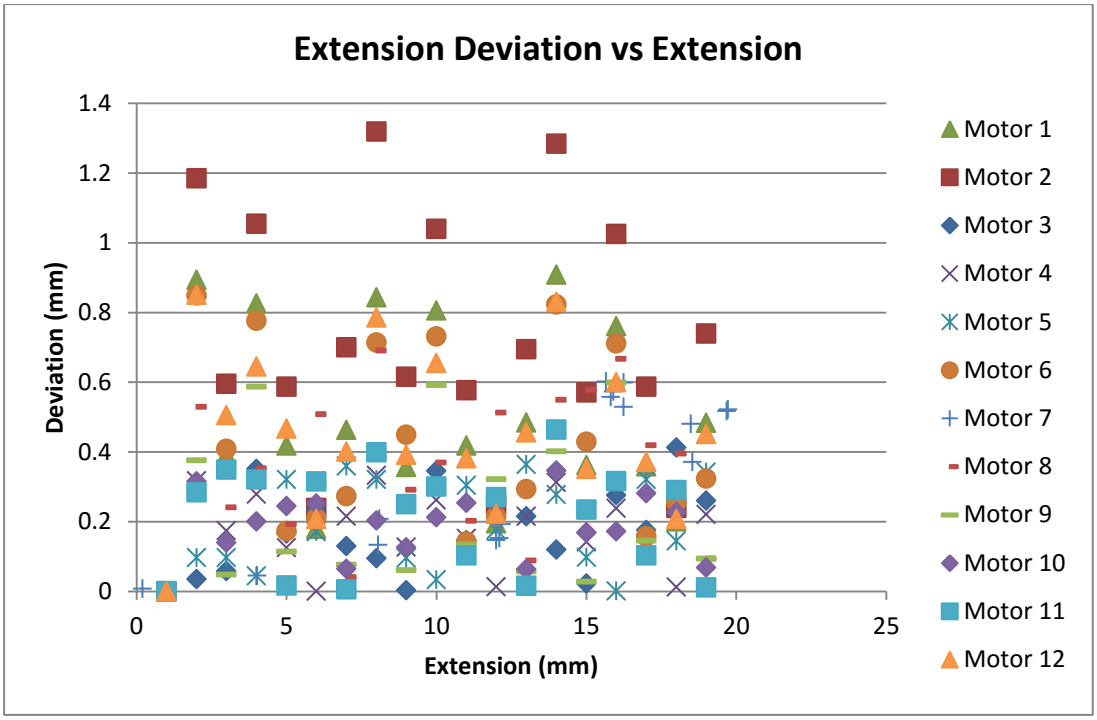


Figure 7.7 Linear actuator calibration errors vs extension.

7.2.2 Motion Capture Validation

Motion capture is a technique used in a wide variety of different industries, from medical to film, in order to capture physical movement (usually from human beings) into digital information that can then be processed. Typically, motion capture systems are composed of a series of markers that are attached to the subject at various strategic locations, along with a set of sensors (for example, cameras) that surround the working area of the subject and detect the location of these markers. The markers can be either active or passive. Passive markers rely on an external source to emit energy, which they then reflect back to the sensors. Active markers directly emit energy which is detected by the sensors. As the sensors are positioned in different locations, each marker will be detected by each sensor differently. By combining the information from each sensor, the location of the markers in three dimensions can be calculated.

For this experiment, the Optotrak Certus system was used. This system uses a set of infrared cameras surrounding the test area and uses active markers that are fitted with infrared LEDs. The components of the system are shown in Figure 7.8.

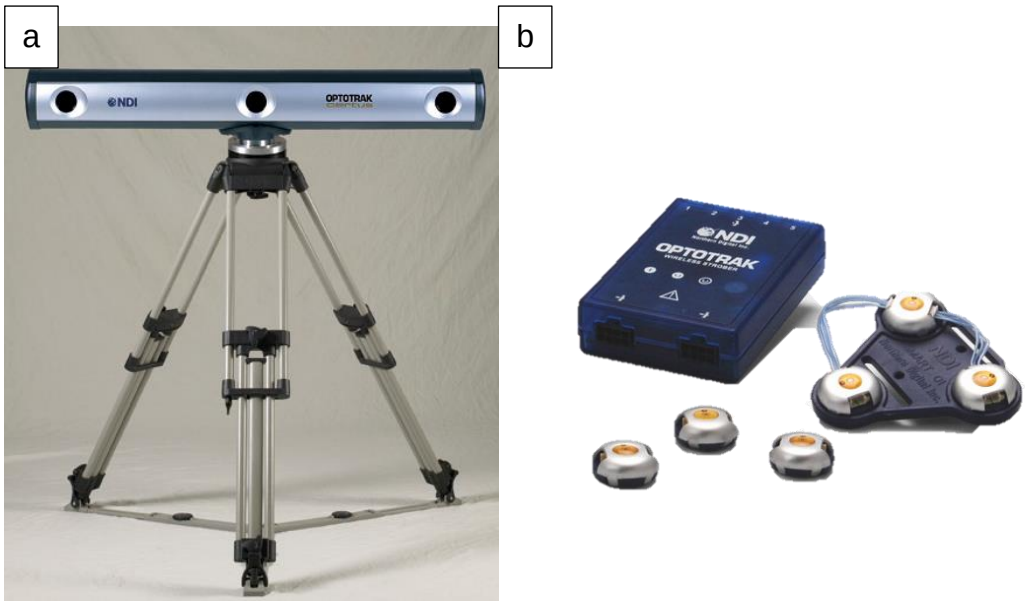


Figure 7.8 Components of the Optotrak Certus motion capture system.
a) The imaging cameras [127]; b) the active markers [128].

7.2.2.1 Marker Positioning and Theoretical Preparation

As mentioned previously, the purpose of the motion capture validation was to compare the theoretical robot position calculated via the IK equations to the actual position as a result of the motor controllers extending the actuators according to the extensions calculated by the equations. The motion capture markers would need to be placed at various locations on the robot and the control software modified to calculate the theoretical locations of those markers for each position that the robot takes. Then, as the control software sends position commands to the robot, the theoretical positions of the markers could be compared to the actual positions as measured by the motion capture system.

Since the top and bottom rings are used to define the robot's local origin and target location and orientation, as explained in Chapter 6, it was decided to place the markers at the corners of the top and bottom rings. This would allow the robot's local origin to be calculated using the bottom three markers, and would allow the target location and orientation to be calculated from the top three markers. A set of adapters were designed and 3D printed that allowed the markers to be securely attached to the robot. A photo of the markers mounted on the robot, as well as images of the mounting adapters are shown in Figure 7.9.

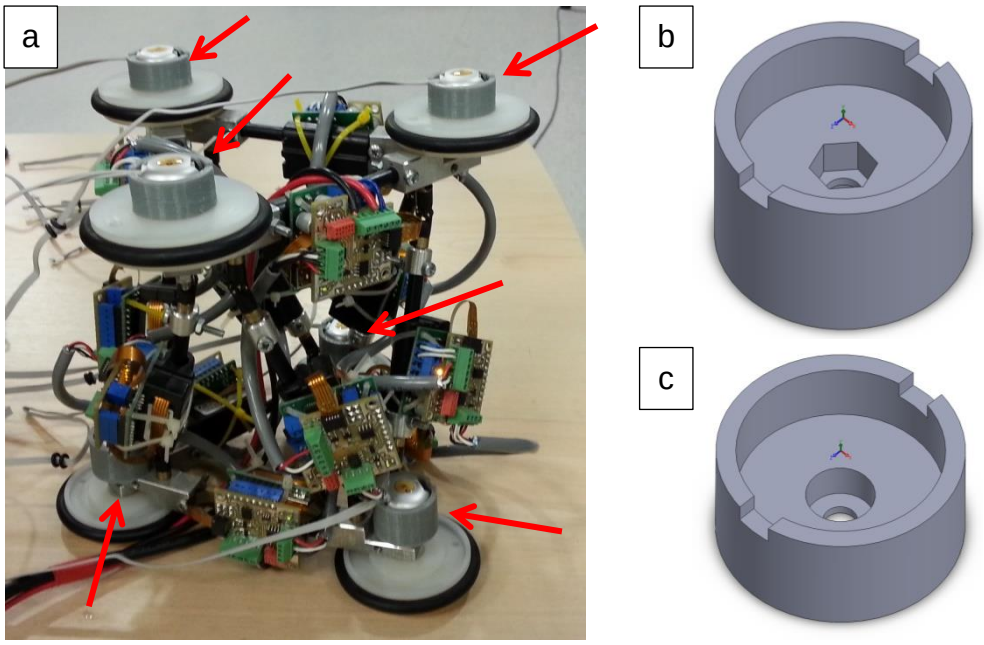


Figure 7.9 Motion capture markers mounted on the robot.

a) Location of the motion capture markers on the robot; b) bottom mounting adapter; c) top mounting adapter.

Once the marker positions were decided, the formulas used to calculate the theoretical positions of those markers needed to be derived and integrated into the control software. In order to do that, the marker positions needed to be related back to the geometric representation of the rings that was presented back in Chapter 6. Figure 7.10 shows the geometric representation of the ring in relation to the marker positions.

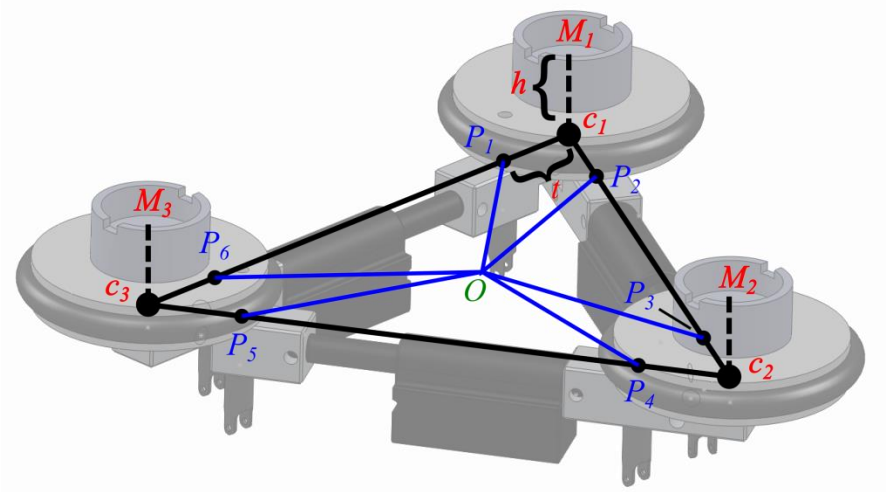


Figure 7.10 Motion capture marker positions in relation to the geometric representation of the ring.

The markers, $M_{1..3}$, are located at the corners of the ring triangle, but at a height offset, h, due to the height of the marker adapters and the height of

the markers themselves. The same geometric layout is also applicable to the bottom ring, with the only difference being that the height offset is different.

In order to calculate the corner positions on the ring, c_x, for any of the markers, the point P_y is moved along the vector $\overrightarrow{P_yP_z}$ by a distance of t. The P_x vectors were already calculated as part of the IK software, and the distance t has already been defined in Table 6.1. Therefore in order to determine $c_{1..3}$, the following equations are used:

$$c_1 = P_1 + t\left|\overrightarrow{P_6P_1}\right| \tag{81}$$

$$c_2 = P_3 + t\left|\overrightarrow{P_2P_3}\right| \tag{82}$$

$$c_3 = P_5 + t\left|\overrightarrow{P_4P_5}\right| \tag{83}$$

This determines the corner positions but without taking into account the height offset. In order to compensate for the offset, the actual marker positions need to be translated downward in the direction of the normal vector of the ring plane. The normal vector can be calculated using the cross product of the two vectors formed from the corner positions as follows:

$$\hat{n} = \widehat{\overrightarrow{c_1c_2}} \times \widehat{\overrightarrow{c_1c_3}} \tag{84}$$

The actual marker positions can then be translated to correspond to their positions on the ring, c_x, as follows:

$$c_x = M_x + h \cdot \hat{n} \tag{85}$$

The IK software can calculate the theoretical corner positions via equations (81)-(83), and the actual marker positions retrieved from the motion capture system can be translated to correspond to the theoretical marker positions using equations (84)-(85).

7.2.2.2 Experimental Procedure

With the markers mounted on the robot and the software modified to generate their theoretical positions, the actual experiment could now be carried out. In order to perform the validation, the following procedure was carried out.

1. The motion capture system was set up and recording of the marker positions was started.
2. A set of random top and bottom ring actuator extensions, target positions and orientations were generated and then passed to the IK equations.
3. These would then calculate the extensions that the Gough-Stewart actuators would need to extend to in order to achieve that target. If

the target resulted in invalid GS actuator extensions (i.e. outside the range of 0-20mm), this was discarded, otherwise, the theoretical corner positions were recorded and commands were sent to the robot to extend all actuators to the required lengths.

4. The robot would then hold that position for 10 seconds, allowing the motion capture system to record the marker positions for that long.
5. Steps 2-4 were then repeated.

The experiment was carried out and a total of 160 random positions were captured, spanning four sessions. The experiment resulted in two sets of data: the theoretical corner positions for each randomly generated target, and the recorded data stream from the Optotrak Certus system. In order to compare the two, the continuous data stream from the motion capture system needed to be split down into the data for each of the individual target positions, then the coordinates for each marker averaged to produce a single set of marker positions for that target.

7.2.2.3 Data Pre-processing

The data generated by the motion capture system is a list of coordinates for each marker tabulated against the captured frame number. In order to separate the data into the individual segments that represent a target, the data was analysed to determine the points at which the robot was transitioning from one target to the next. This was done by calculating the sum of the absolute difference between the coordinates in each frame and the last one. This value was then plotted in a graph, as shown in Figure 7.11.

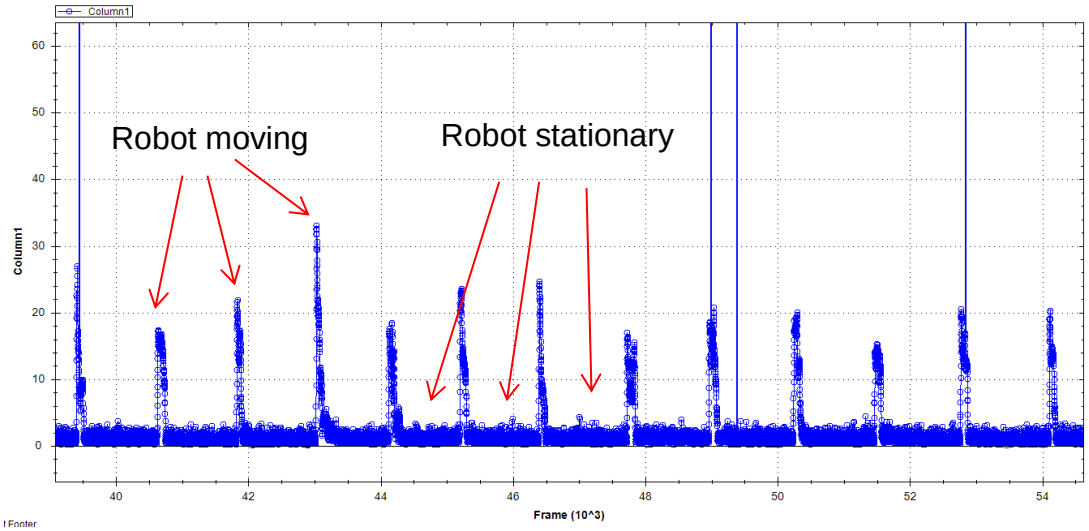


Figure 7.11 Annotated graph showing the difference between each motion capture frame.

As can be seen from the graph, the difference value peaks at regular intervals, with the gap between each interval corresponding to the 10s time period that the robot was left stationary. It was therefore possible to separate the motion capture data into the segments that represented each target by splitting the data along the spikes in the frame difference. Software was written in Python to perform this task automatically. Any invalid segments (i.e. ones where the motion capture system was unable to get a good view of all the markers due to the robot's position) were discarded along with their corresponding theoretical positions, leaving 155 points that could be compared.

7.2.2.4 Coordinate System Translation

At this point, however, the data was not ready to be compared as the coordinate systems for the theoretical data and the motion capture data were completely different. The theoretical data used the robot's axis and coordinate system described in Chapter 6, whereas the motion capture system had its own local coordinate system. Therefore, the coordinates generated by the motion capture system needed to be translated into the robot's coordinate system. This involved the following main steps:

- Determining the robot's axis system and origin within the motion capture coordinate system.
- Applying the height offset, h, described earlier.
- Converting the marker coordinates into the robot's axis system.

Determining the robot's axis system using the motion capture data involved examining the robot's geometry and using that to calculate the three axis vectors using only the marker positions and known geometric constants. Since the robot's origin is linked with the bottom ring, the bottom marker positions are used to determine it. Figure 7.12 shows the relationship between the x- and y-axes and the robot ring geometry.

7.2.2.4.1 X-Axis

The x-axis is calculated simply by determining the unit vector of the vector $\overrightarrow{OP_1}$. Therefore, both the points O and P_1 need to be calculated from the three marker positions, $c_{1..3}$.

The triangle origin, O, is calculated by simply working out the average of the three marker coordinates (note that at this point, the origin O is not the actual origin as the markers used to calculate it are still offset by the height h. This will be adjusted for later on). The point P_1 is calculated by moving along the

vector $\overrightarrow{c_1c_3}$ by a distance of t. The x-axis direction can then be calculated and then normalised to get the unit vector $\hat{x}$.

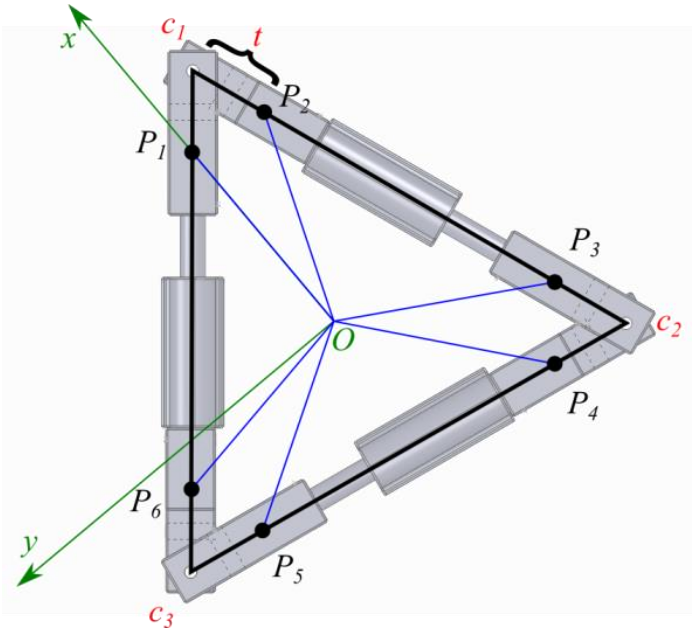


Figure 7.12 The robot's x- and y-axes in relation to the ring geometry.

$$O = \frac{c_1 + c_2 + c_3}{3} \tag{86}$$

$$P_1 = t \cdot \frac{\overrightarrow{c_1c_3}}{\|\overrightarrow{c_1c_3}\|} \tag{87}$$

$$\hat{x} = \frac{\overrightarrow{OP_1}}{\|\overrightarrow{OP_1}\|} \tag{88}$$

7.2.2.4.2 Y-Axis

The y-axis can be calculated by taking advantage of the fact that any vector can be represented by two orthogonal vectors. Therefore, if a vector is selected that is known to have a positive y-component, then that component can be determined by subtracting the x-component from the original vector and then normalising to get the unit vector.

The vector chosen for this was the vector $\overrightarrow{Oc_3}$, as it clearly has a positive y-component. The first step is to determine the component of this vector that lies along the x-axis. This is done by working out the projection of that vector along the x-axis using the dot product, as shown below.

$$\overrightarrow{Oc_{3_{\hat{x}}}} = \overrightarrow{Oc_3} \cdot \hat{x} \tag{89}$$

The y-component could then be calculated as follows:

$$\overrightarrow{Oc_3} = \overrightarrow{Oc_{3_{\hat{x}}}} + \overrightarrow{Oc_{3_{\hat{y}}}} \tag{90}$$

$$\therefore \overrightarrow{Oc_{3_{\hat{y}}}} = \overrightarrow{Oc_3} - \overrightarrow{Oc_{3_{\hat{x}}}} \tag{91}$$

$$\hat{y} = \frac{\overrightarrow{Oc_{3_{\hat{y}}}}}{\left\|\overrightarrow{Oc_{3_{\hat{y}}}}\right\|} \tag{92}$$

7.2.2.4.3 Z-Axis

The final component to be calculated is the z-axis. This is simply done by working out the cross product of both the x- and y-axes, to produce a vector that is orthogonal to both. The order of the cross product is chosen to produce a vector that points “up”.

$$\hat{z} = \hat{x} \times \hat{y} \tag{93}$$

7.2.2.4.4 Height Offset Application

As mentioned previously, the origin O will need to be compensated by shifting it “down” along the direction of the normal vector, $\hat{n}$, of the plane formed from the three markers. The method for doing so is shown in equations (84) and (85).

7.2.2.4.5 Conversion to the Robot Coordinate System

With the robot’s orthogonal axes and origin now determined within the motion capture coordinate system, it now becomes possible to translate the marker coordinates over to the robot coordinate system. First, all 6 marker positions are shifted “down” by their corresponding height offsets along their corresponding plane normals. Then, the x, y and z components are determined by calculating the projection of the line between the robot origin and the marker along each of the axes. This is shown below.

$$c_a = \begin{bmatrix} c_{a_x} \\ c_{a_y} \\ c_{a_z} \end{bmatrix} = \begin{bmatrix} \overrightarrow{Oc_a} \cdot \hat{x} \\ \overrightarrow{Oc_a} \cdot \hat{y} \\ \overrightarrow{Oc_a} \cdot \hat{z} \end{bmatrix} \tag{94}$$

7.2.2.5 Analysis

At this point, all the data recorded by the motion capture system had been converted to use the robot’s coordinate system and was thus directly comparable to the theoretical marker positions. The data was tabulated and the two sets compared by calculating the absolute difference between each coordinate’s x, y and z components in both the theoretical and measured data sets. The absolute distance between the theoretical and actual

coordinates for each marker was then calculated. The data was examined and various statistics derived in order to assess how close the actual positions were to the theoretical ones. Figure 7.13, Figure 7.14 and Table 7.2 summarise the data.

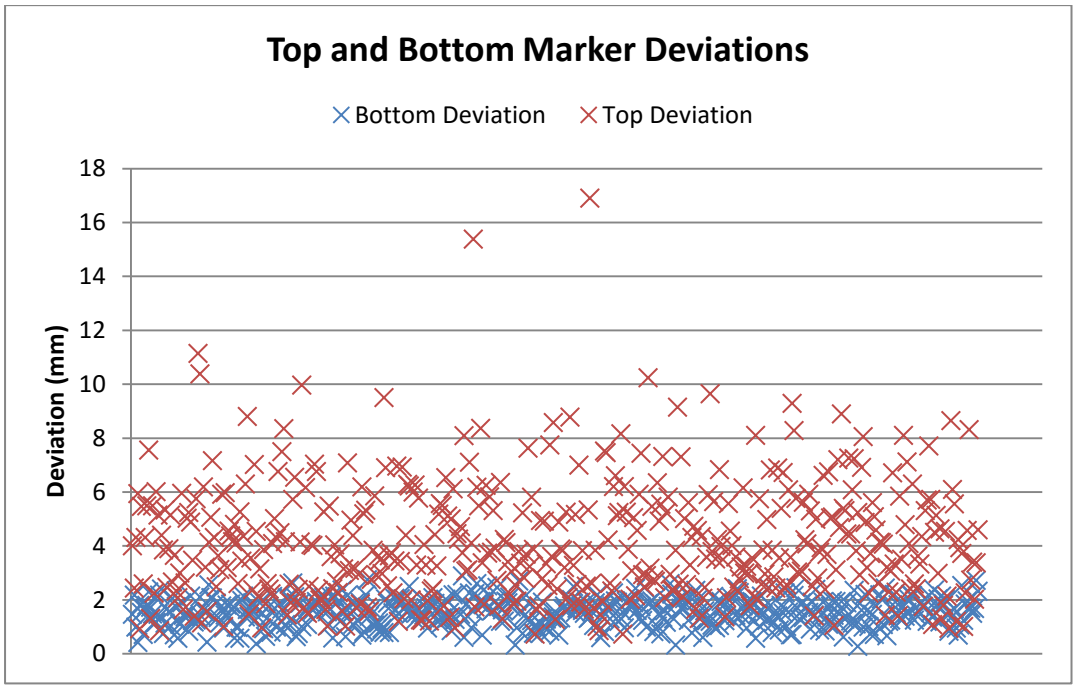


Figure 7.13 Average deviation of the bottom and top markers.

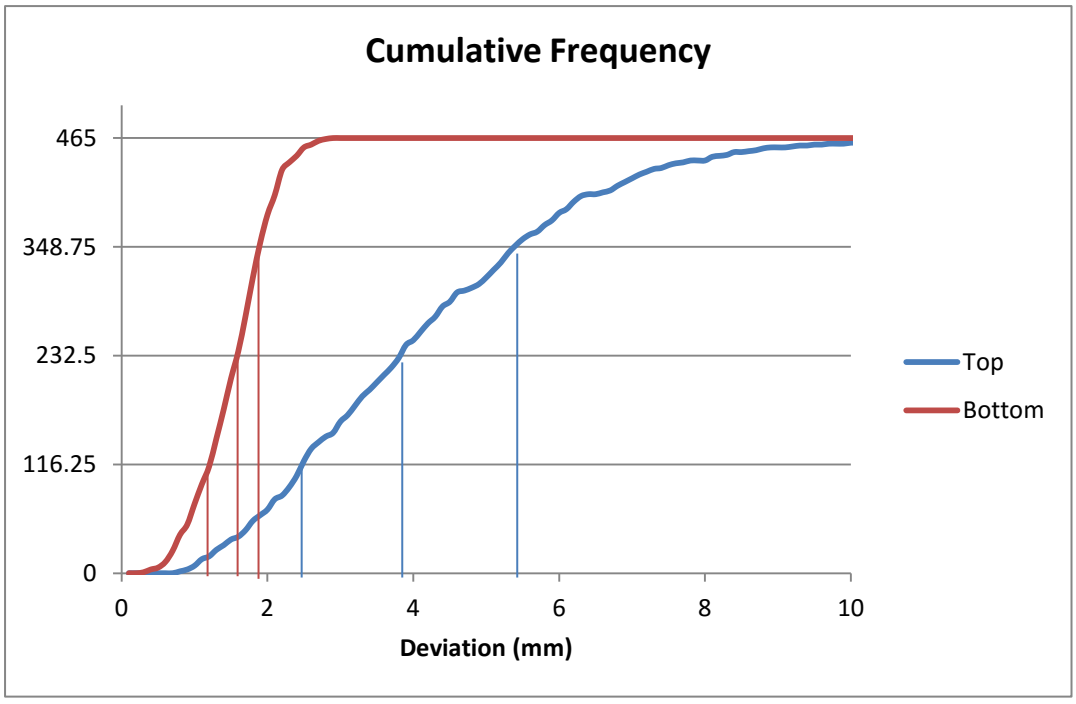


Figure 7.14 Cumulative frequency graph for the top and bottom ring marker deviations, with the median and the upper and lower quartiles annotated.

Table 7.2 Summary of statistics measured on the motion capture data comparison.

Property	Value (mm)
Average deviation for all markers:	2.835
Bottom Markers	
Average deviation for bottom markers:	1.545
Max bottom deviation:	2.885
Min bottom deviation:	0.2674
Lowest 99% deviations <	2.607
Lowest 95% deviations <	2.350
Lowest 90% deviations <	2.165
Lowest 80% deviations <	1.949
Average of the lowest 99% deviations	1.532
Average of the lowest 95% deviations	1.492
Average of the lowest 90% deviations	1.452
Average of the lowest 80% deviations	1.378
Top Markers	
Average deviation for top markers:	4.124
Max top deviation:	16.90
Min top deviation:	0.7228
Lowest 99% deviations <	10.06
Lowest 95% deviations <	7.984
Lowest 90% deviations <	6.903
Lowest 80% deviations <	5.779
Average of the lowest 99% deviations	4.030
Average of the lowest 95% deviations	3.830
Average of the lowest 90% deviations	3.641
Average of the lowest 80% deviations	3.317

In summary, it can be seen from the data that, on average, the markers were within approximately 2.8mm of their theoretical positions, with the bottom markers matching their theoretical positions more closely than the top ones. The spread of deviations was also much smaller for the bottom markers than for the top markers.

The reason for the difference in accuracy between the bottom and top markers is very likely due to the fact that because the bottom markers form the robot's base, they are only affected by the three linear actuators that form the bottom ring and their corresponding joints. The positions of the top markers, on the other hand, are affected by all 12 linear actuators that form the robot and all the joints that connect the robot's structure together. Therefore the bottom markers are only affected by the accumulated errors of

only three actuators and their corresponding mechanical joints, whereas the top markers are affected by all the accumulated errors in the system, which would lead to higher inaccuracy at the top.

The errors in the system can be split into two types: systematic errors and random errors. The systematic errors would be responsible for the average deviation being non-zero. Such errors could be caused by the following:

- Positional accuracy errors due to the linear actuator feedback calibration equations.
- Inaccuracy in the manufactured dimensions of the parts that compose the robot.

Random errors would be responsible for the variance in the deviation for each frame around the average value. These could be caused by the following:

- Lack of rigidity in the joints connecting the mechanical structure of the robot, which could lead to inadvertent movements.
- Hysteresis in the linear actuator feedback potentiometer, resulting in different feedback values reported for the same physical extension.
- Motor controller PID deadband, where the controller deems an actuator's extension to be correct if it is within a certain range of the target extension, which could lead to actuators not extended precisely to the target location.

Using the calibration accuracy values that were derived in section 7.2.1.3 (average positional accuracy +/-0.3371mm, max +/-1.319mm), it is possible to derive some rough estimates for the accumulated errors that would affect each set of markers. These are summarised in Table 7.3 and are compared with the error values derived from the motion capture data in Table 7.2.

When the values in the Table 7.3 are compared to the average and maximum errors derived from the motion capture data, it becomes clear that there is a correlation between the two. It is therefore reasonable to state that it is very likely that most of the positional errors are due to the accuracy of the linear actuator feedback calibration, with the remaining errors likely caused by a combination of the other factors mentioned earlier. Improving the linear actuator feedback calibration (e.g. by increasing the number of calibration points, or using a non-linear calibration curve) would very likely have the most significant effect on the positional accuracy of the system.

Table 7.3 Estimated accumulated errors due to calibration inaccuracies compared with errors derived from the motion capture data.

	Number of linked actuators	Estimated average accumulated error **(average motion capture error)**	Estimated maximum accumulated error **(maximum motion capture error**
Bottom markers	3	3x0.3371 = 1.011mm **(1.545mm)**	3x1.319 = 3.957mm **(2.885mm)**
Top markers	12	12x0.3371 = 4.0452mm **(4.124mm)**	12x1.319 = 15.828mm **(16.90mm)**

Despite the inaccuracies in the positioning, it was clear from the data that the robot did closely match the theoretical positions, and along with the simulation validation done on the IK equations, this proves that the equations derived and their software implementation are a valid and correct representation of the robot.

7.3 In-Pipe Demonstration

With the equations that govern the robot's motion experimentally validated, work was begun on demonstrating the robot's ability to move inside pipes. The robot was first tested in a straight pipe and then in a segmented bend.

7.3.1 Straight Pipe Demonstration

The first in-pipe demonstration was done in a straight section, as this was expected to be simpler. As discussed previously, the robot was designed to operate in a 200mm diameter pipe, as this pipe fell within the range of diameters used for clean water, was easily available and was relatively large, making construction of the robot simple and without the need for expensive miniature components.

The movement algorithm for the robot in a straight pipe is very simple and follows a traditional inchworm-like motion strategy. The robot starts off with its bottom ring expanded, the top ring retracted and all the Gough-Stewart actuators retracted. The movement sequence then repeatedly follows the algorithm shown in Table 7.4.

Table 7.4 Summary of the movement algorithm in a straight pipe.

Sequence no.	Top ring	GS Actuators	Bottom ring
1	Retracted	Retracted	Extended
2	Retracted	Extended	Extended
3	Extended	Extended	Extended
4	Extended	Extended	Retracted
5	Extended	Retracted	Retracted
6	Extended	Retracted	Extended

The robot was tested in a straight pipe using the above algorithm and filmed. Figure 7.15 shows the steps the robot took while climbing up the pipe. The test demonstrated that the robot was capable of climbing up a vertical pipe.

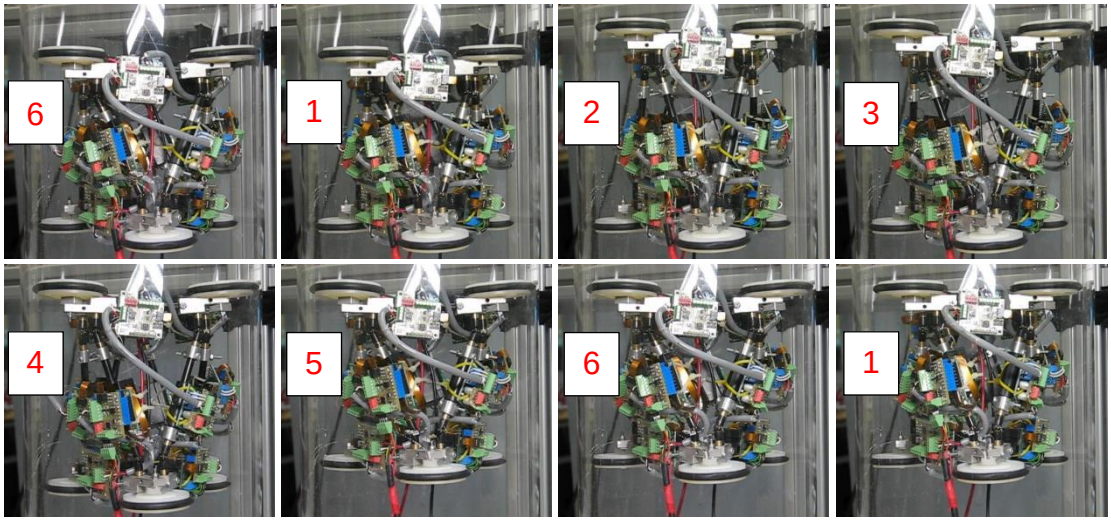


Figure 7.15 Robot climbing up a vertical pipe.

The numbers next to each frame correspond to the sequence number in Table 7.4.

From this experiment, various performance metrics were measured. These are summarised below:

Table 7.5 Straight climbing performance measurements.

Metric	Value
Overall climbing speed	2.389mm/s
Distance per step	21.5mm
Step duration	9.00s

The main performance criteria that can be extracted from the test is the robot's climbing speed. This speed is primarily determined by two properties of the robot: the speed at which the actuators extend and the robot's stroke length.

The actuator extension speed is one of the primary factors that determine the robot's climbing performance. The faster the actuators can extend and retract, the faster the robot can complete each of the motion steps, allowing it to climb faster. For this robot, the actuators were measured to take 1.3s to fully extend or fully retract while the robot was climbing up inside the vertical pipe. Using faster actuators would allow the robot to complete its motion steps in less time, and thus climb the pipe faster.

The second criterion that affects the robot's climbing speed is the stroke length i.e. the distance the robot travels during one cycle of its movement algorithm. Out of the six steps described in Table 7.4, only two of those steps are used to move the robot forward (the transition from step 1 to step 2, and from step 4 to step 5), whereas the other steps are involved with gripping the pipe wall. The other 4 steps can be considered to be "fixed overheads" in the motion algorithm that do not contribute at all to the robot's progression up the pipe. If the total time taken to execute the "fixed overhead" steps is represented by f and the total time taken for the actuators to travel a distance d, is represented by m, the following equation represents the total time taken by one cycle of the movement algorithm to move the robot by the distance d:

$$time_d = f + m \tag{95}$$

In order to move by double the distance, $2d$, the robot would need to execute two movement cycles, and therefore the total time would be:

$$time_{2d} = (f + m) + (f + m) = 2f + 2m \tag{96}$$

If the robot's actuators were changed so that they are capable of allowing the robot to move a distance of $2d$ in each cycle, but at the same extension rate as before, the total time taken to travel a distance of $2d$ would now be:

$$time_{2d} = f + 2m \tag{97}$$

As can be seen from the equation, the time taken to travel the actual distance is the same, but because only one algorithm cycle is needed, the amount of time contributed by the fixed overheads is reduced. It is therefore apparent that increasing the distance that can be travelled during each

algorithm cycle improves the robot's climbing speed. This would be done by using actuators that have higher stroke lengths.

7.3.2 Bent Pipe Demonstration

After proving that the robot was capable of climbing up a straight pipe, the next step was to demonstrate the robot's ability to navigate bends. The first step was to construct the pipe bend that the robot would navigate.

7.3.2.1 Pipe Bend Construction

In section 6.4, the robot's limitations when it comes to navigating bends were determined. With these limits estimated, a pipe bend could now be purchased or constructed in order to test the robot's ability to navigate bends. It was decided to use the same pipe diameter as the existing straight pipe due to the robot's ability to navigate it successfully. Due to the cost of purchasing transparent pipe bends with a 200mm diameter, it was decided to construct a segmented pipe bend. This would be composed of sections of straight pipe cut and glued together in order to form the bend. An example of such a pipe is shown in Figure 7.16.

The main difference between using a swept bend and a segmented bend is that a swept bend is formed from a continuous curved section, whereas a segmented bend is formed from multiple straight sections joined together. It is expected that navigating a segmented bend would be more difficult than navigating a swept bend as the transition between segments would be sharper than for a swept bend. This can be seen in Figure 7.17.

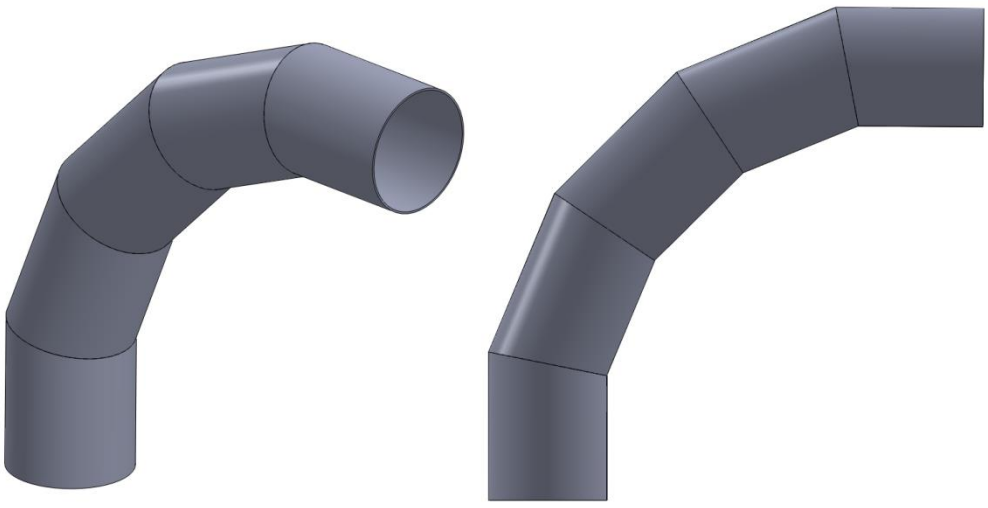


Figure 7.16 A segmented bend, showing how it is constructed.

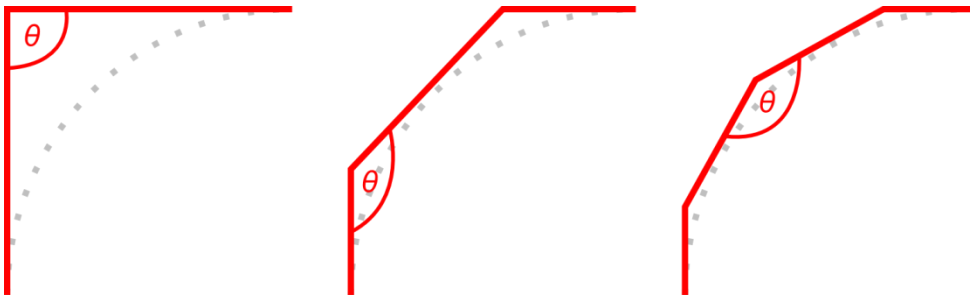


Figure 7.17 Figure showing how the number of segments in a segmented bend affects the angle between segments.

With a small number of segments, the angle between the segments is small, making it more difficult to traverse. As the number of segments increases, the angle between them also increases, making it easier to traverse. A swept bend is essentially a segmented bend formed from an infinite number of segments. Due to this, the use of a segmented bend for testing was deemed to be adequately representative due to the expectation that navigating it would be more difficult. Therefore, if the robot was capable of navigating a segmented bend, it would be capable of navigating a swept bend with the same bend radius.

A segmented bend was designed in Solidworks that has a bend radius of 634mm. This bend radius was achieved by first choosing a target radius of 650mm, as this was the smallest radius where the range of angles that the robot could form stopped increasing, as per Figure 6.31. A 2D representation of the bend was then drawn in Solidworks in order to specify the major dimensions such as the pipe diameter and the angles between the segments. As the dimensions were being adjusted, the overall bend radius became 634mm. Once the dimensions were fully defined, the 3D parts were modelled and combined into the final assembly to verify their dimensions. Figure 7.18 shows the 2D drawing used to define the pipe geometry and Figure 7.16 above shows the 3D model of the bend.

Once the bend segments were designed, transparent straight pipe section was purchased and the individual segments cut and glued together to form the final pipe, which is shown in Figure 7.19.

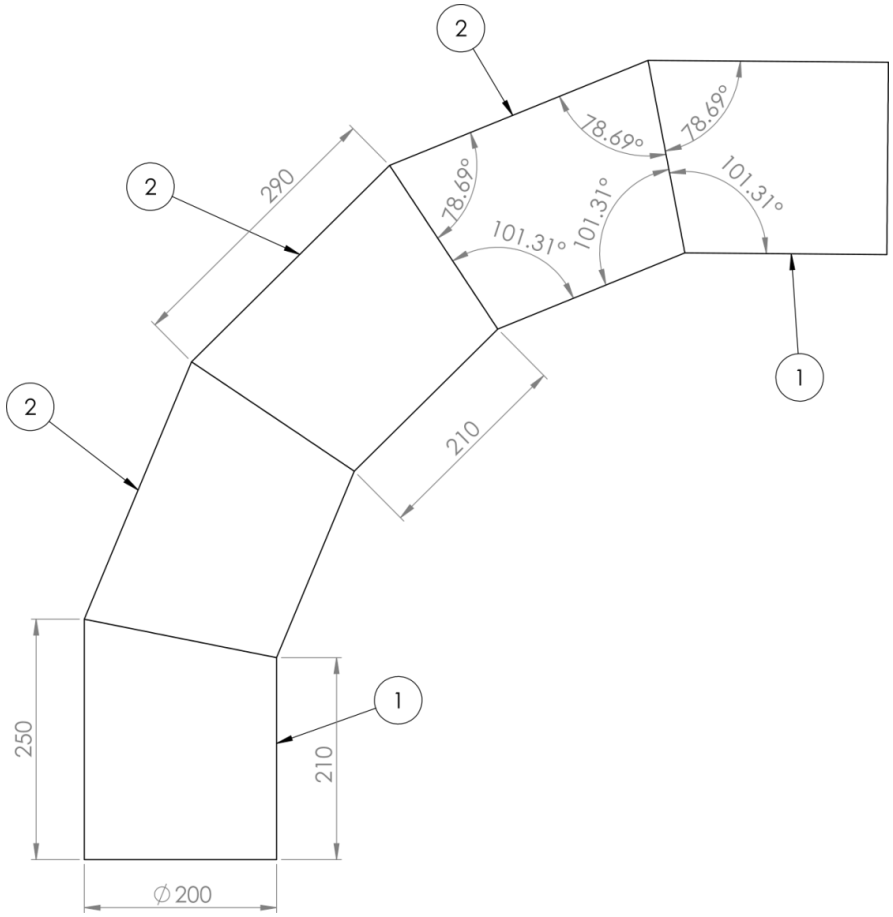


Figure 7.18 2D drawing of the segmented bend.

Parts labelled ① are the end segments, parts labelled ② are the mid segments.

Figure 7.19 Constructed segmented bend.

7.3.2.2 Initial Motion Planning

Once the pipe was constructed, work began on developing the motion algorithm that the robot would use to navigate the pipe bend.

The first algorithm worked on the assumption that the pipe would be a continuous bend and so the robot would alternate between two plane angles as a means of climbing up the bend. This is shown in Figure 7.20.

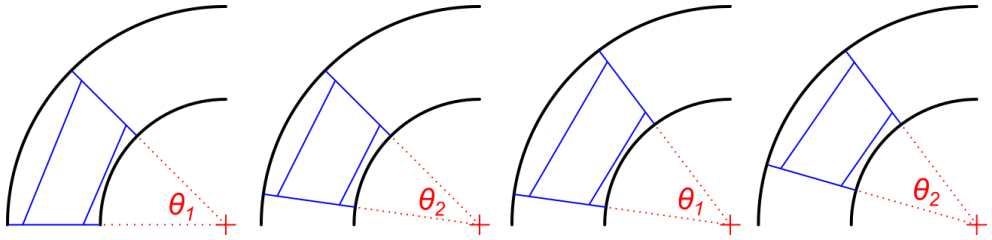


Figure 7.20 Movement algorithm for navigating a bent pipe by alternating between two plane angles.

In order to implement this motion algorithm, two plane angles need to be determined. This can be done simply by using the graph in Figure 6.31 to determine the maximum and minimum angles that are possible for the pipe's bend radius. For a bend radius of 634mm, the maximum and minimum angles are 10.75° and 11.6°.

As well as the angles between the bottom and top planes, the x, y and z offsets of the planes relative to each other also need to be determined. This can be done by examining the geometry of the robot inside the pipe, as per Figure 7.21.

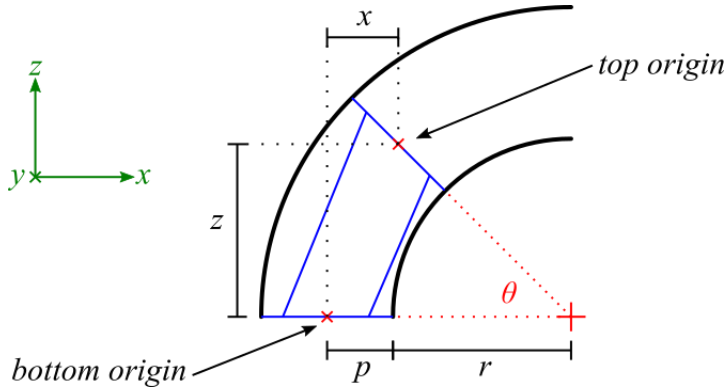


Figure 7.21 Relative positions and orientations of the top and bottom planes in a bent pipe.

In the above figure, the rotation of the top plane is assumed to be along the robot's y-axis, with the y-coordinate of the top origin relative the bottom one

being 0. Therefore, only the x- and z-coordinates need to be determined. Assuming that r is the pipe bend radius and p is the radius of the pipe, the x- and z-coordinates can be calculated as follows:

$$z = (p + r)\sin(\theta) \quad (98)$$

$$x = (p + r) - (p + r)\cos(\theta) \quad (99)$$

With the pipe bend radius being 634mm and the pipe diameter being 200mm, the robot's coordinates for each of the two angles are summarised in Table 7.6.

Table 7.6 Robot's coordinates for each of the two plane angles.

Y-Axis Angle	X	Y	Z
10.75°	12.88	0	137.1
11.6°	14.99	0	147.6

By inputting these values into the IK equations, the actuator extensions required to position the robot can be calculated.

This implementation did not succeed in navigating the pipe bend, with the robot moving out of alignment with the pipe and getting stuck. This was likely due to the fact that the pipe is not a continuously swept bend, but a segmented one. This means that there are certain sections where the robot needs to move as if it were in a straight pipe, and then switch to a bent-pipe algorithm only at the junction between two segments. Using a bent-pipe algorithm throughout the entire pipe results in the robot trying to move in a curved fashion inside the straight sections, which results in the robot moving out of alignment with the pipe's longitudinal axis and eventually getting stuck.

7.3.2.3 Manual Control Method

Due to the nature of the segmented bend, and the need for different motion algorithms at different locations in the bend, it was decided to move the robot under manual control.

The main control strategy used was to ensure that the top and bottom rings were always perpendicular to the pipe section that they were in. This ensures that the gripping force exerted by the rings is maximised, as the force would be acting perpendicular to the pipe wall. This also minimises the cross section of the pipe as seen by the rings. Figure 7.22 shows how the

angle that the ring takes inside the pipe affects the effective cross section that it interacts with.

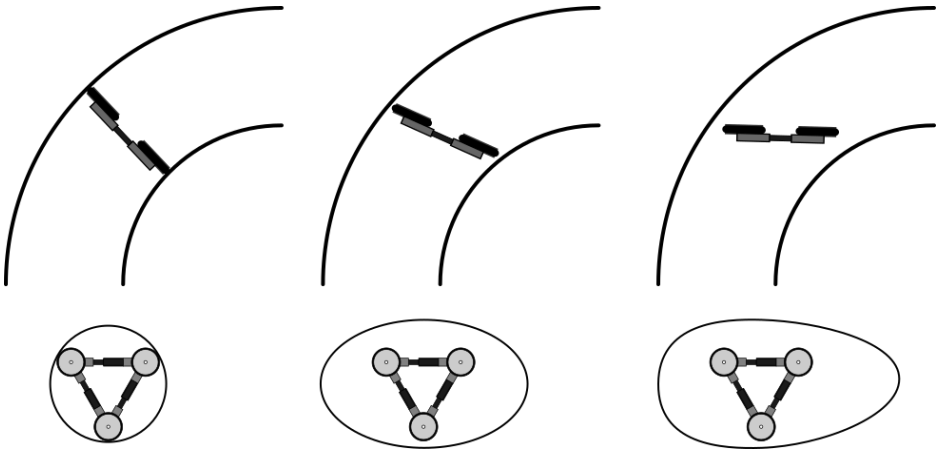


Figure 7.22 The angle that the robot takes inside the pipe affects the effective cross section.

As can be seen from the above, the effective cross section as seen by the robot is at its smallest when the robot ring is angled so that it is perpendicular to the pipe's longitudinal axis. As the angle changes away from 90°, the effective cross section widens and gets more elliptical. This makes it more difficult for the robot to effectively grip the pipe wall because a) the pipe has now become bigger and asymmetrical, so the ring actuators may not have enough extension to grip it securely and uniformly, and b) the ring is now contacting the pipe wall at a non-perpendicular angle, and so not all the actuator force is being used to grip the pipe wall, with a component of the force acting in such a way so as to cause the ring to slip. This can be seen in Figure 7.23.

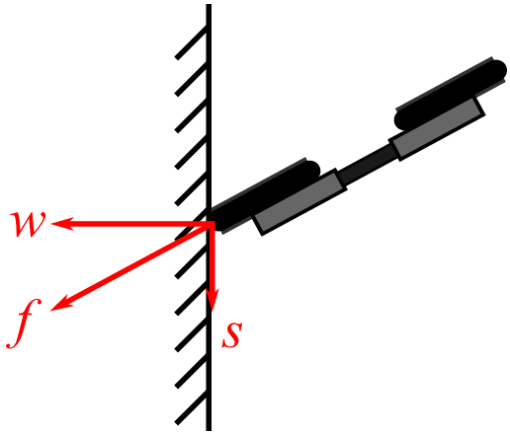


Figure 7.23 Components of the actuator force when acting at an angle.

When acting at an angle to the pipe wall, the actuator force, f, is split into two smaller components: a component, w, that acts to grip the pipe wall, and another component, s, that acts to move the ring away from the pipe.

Therefore, the ring needs to ideally act perpendicularly to the pipe wall in order to maximise the gripping force.

7.3.2.4 Bent Pipe Navigation Under Manual Control

Using the method described above, the robot was controlled manually through the segmented bend. Figure 7.24 shows various stages of the robot's progression in the bend.

Figure 7.24 Progression of the robot through the segmented bend under manual control.

As can be seen from the figure, the robot was successful in navigating the pipe bend and was able to completely navigate the joint between two of the pipe segments. It was not possible to extract any accurate performance data, such as the climbing rate, from this experiment as the robot was under manual control and each step that the robot took inside the pipe needed to be manually assessed and controlled. Due to limitations in the robot's movement ability, some of the movement steps required trial and error in order to determine a stable position for the robot to take. Also, there were several instances during the experiment where the robot slipped a short distance down the pipe, thus requiring that section to be climbed once again. Together, these reasons led to delays of various lengths between each

movement step, thus making it difficult to extract any accurate performance data.

7.3.2.5 Experimental Issues

Despite the robot's success in navigating the pipe bend, there were various issues faced by it that made the movement more difficult and would need to be remedied in subsequent designs if the robot is to be successful.

7.3.2.5.1 Problems

The problems faced by the robot are the following:

- The robot was unable to always keep the rings perpendicular to the pipe's longitudinal axis, due to limitations in its movement range. This led to the rings being placed in sub-optimal positions some of the time.
- The various loads applied on the top and bottom rings, combined with a lack of mechanical stiffness in the rings resulted in the actuator assemblies bending and the overall ring assembly sagging.
- The combination of the above two factors had a negative effect on the robot's gripping force, which led to the robot slipping down the pipe at several points during the experiment. Excessive forces on the actuators also resulted in two of them breaking during the course of the experiment.

The various reasons that contributed towards the above problems are explored below.

7.3.2.5.2 Actuator Extension and Length-Ratio

The limitations in the robot's movement ability are a direct result of the low extension capability and low length-ratio of the linear actuators used to construct the prototype. Using actuators with large extensions allows the robot to move the rings further out in space and to form sharper angles between the top and bottom rings, as shown in Figure 7.25. Therefore, using high extension actuators is one way of improving the robot's mobility.

However, the extension capability of the actuators is not in itself sufficient. The length-ratio of the overall actuator, or the ratio between the actuator's minimum and maximum lengths, is very important in determining how "space efficient" the robot is, and how well it can navigate in confined spaces. The length-ratio, l, can be calculated from the actuator's minimum length, x, and maximum extension, e, using equation (100).

$$l = \frac{e}{x} \tag{100}$$

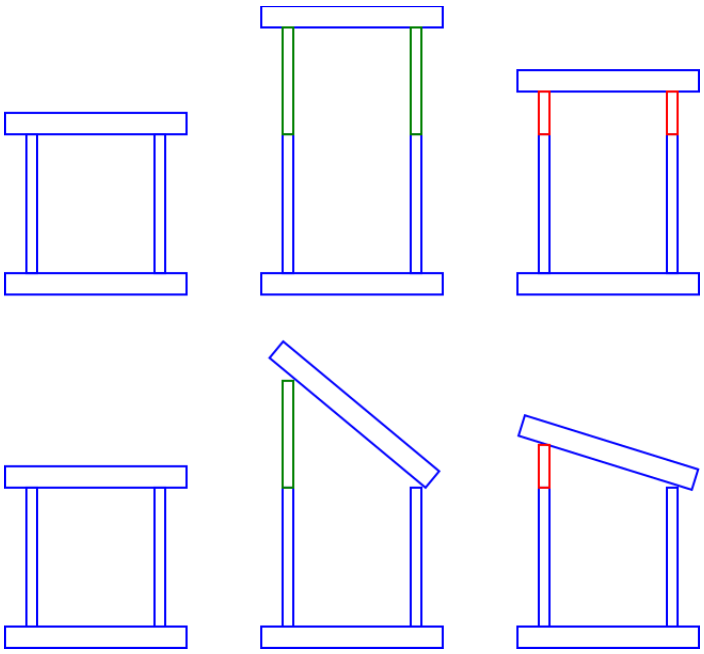


Figure 7.25 High extension actuators (green) allow the robot to extend out further and to form larger angles than low extension actuators (red).

The ideal linear actuator would have a minimum length of 0 and a very large, or infinite maximum extension, giving it a very large length-ratio. A robot built using such actuators would be able to collapse into a very small size, yet still be able to extend out by large distances. On the other hand, a poor linear actuator would have a large minimum size and only be able to extend by a small amount. A robot built using such actuators would suffer from a large "space inefficiency" penalty, whereby the robot is unable to collapse into a small size due to the large size of the linear actuators that comprise it, and is only able to extend out by a small fraction of its size. For a robot designed to operate in confined environments, such as water pipes, this would hinder the robot's ability to navigate through small and tight spaces, especially bends. Figure 7.26 shows how two robots built using actuators with the same extension range but different length-ratios would have different movement abilities.

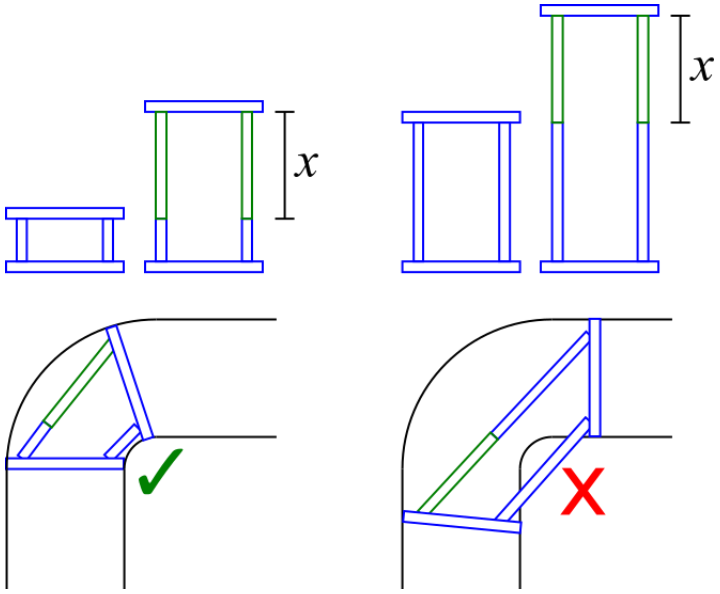


Figure 7.26 Diagram showing how two robots built using actuators that have the same extensions but different length-ratios can have different amounts of mobility.

The overall length-ratio of an actuator is governed by two factors: the length-ratio of the individual actuator itself, and the effect of any mechanical parts attached to the actuator.

The actuator itself will have a length-ratio that will be determined by its mechanical dimensions, and when choosing or designing an actuator, it will be necessary to select one with the largest length-ratio possible.

However, in order to integrate the actuator into the robot, various mechanical connecting parts need to be designed so that the actuator can be attached to the other parts of the robot. These parts have an effect on the overall length-ratio of the actuator as they increase its minimum size. The effect that these parts have can be seen in the assemblies that comprise the robot's ring- and GS-actuators.

As can be seen in Figure 7.27, the linear actuator by itself has a minimum size of 36.5mm, and an extension of 20mm, giving it a length-ratio of 0.548. However, when the actuator is integrated into the ring- and GS-assemblies the mechanical connecting parts increase the minimum size to 106.5mm and 83.7mm respectively, while the actuator extension remains the same. This leads to a reduction in the overall length-ratio of both actuators to 0.188 and 0.239, which is a reduction of 66% and 56% respectively. This is a significant reduction in the actuator's length-ratio. Therefore, in order to maximise the length-ratio of the actuators, it is important to minimise the size of any mechanical connecting parts used to connect the actuators to the rest of the robot.

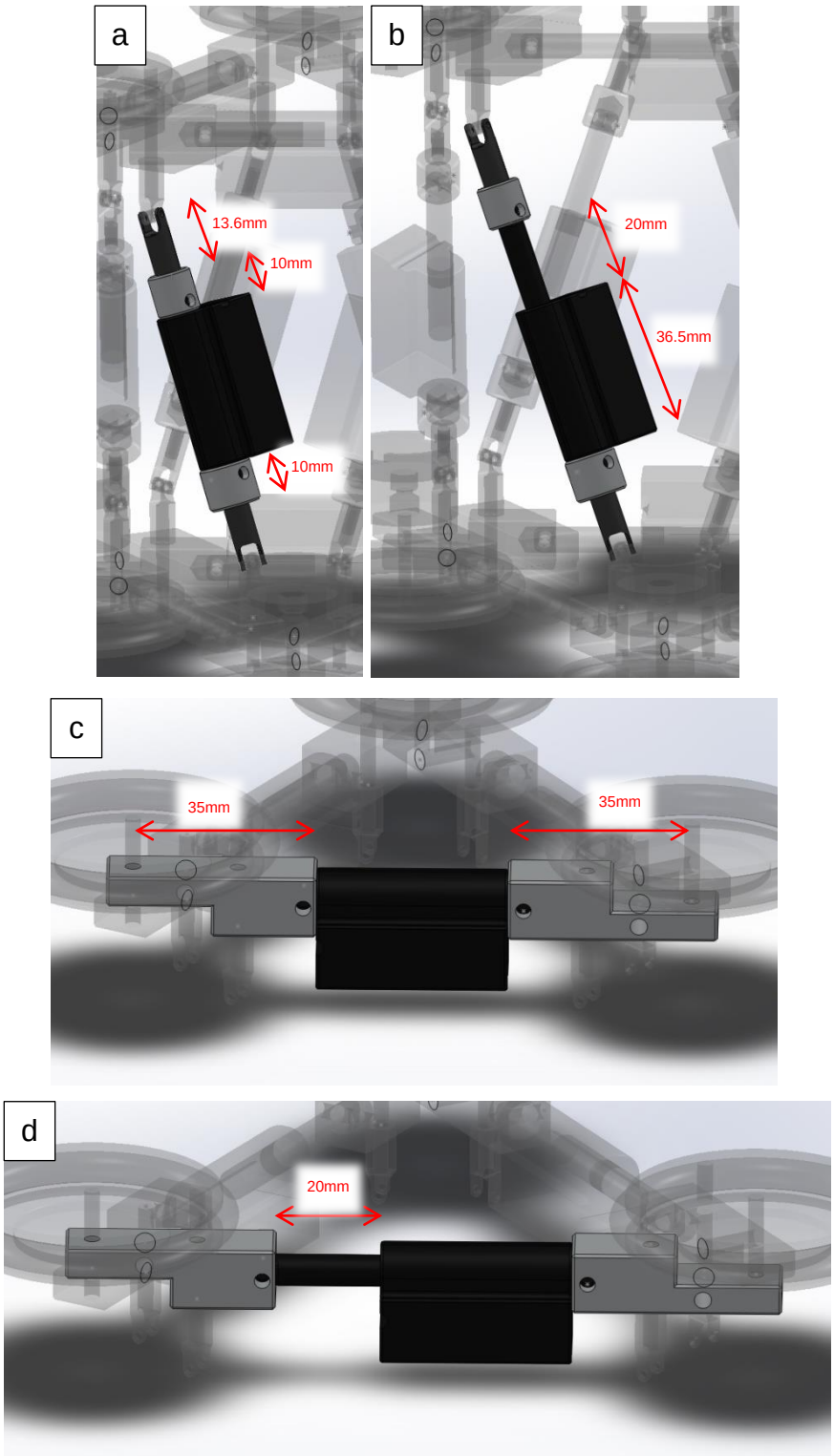


Figure 7.27 The minimum and maximum extents of the linear actuators when assembled in the robot.

Gough-Stewart actuator minimum (a) and maximum (b) extensions; ring actuator minimum (c) and maximum (d) extensions.

7.3.2.5.3 Ring Bending Moments

In section 5.3.3, the use of support discs instead of studs protruding from the rings was deemed to be a better mechanical solution as it would reduce the bending moments applied to the linear actuators. It was also acknowledged, however, that because the discs would be mounted slightly above the longitudinal axis of the linear actuators, there would still be some bending moments generated, albeit much reduced. The effects of these bending moments were observed during the experiment and these contributed towards the sagging of the rings and breakage of two actuators.

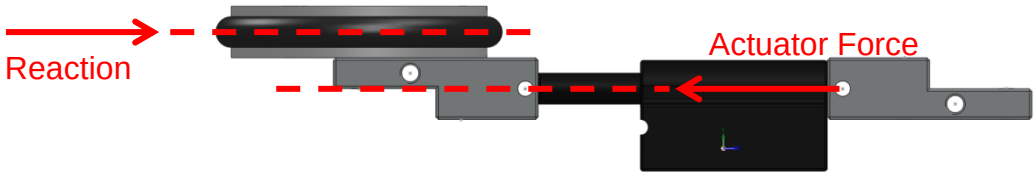


Figure 7.28 Side view of the robot ring showing the vertical offset of the support ring relative to the actuator longitudinal axis.

As can be seen from Figure 7.28, the robot's rings are designed in such a way that the support discs are mounted slightly above the longitudinal axis of the linear actuators. Because of this, the reaction force that is generated as a result of the linear actuators pushing the discs against the pipe wall acts at a vertical offset from the actuator force, which produces a bending moment in the ring. Due to a lack of stiffness and rigidity in the various joints that are found in the ring, this bending moment is not absorbed, and manifests as a bending of the entire ring actuator assembly, leading the whole ring to sag. This results in several issues:

The first is that the bending places a lot of mechanical stress on the joints connecting the various parts together because it effectively works to force the various parts apart. This was especially the case at the joint between the linear actuators and the aluminium connecting components, shown in Figure 7.29, most likely because these were the joints with the least stiffness. The joint is formed by inserting the linear actuator mounting point into the corresponding recess in the aluminium connecting component, then using a screw and nut to fix the two together.

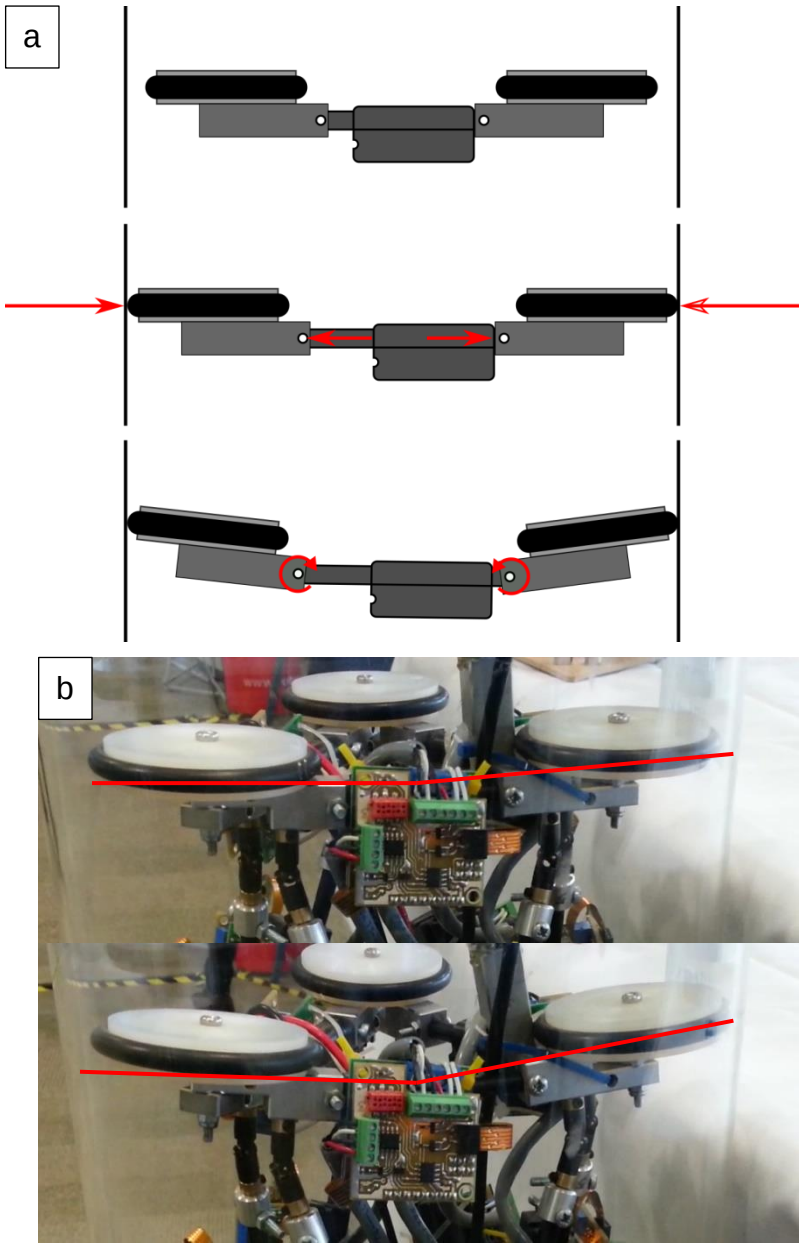


Figure 7.29 The effect of bending moments on the robot ring when bracing the pipe wall.

a) Due to the vertical offset between the actuator force and the reaction forces, a bending moment is generated at the joints between the actuators and the ring connecting parts, causing the ring to bend; b) An example of the bending occurring while the robot is climbing a pipe.

As the ring assembly expands inside the pipe and starts to bend, the linear actuator and aluminium connecting component are forced apart, despite the presence of the retaining screw. This places significant stress on the plastic mounting point of the linear actuator and resulted in two breakages during the experiment. An example of the breakage can be seen in Figure 7.30.

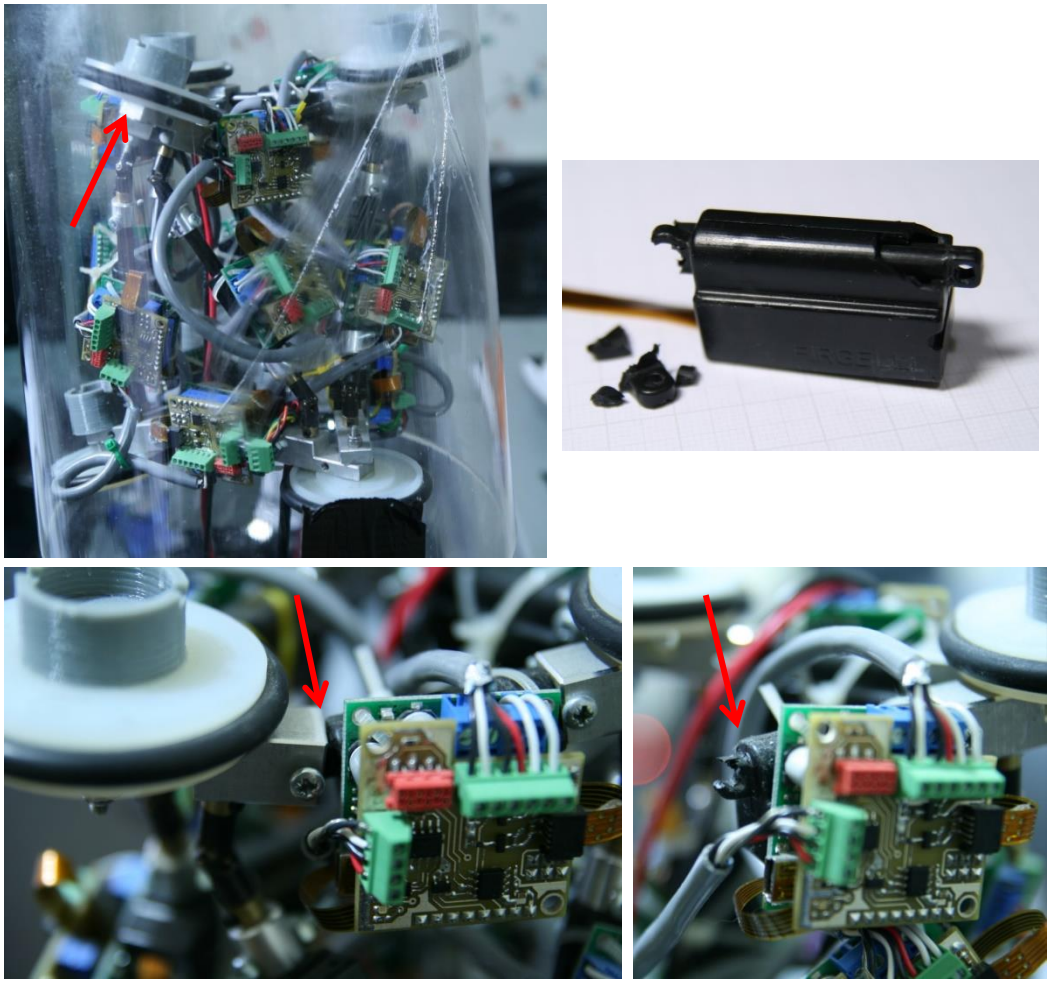


Figure 7.30 Breakage of a linear actuator as a result of excessive bending moments generated while gripping the pipe wall.

The bending also affects the internal joint between the linear actuator piston and the main linear actuator body. As the actuator extends, the bending moment causes the housing of the actuator to split as the piston is forced up against it, as shown in Figure 7.31.

The act of bending also reduces the gripping force available at the support discs in two ways. The first is that rather than the actuator force being completely transmitted through a stiff structure to the support discs, some of that force is being wasted in bending the structure, resulting in a reduced force at the support discs. The second is that as the ring assembly bends, the support discs no longer act perpendicularly to the pipe wall, but at an angle, as shown in Figure 7.29. This results in the remaining actuator force being split into two components, as shown previously in Figure 7.23, further reducing the actual gripping force.

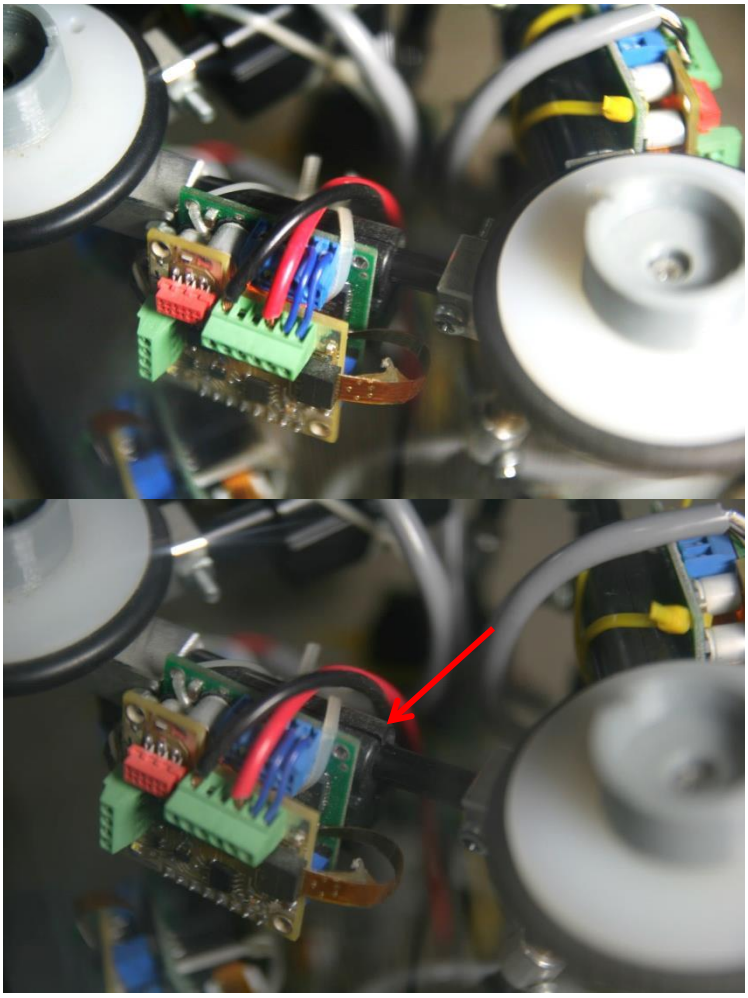


Figure 7.31 Splitting of the linear actuator case due to excessive bending moments.

It is therefore evident that eliminating any bending in the rings is important in order to improve the robot's structural integrity and to also help maximise the gripping force. This could be done in a number of different ways:

- A new design for the support discs could be found so that they lie on the same longitudinal axis as the linear actuators.
- Improving the stiffness of the connection point between the linear actuator and the aluminium connection parts, either by redesigning the part, or selecting an actuator with a more compatible mating point.
- Selecting actuators that are constructed from stiffer materials and are designed to withstand lateral loads.
- Adding additional support structures, such as linear slides, to the rings in order to counteract the lack of stiffness in other parts of the design and to absorb any bending moments.

Despite the problems and limitations faced by the robot, the experiment demonstrated that the robot was capable of navigating bent pipes, and the

problems faced by the robot are ones that can be addressed by re-examining the design and enhancing certain aspects of it.

7.4 Summary

In this chapter, the theoretical work that was done on the robot's design was verified experimentally. Two main aspects were verified: the robot's IK-equations and the robot's ability to navigate straight and bent pipe sections.

Before embarking on the IK validation, the linear actuators were first calibrated and their calibration equations derived. This information was then used to derive positional accuracy values for each of the actuators. The IK-equation validation was then carried out with the help of a motion capture system. After identifying suitable locations on the robot for the motion capture markers, the mathematical equations that related the marker positions to the robot's target location and orientation were derived. These were then integrated into the robot's control software so that it could generate the theoretical marker positions every time it sent positional commands to the robot. The IK validation experiment was then carried out by programming the robot to move to a wide variety of different positions, chosen randomly, while the motion capture system recorded the actual positions of all the markers. Once the recording was completed, the motion capture data was processed in order to extract the theoretical marker coordinates for each of the different positions, compensating for the differences in the robot- and motion capture-axis systems. The actual and theoretical marker positions were then compared.

It was found that the actual positions matched the theoretical ones, with the robot's average positional accuracy being less than 3mm, proving that the IK equations were a valid representation of the robot's geometry. There were deviations between the actual and theoretical positions, with the deviations of the top markers being higher than those of the bottom markers, and it was demonstrated that there was a correlation between the known errors of the system and the measured errors.

Once the IK equations were validated, work was begun to demonstrate the robot's ability to climb inside straight and bent pipes.

First, the robot was tested inside a straight pipe. Using a conventional inchworm-style locomotion method, it was demonstrated that the robot was capable of climbing up a vertical pipe. Methods of improving the robot's climbing speed were discussed, which involved the use of actuators with larger extensions and higher extension speeds.

Secondly, the robot's ability to climb a bent pipe was demonstrated. Based on the robot's bend radius limitations that were derived in the previous chapter, a target bend radius was chosen. Due to the lack of availability of off-the-shelf pipes of that size, it was decided to construct the pipe as a segmented bend. It was demonstrated mathematically how this would be a suitable representation due to the expectation that navigating a segmented bend would be more challenging than navigating a swept bend of the same bend radius.

The robot was demonstrated to successfully navigate the segmented bend. This was carried out under manual control and using a control strategy that aimed to keep the robot's rings perpendicular to the pipe's longitudinal axis, in order to maximise the gripping force. During the experiment, various issues were observed. The robot's limited movement range hindered its ability to keep the rings perpendicular to the pipe's axis, which compromised the robot's grip on the pipe wall during certain motions. The lack of rigidity in some of the robot's joints, coupled with the mechanical arrangement of the actuators and support discs led to excessive bending moments in the robot ring which resulted in breakage of two actuators. Each of these issues was explored in detail, with their causes outlined and methods for resolving them proposed.

Overall, the work presented in this chapter validated the mathematical models that govern the robot's movement and also proved that the robot was physically capable of climbing up both straight and bent pipes. Although there were various problems faced by the robot during the experiments, methods of addressing those problems were suggested and ultimately, these problems did not prevent the robot from fulfilling its functions. Therefore, it can be concluded that the robot design concept presented in this thesis is potentially capable of successfully navigating in a water pipe environment.

Chapter 8
Summary, Conclusions and Future Work

8.1 Assessment of the Research Objectives

In Chapter 1, a set of research objectives are outlined. This section evaluates the extent to which the objectives were achieved.

1. Review the existing literature on the pipe environment and how water companies manage it.

In Chapter 1, information from the annual reports for the five largest water companies in the UK was presented. This helped to identify the importance of water pipes, being a large proportion of a country's infrastructure, and also provided some measured figures that showed how water leakage continues to be a major issue for water companies and is a large source of financial loss.

The first half of Chapter 2 aimed to understand the water pipe environment, the problems faced in it and how water companies tackle those problems. A summary of the typical environmental parameters of a water pipe environment was presented, along with the causes of leaks and bursts. This was followed by a review of the various leak detection and localisation methods, categorised by their granularity, and finally a review of the various methods used to repair and replace pipes was presented.

Based on this, it was identified that leak detection and localisation was an area that could benefit from a robotic solution as it would enable a wide variety of sensors to be brought very close to a problem area, allowing for more accurate data about the problem to be retrieved. The area of pipe repair might not be as suitable, due to the large amount of extra machinery and equipment that is normally required, although some small scale localised repair work may be possible.

2. Investigate the various locomotion methods mobile and climbing robots can use, and how existing pipe inspection robots have used them. Identify the most suitable locomotion to use in this application.

In the second half of Chapter 2, a review of the various locomotion methods used by mobile robots was presented, with specific emphasis on their application in pipe inspection robots. For each locomotion method, its advantages and disadvantages were discussed. Various examples of how

these locomotion methods were implemented for in-pipe inspection robots were presented, along with a critique of each one. The benefits of each implementation were highlighted along with the problems and weaknesses that they faced.

Based on this review, it was deduced that inchworm-locomotion would be the most suitable locomotion method to use for an active water pipe inspection robot, due to its climbing efficiency. It was also noticed that the vast majority of robots were designed to operate in empty pipes and, as such, did not take into account the effect of fluid flowing past them. It was concluded that the development of a controllable autonomous robot capable of navigating inside active water pipes would be of great industrial benefit, as it would allow the robot to perform its duties inside the pipe without any interruption in service.

3. Design and construct a robot based on the chosen locomotion method.

In Chapter 3, a set of design requirements were drawn up based on the knowledge gained from the literature review, that would help guide the design process by specifying the general design aspects that the robot would need to conform to. In order to limit the scope of this project, it was decided to focus solely on the locomotion aspect. A set of design specifications were presented that this project would need to meet.

The conceptual design process is presented in Chapter 4, whereby various initial design ideas are presented and assessed, with the most appropriate design concept being chosen. Before assessing the actual concepts, the requirements and specification presented in Chapter 3 were translated into a set of specific design criteria that would specify the design parameters that the robot would need to address. Each of those parameters was explored in order to understand how they affect the robot’s ability to meet the project specification and how they should be optimised. Various different design concepts were then presented, with each one being assessed against the design criteria. A final design concept, based on a Gough-Stewart platform, was chosen as the most suitable one.

The detailed design process was then presented in Chapter 5. In this chapter, the various mechanical parts that comprise the robot, from the actuators to the structural components, are designed and specified. The main factors affecting the design were the chosen size of pipe, and the actuators that were selected. The remainder of the robot’s mechanical parts

were designed around these two factors. The design of the robot's electronic control system was also presented in this chapter.

4. Analyse the kinematics of the design and the forces it is exposed to in order to assess its feasibility and understand its limitations.

Chapter 6 presented the theoretical work that was done on the robot's design. The theoretical work aimed to achieve three things: to determine the robot's inverse kinematics equations, which form the core of the robot's control system; to understand the robot's movement limitations and the sharpest bend that it can navigate; to estimate how much force will be exerted on the robot by the flow of water and assess whether the robot will be able to overcome that.

The IK equations were derived by examining the robot's geometry and using that to determine how the robot's target position and orientation were affected by its actuators. In order to provide some preliminary validation of these equations, a simulation model of the robot was constructed in MSC Adams. The results of the simulation were then compared to the results of the IK equations in order to help confirm their validity. The simulation model was then used to draw out the robot's working envelope, defining the limits on the robot's movement.

In preparation for the robot's practical validation, it was necessary to understand the bend radii that the robot would be capable of navigating. The data from the simulation was used to generate a bend radius graph which shows the range of bend radii that the robot would be capable of navigating along with an indication of its performance at those radii.

Finally, the forces exerted on the robot by the fluid flow were estimated using CFD software. Various models were constructed with the robot positioned in several different pipe configurations and with the maximum expected fluid flow. In each case, the total force exerted on the robot by the fluid was simulated, and the maximum fluid force determined. The load that this force would exert on the actuators was then calculated. It was found that the chosen actuators would be capable of withstanding the fluid force.

5. Verify that the robot prototype conforms to the theoretical understanding.

Chapter 7 describes the practical work that was done to verify the robot's theoretical models and demonstrate its ability to operate in pipes. The first half of the chapter focuses on the validation of the robot's IK equations and

simulation model. This was done using a motion capture suite to record the actual physical position of the robot while it was being controlled by the control software. The physical locations were then compared to the theoretical ones and it was found that there was a strong correlation between the two. This proved that the mathematical and simulation models were valid representations of the robot's geometry. It was demonstrated that the differences between the theoretical and practical results were caused by the errors that were present in the robot's mechanical design.

6. Test the robot's locomotion ability in various pipes to validate its performance and ability to navigate the pipeline.

The second half of Chapter 7 described the experimental work carried out to demonstrate the robot's ability to operate in a pipeline.

The first experiment involved controlling the robot to climb up a vertical pipe. The robot was controlled using a traditional inchworm-style locomotion method and was successful in climbing up a straight, vertical pipe. The robot's climbing performance was assessed and the various factors that affect the climbing speed were outlined and discussed.

Next, the robot was tested inside a bent pipe. A suitable pipe bend radius was chosen based on the robot's limitations which were outlined in the previous chapter. Due to lack of availability, the bent pipe was constructed as a segmented bend from various straight segments, and it was shown how this was a valid representation of a swept bend. An initial motion algorithm was designed that worked on the assumption that the bend was swept, but when tested inside the pipe, it was not successful in moving the robot. It was then decided to move the robot under manual control, using a motion algorithm that attempted to keep the robot's rings perpendicular to the pipe's longitudinal axis in order to maximise the gripping force. It was demonstrated that this method was successful in moving the robot through the pipe. Through carrying out this experiment, various design issues were noticed. The effects that these issues had on the robot's climbing ability were analysed and explained and various methods to overcome them were presented.

Overall, the work done in this chapter validated the robot's design and demonstrated that it was potentially capable of moving inside the environment it was designed for.

8.2 Conclusions

Water pipes form a very crucial and extensive part of a country's infrastructure, with the length of water pipe in the UK estimated to exceed the total road length. This is a very large amount of critical infrastructure that needs to be continuously maintained. Leakage is a persistent problem faced by water pipe networks and is suffered by water companies all over the world. The proportion of water lost due to leakage is often very high and results in millions of pounds of loss for water companies, both in terms of the amount of water lost and in repair costs. Leakage also results in the loss of a precious natural commodity: clean water. The nature of pipes being underground and exposed to ground movements, corrosion, and third party damage means that this is a problem that is faced by water companies continuously. Although companies can choose to act reactively and only fix leaks and bursts when they are reported, this results in the rapid deterioration of the pipe network. Therefore, water companies need to continuously pursue a proactive approach, seeking out and repairing leaks before they become too significant. It is clear from the amount of water lost due to leakage and the huge financial penalty, that water companies face a significant problem in trying to prevent, locate and repair leaks.

Water companies use a wide range of techniques in order to identify the location and severity of leaks. They will often start with top-down broad leak assessment methods, such as district metering, and then use more accurate locating methods to eventually hone in on the exact location of a leak. However, these methods vary in their spatial resolution, their pipe material compatibility and their labour costs, with no one method capable of being used across all scenarios.

There does not appear to be any commercial robotic solution to the leak localisation problem. The use of robotics in this field has the potential to greatly improve it due to the following:

- It can allow a wide range of different sensors to be brought directly to the problem source, much closer than existing localisation methods. This allows leaks to be very accurately pinpointed and their severity to be accurately assessed, giving water companies much higher quality information on which to base their assessments and repairs. By being able to go to the problem source directly, direct rather than indirect measurement methods can be used. This means that the robotic solution is much less dependent on the pipe material compared to most other indirect localisation methods.

- The mobility and autonomy of a robotic solution means that robots could be inserted at one location in the pipe network and spread to any other location. This significantly reduces the labour costs of deploying localisation systems and allows the same platform to also be used for district-wide leak assessments, by using a swarm of robots spread throughout the network to send back information regarding the whole network.

Like leak localisation, pipe repairs can also be carried out using a wide range of different techniques, from traditional open-cut trenches, to minimally invasive trenchless methods. However, it appears that repairs require a large amount of heavy machinery and equipment. It is therefore not expected that a small, autonomous mobile platform would be capable of replacing the existing repair methodologies used by water companies. However, it may be possible that small, localised repairs could be carried out by a robotic platform, depending on the severity of the damage. A further study of how damage in water pipes starts and progresses would need to be carried out in order to assess the feasibility of this.

A review of existing pipe inspection robots shows that there has been much work done in the field and there are many examples of robots that are capable of navigating the various features of the pipe, using a wide variety of different locomotion methods. However, the vast majority of these robots have not been designed to operate in active water pipes and therefore do not take into account the effect of water flowing past. Many of these robots fill up most of the pipe’s internal space, which would not be suitable in an active pipe. According to the literature review, there are no fully controllable autonomous robots that are capable of navigating a pipeline whilst there is water flowing past. Taking into account the additional fluid forces that a robot would need to overcome, it was deduced that inchworm locomotion is the most appropriate locomotion method for climbing under heavy loads, whilst being a relatively simple locomotion strategy to control.

A unique design for an autonomous pipe inspection robot for operation in active pipelines, based on a Gough-Stewart platform, was thought up. Unlike conventional Gough-Stewart platforms, the top and bottom platforms in this new design are not fixed, but rather are formed from a triangle of linear actuators. This design affords the robot many advantages:

- It gives it 6 DOF of movement, allowing it to freely move inside the constrained pipe environment and gives it the dexterity needed to manipulate tools and instruments.

- It allows it to change its size to adapt to the pipe diameter.
- It allows it be constructed in a skeletal manner, presenting a very small frontal area to the fluid, which is important in order to minimise the drag force generated by the fluid.
- The classification of a Gough-Stewart platform as a parallel platform affords the robot a high degree of speed and accuracy, making it a very good platform for the manipulation of tools and instruments.

Several CFD simulations were carried out on the robot design in order to estimate the drag force that would be generated in various different pipe configurations. The results of the simulations showed that the drag forces were relatively small and that the robot could overcome them using off-the-shelf actuators. This demonstrated that the current robot design is potentially feasible for operation in active pipelines.

The IK equations for the robot, which form the heart of the robot's control system, were derived. A kinematic simulation model of the robot was constructed in MSC Adams. Both of these models were verified experimentally using a motion capture suite. The simulation model was subsequently used to determine the robot's working envelope and movement limitations, specifically, the smallest bend radius it would be capable of navigating.

The robot's climbing ability was demonstrated in both straight and bent vertical pipes. In both cases, the robot was capable of climbing up those pipes, proving that it had the required level of mobility to do so.

In summary, a novel parallel-platform-based inchworm robot was developed that is capable of climbing up straight and bent pipes. The robot has been designed to minimise the fluid force exerted on it and simulation results indicate that the design is successful in this aspect, with estimated fluid forces being small enough to overcome. Overall, this design serves as a successful proof of concept for an in-pipe inspection robot that is capable of operating in active pipelines. This could potentially provide water companies with a much more accurate and less labour intensive leak localisation and assessment tool than existing technologies, helping them to quickly and effectively manage their network leakage.

8.3 Future Work

Although the robot serves as a successful proof of concept, further research work is needed, both to improve the current design and to explore the

various other aspects that the robot needs to address in order to successfully operate in an active pipe.

Further work to be carried out on the current design could cover the following:

- The use of actuators with low extensions and length-ratios limited the robot's movement ability. Redesigning the robot to use actuators with high length-ratios and extensions would improve its motion ability and would open the door to assessing the robot's ability to move between varying pipe diameters.
- The design of the support rings could be improved so as to eliminate or absorb the bending moments that were generated in the ring. The design of the rest of the robot could be improved to eliminate any mechanical sloppiness and increase its stiffness and rigidity.
- Testing in a T-junction could be carried out to ensure that the robot is capable of navigating its features. Although similar to a bent pipe, it is not possible to keep the cross section of a junction circular at all times as the space inside the pipe opens out at the junction. The robot's movement algorithm may need to be adapted to compensate.
- The CFD simulation results need to be verified experimentally. Although waterproofing the working prototype may be very complex, it would be possible to perform the experiment using a dummy model of the robot, for example manufactured using rapid-prototyping, which could be inserted in an active pipe and the forces exerted on it measured.

There are many further aspects that the robot would need to address before it could be successfully used in an active water pipe environment. The development of these aspects could be split into several semi-independent projects, as outlined below:

8.3.1 Operation in a Fluid Environment

Operating continuously in a submerged environment will impose various challenges on the robot's design. The most obvious of these is the need for the robot's electronic and electro-mechanical systems to function successfully in such an environment. Therefore, the issue of waterproofing needs to be addressed. Waterproofing the robot's static components, such as the control electronics, may be relatively simple to achieve, however waterproofing the robot's moving parts, such as the actuators, may be more challenging. The fact that the robot is capable of changing its shape will add

further challenges, as any waterproofing solution needs to be able to work effectively regardless of the robot's physical configuration.

Operation in a wet environment could allow the robot to take advantage of buoyancy in order to reduce the power required from its actuators. A system could be designed to allow the robot to take on and remove water from internal air-filled cavities, thus allowing the robot to use the buoyancy force to assist it during certain motions, such as climbing a pipe.

Operation in a wet environment will also have an effect on the robot's ability to grip the pipe wall. A wet, non-clean pipe interior will reduce the friction between the robot's grippers and the pipe. The conditions inside the pipe would need to be studied and a suitable frictional material designed that would maximise the frictional force between the robot and the pipe wall. The effectiveness of such a material could be initially tested in a dry environment by suspending weights from the robot that represent the drag force. The robot could then be tested inside various pipe configurations to assess the robot's performance under the additional load and determine whether the frictional material performs as expected.

8.3.2 Power Generation

This project would aim to develop the most effective way of generating and storing power for the robot. The flow around the robot would need to be studied and different power generator designs examined and tested in order to allow the robot to generate power under different flow conditions and robot orientations. Also, the power consumption of the robot would need to evaluated and compared against the minimum power that can be generated by the robot. This is so that a suitable battery technology can be chosen that would allow the robot to perform any necessary operations (such as safely anchoring itself) if the flow falls below minimum required conditions. Suitable locations for installing the batteries and power generators would need to be found on the chassis and the effects of those additions on the locomotion and adhesion of the robot would need to be assessed.

8.3.3 Inspection and Maintenance

This project would look at developing the tools and techniques that the robot would require in order to perform inspection and maintenance tasks inside the pipe. A study of the typical causes of failure in pipelines would need to be undertaken in order to determine the failure signs that the robot would need to identify and the tools and techniques that it would require in order to report, repair or stabilise a failure. The effects of the flow on the inspection

and maintenance methods would need to be examined so that suitable sensors and tools are designed that can function effectively despite the presence of fluid flowing around them. The mounting locations and level of dexterity of these tools would need to be considered so that the robot is capable of using the tools on any point of interest regardless of its location in the pipe.

8.3.4 Autonomy and Control

This project would focus on the development of the robot's AI and communication systems. Based on information gathered previously on the physical properties and arrangement of the pipeline, suitable sensors would need to be chosen and control algorithms developed that allow the robot to determine its location inside the pipeline despite being completely enclosed and possibly underground. Likewise, a communication method would need to be developed that allows commands to be sent to the robot and information sent back while the robot is underground and enclosed by the pipe. The robot's AI would need to process the information gathered from sensors in order to allow it to detect and identify upcoming features in the pipeline and to automatically choose the correct locomotion method needed to navigate those features. The robot's AI would also need to be able to autonomously process the data it receives from its sensors regarding the state of the pipe, automatically identifying problems and relaying that information back, and possibly even undertaking minor repairs.

8.3.5 Integration

Although the development of the robot can be separated into several discrete conceptual sections, in practice each project is highly dependent on the developments in other projects and vice versa. For example, a particular choice of repair tool may increase the force exerted on the robot due to the fluid by increasing the robot's surface area. This may, in turn, require a re-examination of the robot's actuators and stronger ones might be needed, leading to an increase in the power consumption of the robot and requiring the installation of a larger battery, which in turn may reduce the space available for the repair tool. It is therefore imperative that the projects are not worked on in isolation and that any developments are linked back to previous work and potential future work in order to catch such design conflicts as early as possible. The integration step is therefore something which will need to be continuously considered as the project progresses, rather than in a single separate step.

Bibliography

[1] F. Tache, W. Fischer, R. Moser, F. Mondada, and R. Siegwart. Adapted magnetic wheel unit for compact robots inspecting complex shaped pipe structures. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 4 September 2007, Switzerland*. pp.1–6.

[2] H. Najjaran. Underwater robot for inspection of in-service transmission mains, *National Research Council Canada*, Jun-2005. [Online]. [Accessed 15 November 2009]. Available from: http://archive.nrc-cnrc.gc.ca/eng/projects/irc/underwater-robot.html.

[3] W. Neubauer. A spider-like robot that climbs vertically in ducts or pipes. In: *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems '94., 12 September 1994, Germany*. pp.1178 –85.

[4] A. Zagler and F. Pfeiffer. 'MORITZ' a pipe crawler for tube junctions. In: *IEEE International Conference on Robotics & Automation, 14 September 2003, Taiwan*. **3**. pp.2954 –2959.

[5] J. K. Ong, D. Kerr, and K. Bouazza-Marouf. Design of a semi-autonomous modular robotic vehicle for gas pipeline inspection. *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, 2003, **217**(2), pp.109 –122.

[6] Thames Water. *Thames Water Annual Report 2014*. UK, 2014.

[7] Severn Trent Water. Severn Trent Water: Facts and Figures. [Online]. Available from: https://www.stwater.co.uk/media/facts-and-figures/.

[8] United Utilities. *United Utilities Annual Report 2014*. UK, 2014.

[9] Yorkshire Water. *Yorkshire Water Annual Report and Financial Statements*. UK, 2014.

[10] Anglian Water. *Anglian Water Annual Report 2014*. UK, 2014.

[11] Office for National Statistics. Population Estimates for UK, England and Wales, Scotland and Northern Ireland, Mid-2013, *Population Estimates for UK, England and Wales, Scotland and Northern Ireland, Mid-2013*, 26-Jun-2014. [Online]. Available from: http://www.ons.gov.uk/ons/rel/pop-estimate/population-estimates-for-uk--england-and-wales--scotland-and-northern-ireland/2013/index.html.

[12] UK Department for Transport. *Road Lengths in Great Britain: 2013*. UK: UK Department for Transport, 2013.

[13] Ofwat. *Service and delivery – performance of the water companies in England and Wales 2009-10*. UK: Ofwat, 2010.

[14] Consumer Council for Water. Consumer Council for Water: Average water use, *Consumer Council for Water*. [Online]. [Accessed 08 November 2014]. Available from: http://www.ccwater.org.uk/savewaterandmoney/averagewateruse/.

[15] Yorkshire Water. *Yorkshire Water Operating Cost Analysis 2014*. UK: Yorkshire Water, 2014.

[16] How much does your water company leak?, *BBC*, 05-Apr-2012. [Online]. [Accessed 05 November 2014]. Available from: http://www.bbc.co.uk/news/uk-17622837.

[17] *Progress on Drinking Water and Sanitation 2012 Update*. Unicef, 2012.

[18] T. Okada and T. Sanemori. Vehicles in Pipe for Monitoring Inside of Pipe, MOGRER. *Transactions of the Society of Instrument and Control Engineers*, 1986, **22**(8), pp.897–903.

[19] S.-G. Roh, D. W. Kim, J.-S. Lee, H. Moon, and H. R. Choi. In-pipe robot based on selective drive mechanism. *International Journal of Control, Automation and Systems*, 2009, **7**(1), pp.105–112.

[20] A. Kuwada, S. Wakimoto, K. Suzumori, and Y. Adomi. Automatic pipe negotiation control for snake-like robot. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2 July 2008, China*. pp.558 –563.

[21] ABB. *Leakage monitoring. Reducing leakage through effective flow measurement*. UK: The ABB Group, 2011.

[22] S. Creig-Smith. To refurbish or replace steel water pipelines, that is the question. *Engineering Failure Analysis*, 2001, **8**(2), pp.107–112.

[23] D. G., A. Agapiou, K. Themistocleous, G. Toulios, S. Perdikou, L. Toulios, and C. Clayto. Detection of Water Pipes and Leakages in Rural Water Supply Networks Using Remote Sensing Techniques. In: Diofantos G. ed. *Remote Sensing of Environment - Integrated Approaches*. Croatia: InTech, 2013, pp.155–180.

[24] G. C. Galletti, P. Bocchini, F. Pinelli, and R. Pozzi. Rapid quality checking of polyethylene water/gas-pipelines: a model based on pyrolysis-gas chromatography/mass spectrometry and discriminant analysis. *Journal of Analytical and Applied Pyrolysis*, 2003, **70**(2), pp.291–302.

[25] M. Abdel-Aleem, C. C. Chibelushi, and M. Moniri. Multisensor data fusion for the simultaneous location and condition assessment of underground water pipelines. In: *2011 IEEE International Conference on Networking, Sensing and Control (ICNSC), April 2011*. pp.416–421.

[26] A. W. O. Hunaidi. Acoustic methods for locating leaks in municipal water pipe networks. presented at the International Water Demand Management Conference, Jordan. pp.1–14.

[27] S. Srikanth, T. S. N. Sankaranarayanan, K. Gopalakrishna, B. R. V. Narasimhan, T. V. K. Das, and S. K. Das. Corrosion in a buried pressurised water pipeline. *Engineering Failure Analysis*, 2005, **12**(4), pp.634–651.

[28] D. Loureiro, C. Amado, A. Martins, D. Vitorino, A. Mamade, and S. T. Coelho. Water distribution systems flow monitoring and anomalous event detection: A practical approach. *Urban Water Journal*, 2015, pp.1–11.

[29] H. A. Kishawy and H. A. Gabbar. Review of pipeline integrity management practices. *International Journal of Pressure Vessels and Piping*, 2010, **87**(7), pp.373–380.

[30] R. Liemberger, P. Marin, and B. Kingdom. *The challenge of reducing non-revenue water (NRW) in developing countries - how the private sector can help : a look at performance-based service contracting*. The World Bank, 2006, pp.1–52.

[31] R. Puust, Z. Kapelan, D. A. Savic, and T. Koppel. A review of methods for leakage management in pipe networks. *Urban Water Journal*, 2010, **7**(1), pp.25–45.

[32] P. F. Boulos and A. S. Aboujaoude. Managing Leaks Using Flow Step-Testing, Network Modeling, and Field Measurement. *Journal - American Water Works Association*, 2011, **103**(2), pp.90–97.

[33] R. Pilcher, S. Hamilton, H. Chapman, D. Field, B. Ristovski, and S. Stapely. *Leak location and repair. Guidance Notes*. International Water Association, 2007.

[34] H. Chapman. Development of a Successful Internal Leak Detection and Pipeline Condition Assessment Technology for Large Diameter Pipes. presented at the 6th Annual WIOA NSW Water Industry Engineers & Operators Conference, UK.

[35] I. El-Baroudy and L. Pinter. *Trenchless Technologies and Work Practices Review for Saskatchewan Municipalities*. Canada: PINTER & Associates, 2013.

[36] R. Zlokovitz and I. Juran. *Trenchless Technology Solutions for Professional Training Sessions*. Urban Utility Center, 2005.

[37] N. Jones. PVC – Review of Available Technologies for Trenchless Pipeline Applications. In: *Plastic Pipes Conference, 22 September 2014, Chicago, IL, USA*.

[38] Pipe Bursting, *Pipe Bursting*. [Online]. [Accessed 23 August 2015]. Available from: http://www.undergroundsolutions.com/pipe-bursting.php.

[39] CIPP “green” resins now available, *CIPP ‘green’ resins now available*. [Online]. [Accessed 23 August 2015]. Available from: http://trenchlessinternational.com/news/cipp_/084269/.

[40] Cured In Place Liner, *Cured In Place Liner - Tri-State Utilities*. [Online]. [Accessed 23 August 2015]. Available from: http://www.tristateutilities.com/our_services/cured-in-place-liner/.

[41] O. Tatar, D. Mandru, and I. Ardelean. Development of mobile minirobots for in pipe inspection tasks. *Mechanika*, 2007, **68**(6), pp.60 –4.

[42] A. Seeni, B. Schafer, B. Rebele, and N. Tolyarenko. Robot mobility concepts for extraterrestrial surface exploration. In: *IEEE Aerospace Conference, 1 March 2008, USA*. pp.1–14.

[43] A. Seeni, B. Schafer, and G. Hirzinger. Robot Mobility Systems for Planetary Surface Exploration - State-of-the-Art and Future Outlook: A Literature Survey. In: *Aerospace Technologies Advancements*. Vukovar, Croatia: InTech, 2010, pp.10–208.

[44] A. A. Transeth, K. Y. Pettersen, and P. Liljebck. A survey on snake robot modeling and locomotion. *Robotica*, 2009, **27**(7), pp.999–1015.

[45] J. K. Hopkins, B. W. Spranklin, and S. K. Gupta. A survey of snake-inspired robot designs. *Bioinspiration & Biomimetics*, 2009, **4**(2), p.021001 (19 pp.).

[46] Zhixiang Li, Jin Zhu, Changzhong He, and Wei Wang. A new pipe cleaning and inspection robot with active pipe-diameter adaptability based on ATmega64. In: *Electronic Measurement & Instruments, 2009. ICEMI ’09. 9th International Conference on, 2009*. pp.2–616–2–619.

[47] V. Kepplin, K.-U. Scholl, and K. Berns. A mechatronic concept for a sewer inspection robot. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 19 September 1999, USA*. pp.724 –729.

[48] H.-B. Kuntze and H. Haffner. Experiences with the development of a robot for smart multisensoric pipe inspection. In: *IEEE International Conference on Robotics & Automation, 16 May 1998, Belgium*. **2**. pp.1773 –1778.

[49] D. Wu, H. Ogai, Y. Yeh, K. Hirai, T. Abe, and G. Sato. Pipe inspection robot using a wireless communication system. *Artificial Life and Robotics*, 2009, **14**(2), pp.154–159.

[50] A. A. F. Nassiraei, Y. Kawamura, A. Ahrary, Y. Mikuriya, and K. Ishii. Concept and design of a fully autonomous sewer pipe inspection mobile robot 'KANTARO'. In: *IEEE International Conference on Robotics and Automation, 2007, Italy*. pp.136–143.

[51] T. Okada and T. Kanade. A three-wheeled self-adjusting vehicle in a pipe, FERRET-1. *International Journal of Robotics Research*, 1987, **6**(4), pp.60–75.

[52] A. Ahrary. *Service Robot Applications*. Croatia: InTech, 2008.

[53] A. A. F. Nassiraei, Y. Kawamura, A. Ahrary, Y. Mikuriya, and K. Ishii. A new approach to the sewer pipe inspection: Fully autonomous mobile robot 'KANTARO'. In: *32nd Annual Conference on IEEE Industrial Electronics, 6 November 2006, France*. pp.4088–4093.

[54] Y. Kawaguchi, I. Yoshida, H. Kurumatani, T. Kikuta, and Y. Yamada. Internal pipe inspection robot. In: *IEEE International Conference on Robotics and Automation, 10 May 1999, USA*. **1**. pp.857–862.

[55] F. Tache, W. Fischer, G. Caprari, R. Siegwart, R. Moser, and F. Mondada. Magnebike: A magnetic wheeled robot with high mobility for inspecting complex-shaped structures. *Journal of Field Robotics*, 2009, **26**(5), pp.453–476.

[56] F. Tache, W. Fischer, R. Siegwart, R. Moser, and F. Mondada. Compact magnetic wheeled robot with high mobility for inspecting complex shaped pipe structures. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 29 November 2007, USA*. pp.261–266.

[57] S. G. Roh, S. M. Ryew, J. H. Yang, and H. R. Choi. Actively steerable in-pipe inspection robots for underground urban gas pipelines. presented at the IEEE International Conference on Robotics & Automation, Korea. **1**. pp.761–766.

[58] S.-G. Roh and H. R. Choi. Differential-drive in-pipe robot for moving inside urban gas pipelines. *IEEE Transactions on Robotics*, 2005, **21**(1), pp.1–17.

[59] H. M. Kim, J. S. Suh, Y. S. Choi, T. D. Trong, H. Moon, J. Koo, S. Ryew, and H. R. Choi. An In-pipe robot with multi-axial differential gear mechanism. In: *26th IEEE/RSJ International Conference on Intelligent Robots and Systems: New Horizon, 3 November 2013, Japan*. pp.252–257.

[60] C. Jun, Z. Deng, and S. Jiang. Study of locomotion control characteristics for six wheels driven in-pipe robot. presented at the IEEE International Conference on Robotics and Biomimetics, China. pp.119 –124.

[61] X. Zhang and H. Chen. Research on individual wheel drive and control of mobile robot at bent pipeline. presented at the Emerging Technologies and Factory Automation (ETFA), International Conference on. **2**. pp.391–395.

[62] O. Tătar, C. Cirebea, A. Alutei, and D. Mândru. The design of adaptable indoor pipeline inspection robots. In: *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on, 2010*. **1**. pp.1–4.

[63] Jong-Hoon Kim, G. Sharma, and S. S. Iyengar. FAMPER: A fully autonomous mobile robot for pipeline exploration. In: *Industrial Technology (ICIT), 2010 IEEE International Conference on, 2010*. pp.517–523.

[64] L. A. Mateos, Kai Zhou, and M. Vincze. Towards efficient pipe maintenance: DeWaLoP in-pipe robot stability controller. In: *IEEE International Conference on Mechatronics and Automation (ICMA), 5 August 2012, China*. pp.1–6.

[65] J. Chen, T. Chen, and Z. Deng. Design method of modular units for articulated in-pipe robot inspecting system. In: *2nd International Conference on Digital Manufacturing and Automation, 5 August 2011*. pp.389–392.

[66] A. Kakogawa and S. Ma. Design of an underactuated parallelogram crawler module for an in-pipe robot. In: *IEEE International Conference on Robotics and Biomimetics (ROBIO), 12 December 2013, China*. pp.1324–9.

[67] A. Kakogawa, Shugen Ma, and S. Hirose. An in-pipe robot with underactuated parallelogram crawler modules. In: *2014*. pp.1687–92.

[68] S. Fujiwara, R. Kanehara, T. Okada, and T. Sanemori. An articulated multi-vehicle robot for inspection and testing of pipeline interiors. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 1993, Japan*. pp.509 –16.

[69] T. Okada and T. Sanemori. MOGRER: a vehicle study and realization for in-pipe inspection tasks. *IEEE Journal of Robotics and Automation*, 1987, **RA**-**3**(6), pp.573 –82.

[70] P. Li, S. Ma, B. Li, and Y. Wang. Design of a mobile mechanism possessing driving ability and detecting function for in-pipe inspection. presented at the IEEE International Conference on Robotics and Automation, USA. pp.3992 –3997.

[71] P. Li, S. Ma, B. Li, and Y. Wang. Development of an adaptive mobile robot for in-pipe inspection task. presented at the IEEE International Conference on Mechatronics and Automation, China. pp.3622 –3627.

[72] P. Li, S. Ma, B. Li, and Y. Wang. Multifunctional mobile units with a same platform for in-pipe inspection robots. presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, France. pp.2643 –2648.

[73] T. Sren, Q. Sliu, K. Sbao, and Y. Schen. Mobility and eccentricity analysis of a helical belt drive in-pipe robot. In: *IEEE International Conference on Mechatronics and Automation, 4 August 2013, Japan*. pp.1507–1512.

[74] Q. Liu, Y. Chen, T. Ren, and Y. Wei. Optimized inchworm motion planning for a novel in-pipe robot. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 2014, **228**(7), pp.1248–1258.

[75] J. J. Park, J. W. Moon, H. Kim, S. C. Jang, D. G. Kim, K. Ahn, S. M. Ryew, H. Moon, and H. R. Choi. Development of the untethered in-pipe inspection robot for natural gas pipelines. In: *10th International*

Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 31 November 2013, Korea. pp.55–58.

[76] G. Vradis, D. Driscoll, H. Schempf, and R. Baker. Industry gains a new tool for long-range visual inspection of live gas pipe. *Pipeline and Gas Journal*, 2005, **232**(4), pp.14 –18.

[77] H. Schempf, E. Mutschler, A. Gavaert, G. Skoptsov, and W. Crowley. Visual and nondestructive evaluation inspection of live gas mains using the explorer family of pipe robots. *Journal of Field Robotics*, 2010, **27**(3), pp.217–249.

[78] M. Ciszewski, T. Buratowski, M. Giergiel, P. Malka, and K. Kurc. Virtual Prototyping, Design and Analysis of an In-Pipe Inspection Mobile Robot. *Journal of Theoretical and Applied Mechanics*, 2014, **52**(2), pp.417–29.

[79] Z. Wang and H. Gu. A bristle-based pipeline robot for Ill-constraint pipes. *IEEE/ASME Transactions on Mechatronics*, 2008, **13**(3), pp.383 –392.

[80] L. Sun, P. Sun, L. Lian, X. Qin, and Z. Gong. Improvement of characteristics of in-pipe micro robot. In: *International Symposium on Micromechatronics and Human Science., 1999*. pp.153–156.

[81] Z. Hu and E. Appleton. Dynamic characteristics of a novel self-drive pipeline pig. *IEEE Transactions on Robotics*, 2005, **21**(5), pp.781 –9.

[82] Y.-J. Kim, K.-H. Yoon, and Y.-W. Park. Development of the inpipe robot for various sizes. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 14 July 2009, Singapore*. pp.1745 –9.

[83] H. Yu, P. Ma, and C. Cao. A novel in-pipe worming robot based on SMA. presented at the IEEE International Conference on Mechatronics and Automation, Canada. pp.923 –927.

[84] L. Chaodong and H. Xiaojing. A bio-mimetic pipe crawling microrobot driven based on self-excited vibration. In: *IEEE International Conference on Robotics and Biomimetics, 15 December 2007, China*. pp.984 –988.

[85] R. Aracil, R. Saltaren, and O. Reinoso. Parallel robots for autonomous climbing along tubular structures. *Robotics and Autonomous Systems*, 2003, **42**(2), pp.125 –134.

[86] J. M. Sabater, R. J. Saltaren, R. Aracil, E. Yime, and J. M. Azorin. Teleoperated parallel climbing robots in nuclear installations. *Industrial Robot*, 2006, **33**(5), pp.381–386.

[87] K. Harigaya, K. Adachi, T. Yanagida, M. Yokojima, and T. Nakamura. Development of a peristaltic crawling robot for sewer pipe inspection. In: *IEEE International Conference on Mechatronics (ICM), 27 March 2013*. pp.267–72.

[88] T. Tanaka, K. Harigaya, and T. Nakamura. Development of a peristaltic crawling robot for long-distance inspection of sewer pipes. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 8 July 2014, France*. pp.1552–7.

[89] Y. Zhang, M. Zhang, H. Sun, and Q. Jia. Motion analysis of a flexible squirm pipe robot. In: *International Conference on Intelligent System Design and Engineering Application, ISDEA, 13 October 2010*. **1**. pp.959–963.

[90] A. Yamaguchi, K. Takemura, S. Yokota, and K. Edamura. An in-pipe mobile robot using electro-conjugate fluid. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 2011, **5**(3), pp.214–26.

[91] S. Sakamoto, F. Hara, H. Hosokai, H. Kinoshita, and Y. Abe. Parallel-link robot for pipe inspection. In: *IECON 2005: 31st Annual Conference of IEEE Industrial Electronics Society, 6 November 2005*. pp.345–350.

[92] TREPA: Palmtree Climbing Teleoperated Robot, *TREPA: Palmtree Climbing Teleoperated Robot*. [Online]. [Accessed 31 August 2015]. Available from: http://arvc.umh.es/proyectos/trepa/index.php?type=proy&dest=inicio&lang=en&vista=normal&idp=trepa&ficha=on.

[93] J. C. Zagal, C. Armstrong, and S. Li. Deformable Octahedron Burrowing Robot. presented at the 13th International Conference on the Simulation and Synthesis of Living Systems, USA. pp.431–438.

[94] S. A. Fjerdingen, P. Liljeback, and A. A. Transeth. A snake-like robot for internal inspection of complex pipe structures (PIKo). In: *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, 2009*. pp.5665–5671.

[95] C. Wright, A. Johnson, A. Peck, Z. McCord, A. Naaktgeboren, P. Gianfortoni, M. Gonzalez-Rivero, R. Hatton, and H. Choset. Design of a modular snake robot. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 29 November 2007, USA*. pp.2609 – 2614.

[96] S. Wakimoto, J. Nakajima, M. Takata, T. Kanda, and K. Suzumori. A micro snake-like robot for small pipe inspection. In: *International Symposium on Micromechatronics and Human Science, 19 October 2003, Japan*. pp.303 –8.

[97] K. Suzumori, S. Wakimoto, and M. Takata. A miniature inspection robot negotiating pipes of widely varying diameter. presented at the IEEE International Conference on Robotics & Automation, Taiwan. **2**. pp.2735 –2740.

[98] A. Kuwada, K. Tsujino, K. Suzumori, and T. Kanda. Intelligent actuators realizing snake-like small robot for pipe inspection. presented at the IEEE International Symposium on Micro-NanoMechatronics and Human Science, Japan. pp.1–6.

[99] F. Enner, D. Rollinson, and H. Choset. Motion estimation of snake robots in straight pipes. In: *IEEE International Conference on Robotics and Automation (ICRA), 6 May 2013, Germany*. pp.5168–73.

[100] A. Brunete, J. E. Torres, M. Hernando, and E. Gambao. Multi-drive control for in-pipe snakelike heterogeneous modular micro-robots. In: *IEEE International Conference on Robotics and Biomimetics, 15 December 2007, China*. pp.490 –495.

[101] A. Brunete, M. Hernando, and E. Gambao. Drive modules for low diameter pipe inspection multi-configurable micro-robots. pp.4351 –3.

[102] X. Zhou and S. Bi. A survey of bio-inspired compliant legged robot designs. *Bioinspiration and Biomimetics*, 2012, **7**(4).

[103] G. Carbone and M. Ceccarelli. Legged Robotic Systems. In: *Cutting Edge Robotics*. Rijeka, Croatia: InTech, 2005, pp.553–76.

[104] J. Liu, M. Tan, and X. Zhao. Legged robots — an overview. *Transactions of the Institute of Measurement and Control*, 2007, **29**(2), pp.185–202.

[105] W.-S. Chu, K.-T. Lee, S.-H. Song, M.-W. Han, J.-Y. Lee, H.-S. Kim, M.-S. Kim, Y.-J. Park, K.-J. Cho, and S.-H. Ahn. Review of biomimetic underwater robots using smart actuators. *International Journal of Precision Engineering and Manufacturing*, 2012, **13**(7), pp.1281–1292.

[106] R. Stopforth, S. Holtzhausen, G. Bright, N. S. Tlale, and C. M. Kumile. Robots for search and rescue purposes in urban and underwater environments: a survey and comparison. In: *15th International Conference on Mechatronics and Machine Vision in Practice, 2 December 2008, New Zealand*. pp.476–80.

[107] M. Sfakiotakis, D. M. Lane, and J. B. C. Davies. Review of fish swimming modes for aquatic locomotion. *IEEE Journal of Oceanic Engineering*, 1999, **24**(2), pp.237–52.

[108] J. Sattar, G. Dudek, O. Chiu, I. Rekleitis, P. Giguere, A. Mills, N. Plamondon, C. Prahacs, Y. Girdhar, M. Nahon, and J.-P. Lobos. Enabling autonomous capabilities in underwater robotics. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, 22 September 2008, France*. pp.3628–34.

[109] J. Quarini and S. Shire. A Review of Fluid-Driven Pipeline Pigs and their Applications. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 2007, **221**(1), pp.1–10.

[110] J. Faber Archila and M. Becker. Study of Robots to Pipelines, Mathematical Models and Simulation. In: *Latin American Robotics Symposium (LARS 2013), 21 October 2013, Peru*. pp.18–23.

[111] J. M. M. Tur and W. Garthwaite. Robotic devices for water main in-pipe inspection: A survey. *Journal of Field Robotics*, 2010, **27**(4), pp.491–508.

[112] J. Nelson. Leak detector is on a roll, *Canadian Business*, 19-May-2011. [Online]. [Accessed 31 August 2015]. Available from: http://www.canadianbusiness.com/technology-news/leak-detector-is-on-a-roll/.

[113] C. Choi and K. Youcef-Toumi. Robot design for high flow liquid pipe networks. In: *26th IEEE/RSJ International Conference on Intelligent Robots and Systems: New Horizon, IROS 2013, 3 November 2013, Japan*. pp.246–251.

[114] U. V. Painumgal, B. Thornton, T. Uray, and Y. Nose. Positioning and control of an AUV inside a water pipeline for non-contact in-service inspection. In: *OCEANS 2013 MTS/IEEE San Diego Conference: An Ocean in Common, 23 September 2013, USA*.

[115] C. Matthews. *IMechE Engineers' Data Book*. Chichester, England: Wiley-Blackwell, 2005.

[116] Drag coefficient, *Wikipedia, the free encyclopedia*, 08-Aug-2015. [Online]. [Accessed 23 August 2015]. Available from: https://en.wikipedia.org/w/index.php?title=Drag_coefficient&oldid=675067672.

[117] Boundary layer, *Wikipedia, the free encyclopedia*, 05-Aug-2015. [Online]. [Accessed 23 August 2015]. Available from:

https://en.wikipedia.org/w/index.php?title=Boundary_layer&oldid=674636996.
[118] L. J. Clancy. *Aerodynamics*. New York: Wiley, 1975.
[119] Coefficients Of Friction. [Online]. [Accessed 27 December 2012]. Available from: http://www.roymech.co.uk/Useful_Tables/Tribology/co_of_frict.htm.
[120] Pololu Jrk 21v3 USB Motor Controller with Feedback, *Pololu Robotics & Electronics*. [Online]. [Accessed 31 August 2015]. Available from: https://www.pololu.com/product/1392.
[121] L.-W. Tsai. *Robot Analysis and Design: The Mechanics of Serial and Parallel Manipulators*. New York, NY, USA: John Wiley & Sons, Inc., 1999.
[122] D. Karimi and M. J. Nategh. A Statistical Approach to the Forward Kinematics Nonlinearity Analysis of Gough-Stewart Mechanism. *Journal of Applied Mathematics*, 2011, **2001**, pp.1–17.
[123] J. Borras, F. Thomas, and C. Torras. New Geometric Approaches to the Analysis and Design of Stewart-Gough Platforms. *IEEE/ASME Transactions on Mechatronics*, 2014, **19**(2), pp.445–455.
[124] Guillermo Palau Salvador, Jaime Arviza Valvedre, and Steven Frankel. Three-Dimensional Control Valve with Complex Geometry: CFD Modeling and Experimental Validation. In: *34th AIAA Fluid Dynamics Conference and Exhibit*. American Institute of Aeronautics and Astronautics, 2004.
[125] J. Ferrari and Z. Leutwyler. Fluid Flow Force Measurement Under Various Cavitation State on a Globe Valve Model. *ASME 2008 Pressure Vessels and Piping Conference*, 2008, **4**, pp.157–165.
[126] A. Del Toro. *Computational Fluid Dynamics Analysis of Butterfly Valve Performance Factors*. MSc Thesis, Utah State University, 2012.
[127] Research Equipment in the Lab, *CompNeurosci Lab - Dr Gunnar BLOHM*, 23-Aug-2015. [Online]. Available from: http://www.compneurosci.com/equipment.html.
[128] Optotrak Accessories, *Measurement Sciences*. [Online]. [Accessed 23 August 2015]. Available from: http://www.ndigital.com/msci/products/optical-accessories/.