Design of a bio-inspired robot for inspection of pipelines

Swaminath Venkateswaran

THESE DE DOCTORAT DE

L'ÉCOLE CENTRALE DE NANTES

ECOLE DOCTORALE N° 602
Sciences pour l'Ingénieur
Spécialité : *Robotique-Mécanique*

Par

Swaminath VENKATESWARAN

Conception d'un robot bio-inspiré pour l'inspection des canalisations

Thèse présentée et soutenue à Nantes, le 04 novembre 2020
Unité de recherche : UMR 6004, Laboratoire des Sciences du Numérique de Nantes (LS2N)

Rapporteurs:

Belhassen Chedli BOUZGARROU	Professeur des universités, SIGMA Clermont-Ferrand
Pierre RENAUD	Professeur des universités, INSA Strasbourg

Composition du Jury :

Président	: Pierre MORETTO	Professeur des universités, Université de Toulouse III
Examinateurs	: Belhassen Chedli BOUZGARROU	Professeur des universités, SIGMA Clermont-Ferrand
	Olivier COMPANY	Maître de conférences, Université Montpellier II
	Ramakrishnan RAMACHANDRAN	Associate professor, Vellore Institute of Technology (Inde)
	Pierre RENAUD	Professeur d'université, INSA, Strasbourg
	Margot VULLIEZ	Maître de conférences, Université de Poitiers
Dir. de thèse	: Damien CHABLAT	Directeur de recherche, Ecole Centrale de Nantes

Résumé

Les robots d'inspection de canalisations jouent un rôle important dans des industries telles que le nucléaire, la chimie et les eaux usées. Ils peuvent opérer avec précision dans un environnement irradié ou pollué, réduisant ainsi les risques pour les humains. Cette thèse porte sur la conception d'un robot bio-inspiré pour l'inspection des canalisations. La thèse commence par l'étude du cas d'un robot d'inspection bio-inspiré rigide qui a été développé au LS2N, France pour AREVA. Des modèles statiques et dynamiques sont développés pour comprendre les forces de serrage et les couples des actionneurs du robot. Des validations expérimentales sont également effectuées sur le prototype pour interpréter les forces d'actionnement en temps réel. Pour améliorer sa mobilité, l'architecture du robot est rendue flexible par l'ajout d'un mécanisme de tenségrité. Deux types de mécanismes de tenségrité sont proposés et analysés avec des méthodes algébriques pour comprendre leurs limites d'inclinaison et pour connaître l'influence des paramètres de conception. Des expériences sont réalisées sur l'un des prototypes des mécanismes de tenségrité développés au LS2N avec deux types de trajectoire en positions horizontale et verticale. Ensuite, une optimisation est réalisée pour identifier les moteurs qui peuvent permettre du robot d'inspection de canalisation flexible de passer les coudes et les jonctions pour une plage donnée de diamètres de tuyaux. Une maquette numérique du robot flexible est réalisée dans un logiciel de CAO.

Mots-clés: Inspection de la canalisation, robots bio-inspirés, tenségrité, singularités, optimisation.

Abstract

Piping inspection robots play an important role in industries such as nuclear, chemical and sewage. They can perform the assigned task with better accuracy and at the same time, they can operate within an irradiated or a polluted environment thereby reducing the risks for humans. This doctoral thesis focuses on the design of a bio-inspired robot for the inspection of pipelines. The thesis begins with the case study of a rigid bio-inspired piping inspection robot which was developed at LS2N, France for a project with AREVA. Static and dynamic force models are developed to understand the clamping forces and the torques on the actuators of the robot. Experimental validations are then done on the prototype to interpret the real-time actuator forces. In order to improve mobility, the robot architecture is made flexible by the addition of a tensegrity mechanism. Two types of tensegrity mechanisms are proposed and analyzed using algebraic methods to understand their tilt limits and to identify the influences on the design parameters. Experiments are performed on one of the prototypes of the tensegrity mechanism developed at LS2N for two types of trajectories in the vertical and horizontal orientations. An optimization approach is then being implemented to identify the sizes of motors that can permit the flexible piping inspection robot to overcome bends and junctions for a given range of pipeline diameters. A digital model of the flexible robot is then realized in CAD software.

Keywords: Piping inspection, Bio-inpired robots, Tensegrity, Singularities, Optimization

Dedicated to Dad, Mom, Poori, Pavan, Yuki & Sanjana

Acknowledgements

"No one who achieves success does so without acknowledging the help of others. The wise and confident acknowledge this help with gratitude." - Alfred North Whitehead

First and foremost, I would like to thank my thesis director, Damien Chablat for proposing to me this interesting thesis topic. I am extremely thankful for his support, ideas and not to be forgotten his immense patience during all our weekly discussions. Be it technical or any personal conversation, Damien ensured that I always felt comfortable. His in-depth knowledge and problem-solving approaches to the design of mechanisms helped me to develop my curiosity and broaden my technical skills in the field of robotics design. Apart from the thesis works, he helped me discover my strong potential as a teacher and a supervisor in training students and addressing their technical difficulties. Under his guidance, I became more and more autonomous and learned to work within strict deadlines.

I would also like to thank Philippe Wenger and Matthieu Furet for their interest in the participation of the study of Tensegrity mechanisms. Their valuable inputs helped me a lot to explore further on the design and analysis of tensegrity structures. I also want to thank Stéphane Jolivet, without whom I couldn't have carried out my experiments. I really want to appreciate his patience and support in printing parts for conducting experiments, be there any errors or problems, he does the job with a big smile and full energy. I also want to thank Philippe Lemoine for his support extended before conducting the experiments and also by saving me from burning the micro-controllers.

I also want to thank Fouad Bennis, who also recognized me as a teacher and also for his immense technical support whenever I had doubts in optimization. Also special thanks to Anne-Laure Frémondière of international office for her continuous and friendly discussions since before my arrival here in France. I also want to thank the French language department of Centrale Nantes, especially Silvia, Karen and Cathy, who laid the foundation in my career to learn and speak good french. Also thanks to all administrative members of LS2N and Centrale Nantes Virginie, Christine, Carine, Edith and people at the scolarité for making all the complex administrative tasks look much simpler.

I am deeply indebted to my Dad and Mom. They really encouraged me and still doing the same to pursue things that I wish to do in my career. Without their tireless efforts, I couldn't have imagined my Master's degree, my doctoral research and even my presence here in France.

I am very much thankful to my sister, Poorni, for her immense support and patience towards me. We have always had different opinions towards things in our life, despite all this, our affection, support and those silly fights between each other have no limits. Her encouragement and personal support since when I was a kid is the key factor on what I have become and achieved in my life today. Also, I want to thank my brother-in-law, Pavan, for his love and care towards our family and especially in taking care of my parents during the COVID-19 pandemic.

Special thanks to my fiancée and future wife, Sanjana. She stepped into my life when I was personally down and depressed. Her care, her love, her affection towards me have no limits. Also special thanks to her for cooking me delicious dishes and her patience in tolerating my fitness freak goals as well as my childish behaviours. I am really excited and happy to start my life with her.

Special thanks to my family away from India in France: Murali and Shobana at Bordeaux, for making me comfortable all the time and treating me as their own family member, especially when I was personally down.

My achievements wouldn't have been possible without the support of my friends. I am thankful to my school friends Preeth, Sucha, Venky and VP (VP Narayanan to be precise). They travelled throughout my life since my early school days and have personally played a vital role during the ups and downs in my life. Thanks to my travel buddies and special friends at Centrale Nantes: Sashi, Ajai, Puvi, Surenthar, Kaushik, Krisna and Prabu, for their personal support and meetups during my stay here in Nantes. I really enjoyed my time with them during my PhD journey and I wish this continues forever. Thanks to Saman, who helped me with some good latex/beamer templates during my thesis and also introducing to me the Adobe Illustrator package which helped me make good graphics for my thesis. Also special thanks to Elahe and Pierre for their personal support when I was trapped during confinement.

Contents

List of Figures

List of Tables

Abbreviations & Nomenclature

Abbreviations

CG	Center of Gravity
DKP	Direct Kinematic Problem
IKP	Inverse Kinematic Problem
NE	Newton-Euler
PWM	Pulse Width Modulation
GPIO	General-purpose Input-Output
ADC	Analog to Digital Convertor
BB	BeagleBone
S	Spherical joint
$\underline{\text{P}}$	Actuated Prismatic joint
U	Universal joint
T&T	Tilt & Torsion
CAD	Cylindrical Algebraic Decomposition
GA	Genetic Algorithm
PID	Proportional Integral Derivative
CLR	Center Line Radius
CAD	Computer Aided Design
IFToMM	International Federation for the Promotion of Mechanism and Machine Science
ASME	American Society of Mechanical Engineers

Nomenclature

θ	Rotation angle of robot about z-axis
δ	Rotation angle of pipeline about x-axis
$\mathbf{R}_x$	Rotation matrix in $\mathbb{R}^3$ about x-axis
$\mathbf{R}_y$	Rotation matrix in $\mathbb{R}^3$ about y-axis
$\mathbf{R}_z$	Rotation matrix in $\mathbb{R}^3$ about z-axis

φ	Coefficient of friction
$\mathbf{J}$	Jacobian matrix
η_f	Force transmission factor
α	Tilt angle
β	Azimuth angle
$\mathbf{R}$	Resultant Tilt & Azimuth transformation matrix
$\mathcal{C}_i$	Joint limits/Constraint equation of the i^{th} joint
U_{tot}	Total potential energy of the system
K_α	Stiffness of the mechanism with respect to tilt angle α
η	The 'x' axis Euler rotation angle of universal joint in $\mathbb{R}^3$
ϕ	The 'y' axis Euler rotation angle of universal joint in $\mathbb{R}^3$
$\mathbf{E}$	Resultant Euler angles transformation matrix
$\mathbf{A}$	Direct kinematics/Forward Jacobian matrix
$\mathbf{B}$	Inverse kinematics/Inverse Jacobian matrix
$\mathcal{D}_p$	Determinant value of the direct kinematics matrix for the 3-S$\underline{\text{P}}$S-U mechanism
$\mathcal{D}_s$	Determinant value of the inverse kinematics matrix for the 3-S$\underline{\text{P}}$S-U mechanism
$\mathcal{G}_i$	Joint limits/Constraint equation of the i^{th} joint for the 3-S$\underline{\text{P}}$S-U mechanism
$\mathcal{J}_p$	Determinant value of the direct kinematics matrix for the 4-S$\underline{\text{P}}$S-U mechanism
$\mathcal{J}_s$	Determinant value of the inverse kinematics matrix for the 4-S$\underline{\text{P}}$S-U mechanism
$\mathcal{H}_i$	Joint limits/Constraint equation of the i^{th} joint for the 4-S$\underline{\text{P}}$S-U mechanism
$\mathbf{H}$	Hessian matrix
Γ	Motor torque obtained after PID control
δ_k	Rotation angle of k^{th} module of the robot
fun	Objective function
g_k	Inequality constraint for k^{th} module
h_k	Equality constraint for k^{th} module

Introduction

Motivation

Robots have been into existence since the late 20th century for engineering and industrial applications. It was the manufacturing industries, which first started using simple robotic arms for performing repetitive and simple tasks. Since then, technological advances have allowed robots to be more versatile and flexible. At present, in the automotive industry, a single production line can work on several models of vehicles. With advancements in technology, mobile robots started to appear for exploration and interaction with an environment. Robots also find their applications for military and civil rescue operation in the form of observation and combat drones. Robots have also played an important role in the field of medicine by making procedures safer and less expensive for patients. They can perform surgery with better accuracy and can access through tiny places. Recently, piping inspection robots have started to appear and they have created a major attraction in the scientific research community. These robots can carry out an inspection or an intervention on areas that are difficult to access by humans. They also reduce the chances of hazards or risks to human life as inspection robots can potentially work within an irradiated or a polluted environment. With the automation of control systems, inspection robots can accomplish the desired task within a short period as well as with better accuracy. Generally, the locomotion principles of these robots can employ either mechanical systems or inspire their motion from nature, otherwise referred to as Bio-inspired techniques. This doctoral thesis focuses on the design of a bio-inspired robot for the inspection of pipelines. The operational context and the general problems associated with the design of a piping inspection is presented at the beginning. The thesis is then subdivided into multiple chapters for arriving at a design that can be suitable for the operational context.

Operational context and general issues

A piping inspection robot is usually subject to many objectives depending on the application and requirement of the industry. In this section, the operational context of the piping inspection robot for the thesis will be defined. Inspection robots for pipelines can operate for smaller diameter ranges of 40–50 mm to bigger diameter ranges of 500–600 mm. In reality, inside an industrial pipeline, inspection robots can encounter various kinds of profiles such as bends,

intersections or even deposits as shown below in Figure 1.

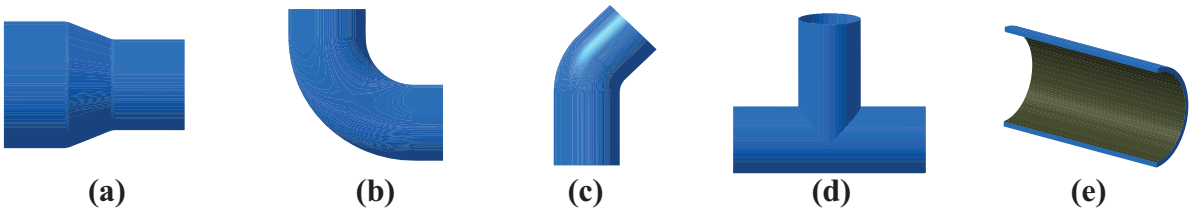


Figure 1 – Possible profiles encountered by an inspection robot inside an industrial pipeline: (a) varying diameter; (b) curvature; (c) inclination; (d) branching; and (e) uneven surface walls [PHC+11]

The design of inspection robots for pipeline diameters less than 150 mm is a big challenge. In nuclear industries that employ Pressurized Heavy Water Reactors (PHWR), corrosion is a common issue [Tap12] and the feed water pipelines have diameters between 55–75 mm [CPJ09]. In this thesis, a focus will be kept on arriving at the design of a piping inspection robot that can operate in a pipeline having a diameter of 150 mm or less and at the same time, it must be able to overcome bends or intersections (Figures 1b to 1d). Also, the robot must be capable of adapting its structure when it encounters a diameter change inside a pipeline (Figure 1a). The classical architecture of a piping inspection robot comprises: a head, a platform and cables. The functions of these components are given by:

- **Head**: This is the main part of the robot and it usually houses equipment such as Non-Destructive Testing (NDT) tools, cleaning or cutting tool or a Tungsten Inert Gas (TIG) weld torch. These equipments are chosen based on the application.
- **Platform**: The robot platform houses all the tools, manages the cables and accommodates the robot head. The platform withstands all the reaction forces generated under operation modes.
- **Umbilicus/Cables**: This is the key feature of a robot and they are responsible for transferring information between the point of supervision and the robot. They are normally of retracting type, which is essential for the withdrawal of the robot outside the pipeline in case of emergencies.

The inspection and intervention phases require specific tools, so different types of heads will be accommodated according to the operation, unlike the platform, which is common to all the operations. Generally, piping inspection robots are encountered with some issues. The general problems associated with a piping inspection robot are given by:

1. **Movement**: This problem is concerned with the difficulties in moving the robot inside a pipe. One of the major problem associated during movement will be to pull or push the payload as the robot tries to pass through or overcome an obstacle. The various kinds of obstacles that can be encountered within a pipeline were studied by Park et al.[PHC+11], which includes bends, elbows, T-unions, diameter variations and presence of deposits on

the pipeline walls. Another problem associated with the movement is the entanglement of robot cables while passing through multiple bends from the initial point of locomotion [HOMS99].

2. **Positioning**: It is very important to position the robot accurately on the zone to be inspected or intervened. It is necessary to take into account the tolerance of the design data and there are chances that there exists a mismatch between reference and target positions caused by data which are not updated.
3. **Inspection**: When a robot performs an inspection using NDT tools the precision is very important while inspecting a surface taking into consideration the plating force on pipelines and the orientation of the probe. Also, in case of ultrasounds, which makes use of gels for passage inside the pipe, there are chances that the probes might transfer irrelevant information owing to the nature of the deposits. When inspection cameras are employed, a special lighting system is required to highlight cracks to avoid glares.
4. **Intervention**: During an intervention, the robot must be capable of satisfying various factors. For example, when a cleaning operation is performed, there should be a suitable storage facility mounted along with the robot to collect the deposits. If the robot is used for heavy operations such as drilling or welding, the robot platform as well as the head must be capable of withstanding the vibrations and other mechanical forces generated by the equipment.

This doctoral thesis will address the first issue, which is associated with the movement of robot inside a pipeline having varying diameters and bends. The thesis has been subdivided into five chapters and a problem statement is defined and addressed in each chapter. An overview of the work carried out in each chapter are explained in the following section.

Thesis contributions

1. Literature review and mathematical concepts

The first chapter of the doctoral thesis focuses on the literature review of existing researches in the field of inspection robots and description of some important mathematical tools. The first half of this chapter presents the state of the art of locomotion techniques of robots based on Mechanical type and Bio-inspired type systems. A literature review on the articulation units which can permit a robot to pass through pipe bends or junctions is then carried out. A synthesis table is presented for the locomotion techniques as well as for the articulation units to identify solutions that are compatible for the operational context. The second half of this chapter focuses mainly on the definitions of certain mathematical concepts that are employed in the thesis chapters. Some software tools namely: the SIROPA library of Maple and some optimization functions of MATLAB software which were employed for analysis in the thesis chapters are also presented.

2. Rigid bio-inspired piping inspection robot

The second chapter of the doctoral thesis is mainly associated with solving the following problem statements:

1. *What are the different ways for an inspection robot to accomplish locomotion inside a pipeline?*
2. *What are the forces required to ensure good contact with the walls of a pipeline having various orientations and diameters?*

In-line with the operational context as well as from the synthesis table of locomotion techniques in the literature review, the second chapter of the doctoral thesis begins with the study of a bio-inspired piping inspection robot developed at LS2N, France. As a part of project with AREVA, a caterpillar type inspection robot for nuclear pipelines was designed and proposed by Henry et al.[HCP+14]. This robot accomplishes the locomotion of a caterpillar in six steps by using slot-follower leg mechanisms, DC-Motors and spindle drives. The robot is capable of working inside 40–94 mm straight pipelines. However, the force models associated with the robot were not analyzed. The first half of Chapter-2 focuses on the development of static and dynamic force models for this piping inspection robot. Using Coulomb's law of dry friction, the static force model is developed and solved to interpret the actuator forces of the leg mechanisms during clamping phases of the robot. The dynamic force model is then developed using the recursive Newton-Euler algorithm [Kha10] for interpreting the actuator forces on clamping modules and the elongation module during locomotion. The second half of Chapter-2 focuses on the experimental validation of the prototype of the inspection robot. Using Maxon DC-motors, spindle drives, standard fasteners and the slot-follower leg mechanism, the bio-inspired piping inspection robot is realized at LS2N. With the help of a BeagleBone black microcomputer and servo-controllers, a force control algorithm is developed for the control of this piping inspection robot. The peak results coming from the static force model are set as threshold limits for attaining static/clamping phases during the locomotion cycle. The real time actuator forces are interpreted for the robot by performing experiments inside a 74 mm diameter test pipeline.

3. Design and analysis of Tensegrity mechanisms

The third chapter of the doctoral thesis addresses the following problem statements:

1. *What are the key design issues that must be addressed for the design of an articulation unit which can facilitate a robot to overcome pipe bends and junctions under passive and active modes?*
2. *What are the tilt limits for the mechanism?*
3. *What are the optimal design parameters of the mechanism for having a stable configuration under static modes?*

The second chapter of the thesis mainly focused on the locomotion principle and the estimation of contact forces with the pipeline walls for an inspection robot developed at LS2N. However, this prototype is a rigid model and its applications are limited to straight pipeline profiles having varying diameters. In reality, a piping inspection robot can encounter various pipe profiles such as 90° bends, T-unions, or junctions. Chapter-3 focuses on the design modification of the rigid bio-inspired piping inspection robot into a flexible system by the incorporation of articulation units in the existing architecture. This chapter initially addresses the general problems associated with the design of an articulation unit viz: Passive compliance, Active compliance and Tilt limits. By taking these factors into account, a tensegrity mechanism that employs a passive universal joint and three tension springs is proposed as an articulation unit for the bio-inspired robot. This mechanism can operate passively while encountering 90° bends and in the event of a junction, cables that pass through the springs of the mechanism can be actuated to tilt along a certain direction. The tensegrity mechanism is correlated to a parallel manipulator of type 3-S$\underline{P}$S-U for estimating the geometrical equations. The analysis of the positioning of the end-effector is carried out in two ways. The first analysis employs the theory of Tilt & Torsion (T&T) [BZG02] to estimate the tilt limits of the mechanism by decomposition of the architecture into three case study postures. Using the Cylindrical Algebraic Decomposition (CAD) algorithm and Groebner bases elimination technique, the tilt limits for the three case study postures with respect to the joint limits are identified using the SIROPA library of Maple [JCB+18, CMRW20]. The posture with the maximum tilt limits is identified and stability analysis with respect to the T&T angles is carried out for the mechanism under static modes. The second analysis focuses on the study of singularities of the tensegrity mechanism which is carried out for the Euler angles of the universal joint. The singularity analysis helps in determining the presence of parallel and serial singularities, based on which the workspaces of the mechanism can be identified to understand the tilt limits. By the addition of another tension spring, the mechanism is converted into a 4-S$\underline{P}$S-U architecture. The singularity analysis is performed again for the modified architecture and a comparison is being made between both architectures based on tilt limits. The comparison helps in determining the architecture's ability to address the issues of Passive compliance, Active compliance, or both.

The final sections of this chapter presents the stability analysis under static modes for the Euler angles of the universal joint. By following a heuristic optimization approach in MATLAB, the optimal design parameters for the 3-S$\underline{P}$S-U and 4-S$\underline{P}$S-U tensegrity mechanism under the presence of external forces are determined. The singularity analysis is revisited for the modified design parameters for understanding and comparing the tilt limits of both architectures. Prototypes of the 3-S$\underline{P}$S-U tensegrity mechanism are realized for initial and optimal design parameters for demonstrating the stability under static modes.

4. Experimental validation of the 3-S$\underline{P}$S-U tensegrity mechanism

The fourth chapter of the doctoral thesis addresses the following problem statements:

1. *What are the possible ways for actuating a 3-SPS-U tensegrity mechanism?*
2. *What are the governing equations for achieving the the desired trajectory and the desired positions by the mechanism?*

Chapter-4 of the doctoral thesis focuses on the experimental validation on the prototype of the 3-SPS-U tensegrity mechanism that was presented at the end of Chapter-3. The main purpose of this chapter is to understand how the issue of active compliance could be addressed through the actuation of cables used in the mechanism. Using DC-Motors, planetary gearbox, pulleys and cables, the actuation of the springs of the tensegrity mechanism is performed. Similar to Chapter-2, a force control algorithm is incorporated using a BeagleBone black microcomputer for controlling the tilt limits of the mechanism. Using a closed-loop PID controller, the mechanism is actuated to tilt along a certain direction within the singularity free workspace with minimal errors between the measured and desired position data. Two sets of experiments viz: a linear trajectory and a circular trajectory [CCLW15] is performed where the Euler angles are passed as inputs to the controller with the help of which the Inverse Kinematic Problem (IKP) is solved using the Jacobian matrix and its derivative. The two sets of experiments are carried out for two orientations, namely: Vertical and Horizontal and they correspond to the orientations of the bio-inspired robot of Chapter-2 inside a vertical or a horizontal pipeline. These orientations help in identifying the influences of gravity and the self-weight of the mechanism on the motor torques.

5. An optimal design of the flexible piping inspection robot

The fifth chapter of the thesis addresses the following problem statements:

1. *What is the optimum size required for the modules of a piping inspection robot to overcome a bend at 90° ?*
2. *What are the optimum sizes for the clamping modules that could be accommodated for a piping inspection robot to work inside straight and curved pipe profiles?*

By the implementation of a tensegrity mechanism into the bio-inspired robot studied in Chapter-2, a flexible system was obtained. However, if the motor units and spindle drives used in the existing prototype are carried over to the flexible robot, the assembly proves to be oversized and it cannot overcome a pipe bend, especially at 90°. This chapter thus focuses on the determination of motor sizing by an optimization approach. Two optimization problems are defined and solved by a deterministic approach in MATLAB. Both problems are carried out inside a planar test bench that consists of a pipeline with a 90° bend, which is constructed by discretization equations in MATLAB. The first optimization problem aims to determine the size of the motor-spindle drive for each module of the caterpillar type inspection robot. The robot modules are considered as planar multi-body blocks within the test bench for simulation. The first optimization is carried out for pipeline diameters starting from 70 mm to 160 mm. From the

results obtained, an optimal motor sizing is determined for the modules based on dimensions, velocity and feed force factors from catalogue products of Maxon [Max]. The first optimization problem also helps in identifying the working diameter range for the flexible robot. The second optimization problem is then performed which aims to determine the size of the slot-follower leg mechanism that could be accommodated for the results of the first optimization problem. This optimization problem is performed for the working diameter range identified from the first problem and the dimensions of the leg mechanism are determined by a geometric approach. From the results of both optimization problems, the entire robot is digitally realized in CAD software by assembling DC-Motors, spindle drives, standard fasteners, tensegrity mechanisms and slot-follower leg mechanisms. The flexible robot permits to have an “Elephant trunk” type model.

CHAPTER 1

Literature review and mathematical concepts

This chapter is the foundation for the doctoral thesis. The first half of this chapter analyzes in detail the existing researches in the field of piping inspection robots, their locomotion principles and their actuation strategies. Also, some of the articulation units existing in the literature are presented. The second half of this chapter discusses the various mathematical concepts and some software tools that were employed in the thesis chapters.

1.1 Introduction

In the present world, with the rapid growth of industries such as nuclear, chemical, oil & gas and sewage, the need for piping inspection robots has become the topmost priority. Failures of pipelines in such industries lead to major consequences. Manual intervention of these pipelines for performing activities such as inspection, welding and corrosion detection is not preferable as they not only lead to the loss of human life but also poses other threats such as long term radiation effects and serious injuries. These pipelines can also be termed as Hazardous pipelines as they operate to transfer dangerous fluids or substances for larger distances from the installation premises [Pap99]. Frequent intervention or maintenance are required for such pipelines in order to avoid severe consequences to the environment as well as to the people. Annually, the number of incidents and fatalities related to gas pipelines are analyzed by the European Gas pipeline Incident data Group (EGIG) in Europe and Pipeline and Hazardous Materials Safety Administration (PHMSA) in the United-States. The statistics of the incidents reported in the last ten years by the PHMSA is represented in Figure 1.1.

A total of 731 incidents, 280 fatalities and 1183 injuries have been reported as per the

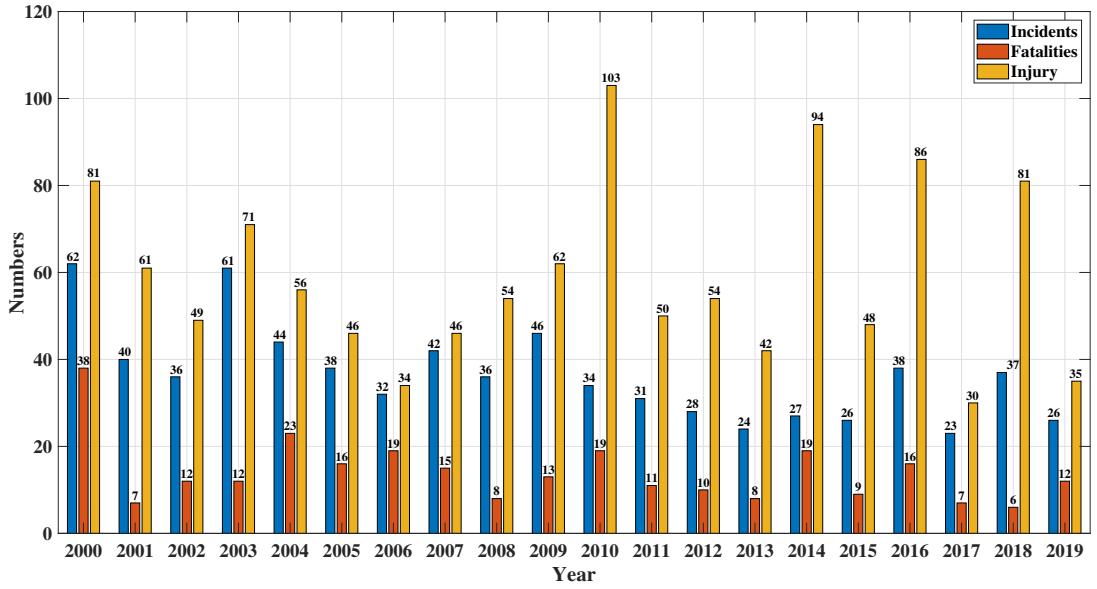


Figure 1.1 – Pipeline incidents statistics issued by PHMSA [PHM]

statistics of PHMSA over the span of last ten years in gas transmission pipelines. Incidents normally included corrosion issues, breakage of pipes and fire explosions. The total number of fatalities and injuries prove to be a concern and this figures appear more larger when the entire piping incidents connected with all hazardous industries are studied across the globe. This concern proves the importance of inspection robots, which can potentially reduce the hazards caused to the workers of these industries to a greater extent. Through automation of the robot control system, it is possible to carry out intervention without affecting the operations of the installations.

In the following section, a literature review of existing piping inspection robots is carried out based on the locomotion principles, their actuation strategies and associated articulation units. In the subsequent section, some important mathematical concepts employed in the thesis chapters are presented. Followed by that, some mathematical tools such as the SIROPA library of Maple and the optimization tools of MATLAB software are presented.

1.2 Locomotion techniques of robots- A literature review

This sections mainly focuses on the various locomotion technique of robots in literature and their actuation strategies. The study will try to focus on the robots that have inspection applications as well as some medical applications such as colonoscopy. A general classification of in-pipe inspection robots was proposed by Roh et al.[RC05] and it is represented in Figure 1.2. The classification proposed by Roh et al.[RC05] gives an overview of possible locomotions techniques for robots to move inside pipelines. A further classification can also be done based on the type

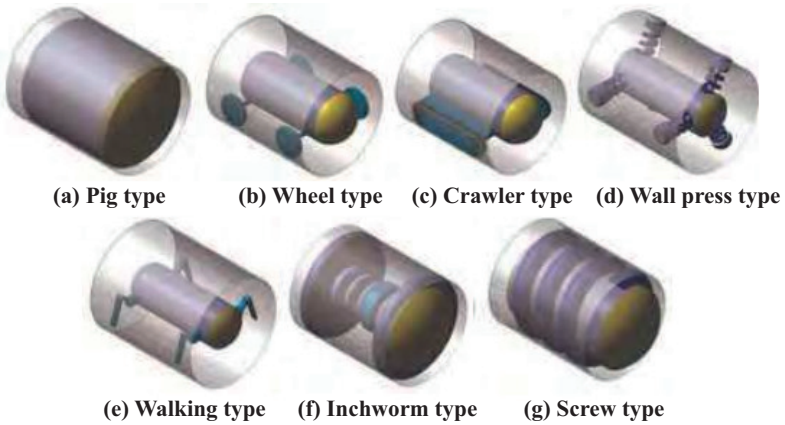


Figure 1.2 – The classification of in-pipe inspection robots by Roh et al.[RC05]

of locomotion technique viz: Mechanical and Bio-inspired solutions. Kassim et al.[KPN+06] proposed a distinction between these two categories for the application of robots in the field of colonoscopy. A literature review will be carried out based on the classification proposed by Kassim et el.[KPN+06].

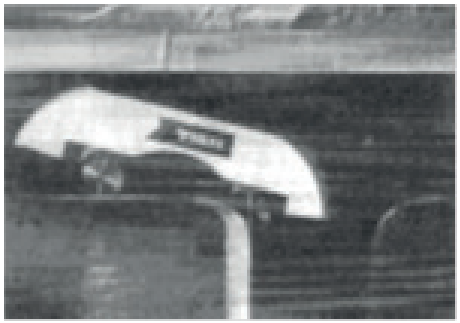
1.2.1 Mechanical type locomotion systems

Mechanical type locomotions techniques could be imagined for robots that employ kinematic architectures such as wheels, pulleys etc., to accomplish movements within a certain space or environment. The various contributions for this type of locomotion technique will be explored in this section.

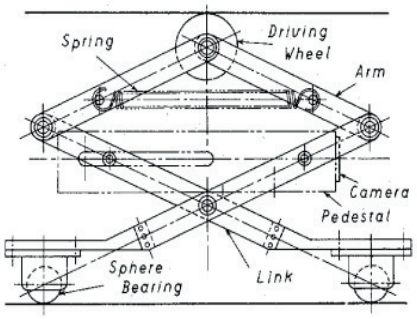
Wheel type system

Wheel type locomotion systems are widely used in pipeline inspection. For working inside vertical passages, the robots required a specific architecture to maintain permanent contact between the wheel and the pipe regardless of the robot orientation. Normally, diameter changes in the pipe do not prove to be a hindrance for the wheel based systems. Most of the wheeled robots use a mechanical clamping system for better adhesion with pipeline walls. An alternate technique that used magnetic wheels for pipeline walls was proposed by Kawaguchi et al.[KYK+95]. This robot can work inside a 150 mm diameter pipeline having a bend. For vertical passages, the robotic system is centered using several branches which can vary from two to six. For the two branch robots, the MOGRER robot of Okada et al.[OS87] which uses one driving wheel and the robot PIRATE of Dertien et al.[DS11] which employs two driving wheels are some examples. These robots are suitable for pipeline diameters ranging from 40 to 125 mm and they employ DC-motors with gear-reduction units for actuation. For the 3-branch system, the MRINSPECT robot of Roh et al.[RRYC01]is a good example. The main advantage of the MRINSPECT series is their strong adaptation to pipeline diameters with articulation units behaving either passive

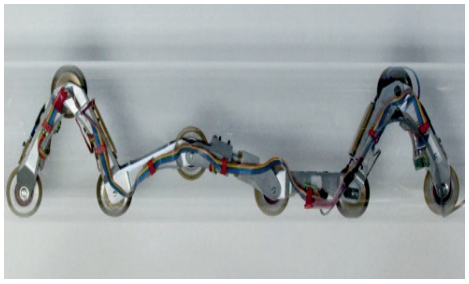
(MRINSPECT-01 , MRINSPECT-02) or active (MRINSPECT-04). All these wheel based systems employ either a DC-motor or a DC-motor coupled with gear-reduction unit for actuation. The wheel type systems generally do not occupy more space but their ability to adapt to change to diameter changes might be limited. A representation of the wheel based systems studied in this section is shown below in Figure 1.3.



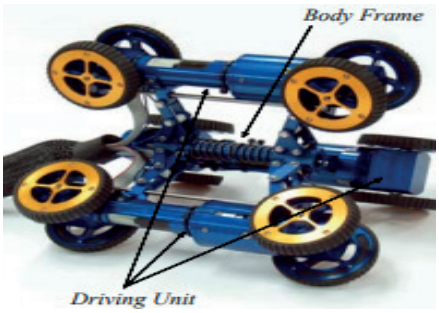
(a)



(b)



(c)



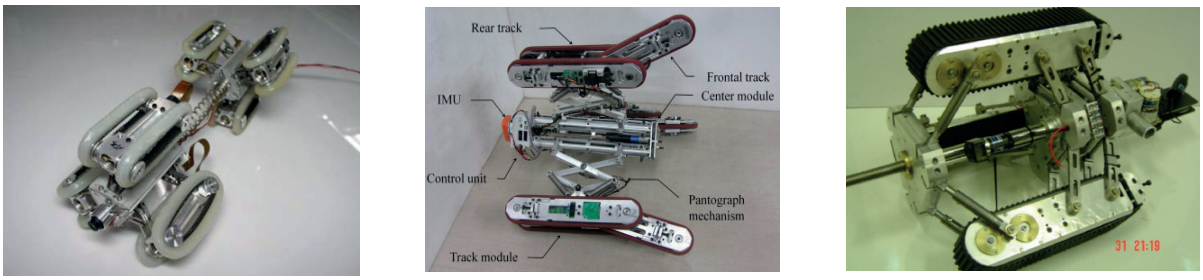
(d)

Figure 1.3 – Some existing wheel type systems for piping inspection robots: (a) Magnetic wheel type [KYK+95], (b) MOGRER robot [OS87] , (c) PIRATE robot [DS11] and (d) MRINSPECT-04 robot [RRYC01]

Track type system

Track based systems as the name indicates employ a conveyor based system to have contact with pipeline walls. These systems occupy enormous space over the wheel type systems. The main advantage of this system is that they ensure good adhesion, especially while crossing obstacles. On the other hand, they prove to be more voluminous. The robots employing this system are highly suitable for pipelines having larger diameters. These systems can adapt well according to the diameter change but they are hardly minute for the operational context. Some existing researches that employ the track systems are the robot of Kwon et al.[KLJY08] which comprises two motorized units for operation inside 100 mm diameter pipeline and the PAROYS-II robot proposed by Park et al.[PHC+11], which is equipped with two-track systems

for pipeline diameters ranging between 400–700 mm. Another example is the robot of Moghaddam et al.[MAH11], which also employs a track based system for 300 mm diameter pipelines. A representation of the track systems studied in this section is shown below in Figure 1.4.



(a) **(b)** **(c)**

Figure 1.4 – Some existing track type systems for piping inspection robots: (a) Robot of Kwon et al.[KLJY08], (b) the PAROYS-II robot [PHC+11] and (c) Robot of Moghaddam et al.[MAH11]

Helical & Screw-type systems

The helical and screw-type locomotion systems are more or less similar where the desired movement is achieved through translation over a given pitch. Helical systems employ kinematics derived from wheeled systems. An interesting example is the HELI-PIPE robot of Horodinca et al.[HPB03, HDMP02], which has a rotating head that performs a propeller-shaped movement by actuation of a DC-motor with a gear reduction unit. When the system rotates, the wheels have a helix angle that creates a helical path and makes the robot progress longitudinally in the tube. The main problem associated with the helical system is that a variation in the diameter of the pipe is difficult to manage. The HELI-PIPE robot comes in different versions with different rotary heads, which are compatible for a certain range of diameters. Screw-type systems, on the other hand, operate similar to the helical systems. Normally, a screw mechanism is actuated by a DC-motor to have a screw motion yielding to one degree of freedom. An example of a robot that employs this system is the robot of Nayak et al.[NP14]. Using a central actuated motor, a screw motion is generated to a wheel system which ensures contact with pipeline walls at all instances of locomotion. This robot is limited to diameter ranges of 127–152 mm. The helical and screw systems prove to be an interesting candidate for locomotion but their adaptability to diameter changes and overcoming elbows might be an issue. A representation of the helical and screw-type systems studied in this section is shown in Figure 1.5.

Flexible feet system

These kind of systems are generally suitable for pipelines having very small diameters of less than 10 mm. An example is the Microrobot of Nishikawa et al.[NSS+99], which is shown in Figure 1.5a. This robot uses a piezoelectric biomorph as an actuator. The robot can circulate in a vertical pipe with the technique of “Stick and slip” where the rapid movement of the biomorphs

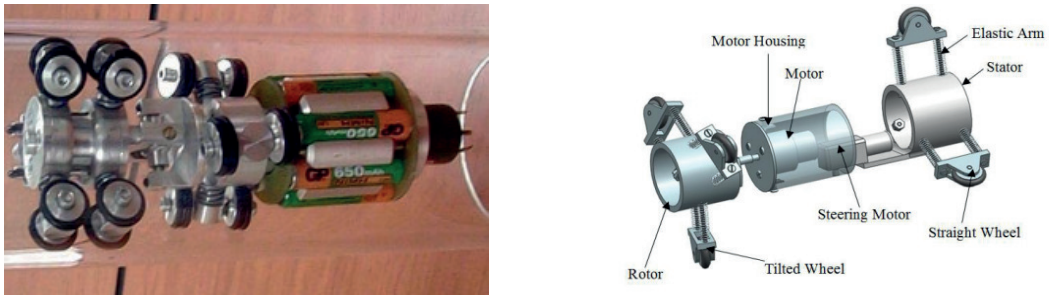


Figure 1.5 – Representations of the robots using (a)Helical type [HPB03] and (b) Screw-type [NP14] systems

advances the robot by one step. The power source for this robot is transmitted with the help of waves (Figure 1.5b). The main disadvantage of this kind of system is that they require a surface without deposits of pollutants such as oil, dust etc. Also, the final application of these robots is limited to visual inspection.

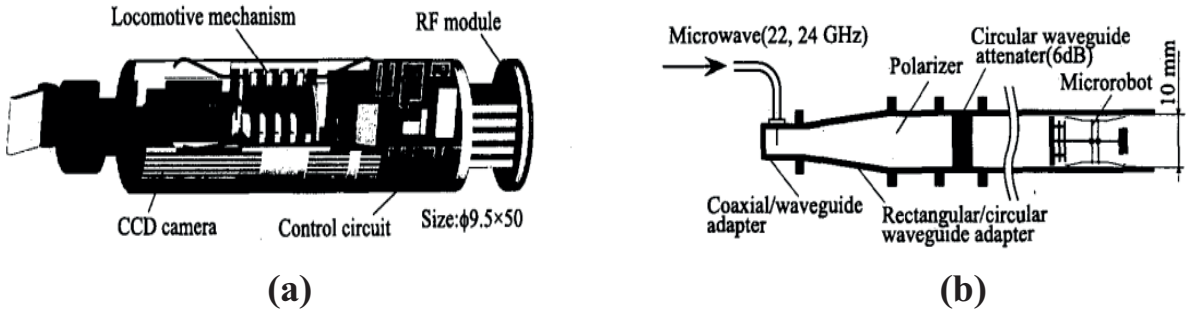


Figure 1.6 – The flexible feet robot of Nishikawa et al.[NSS+99]

1.2.2 Bio-inspired locomotion systems

Bio-inspired locomotion systems are purely based on the application of animal locomotions to general engineering problems. There are several axes of development in bio-inspiration to name a few: the method of locomotion, the techniques of actuation and the management of contacts. Some of the researches in the field of robotics that inspires locomotion from animals are explored in this section.

Worm type system

The worm type locomotion can be imagined in two types viz: Caterpillar and Earthworm type, with both being differentiated by the contact established with the ground. The principle of locomotion of the caterpillar, also called Inchworm is based on a step by step displacement. Basic caterpillar-type locomotion requires three different actions: two blocking functions and an

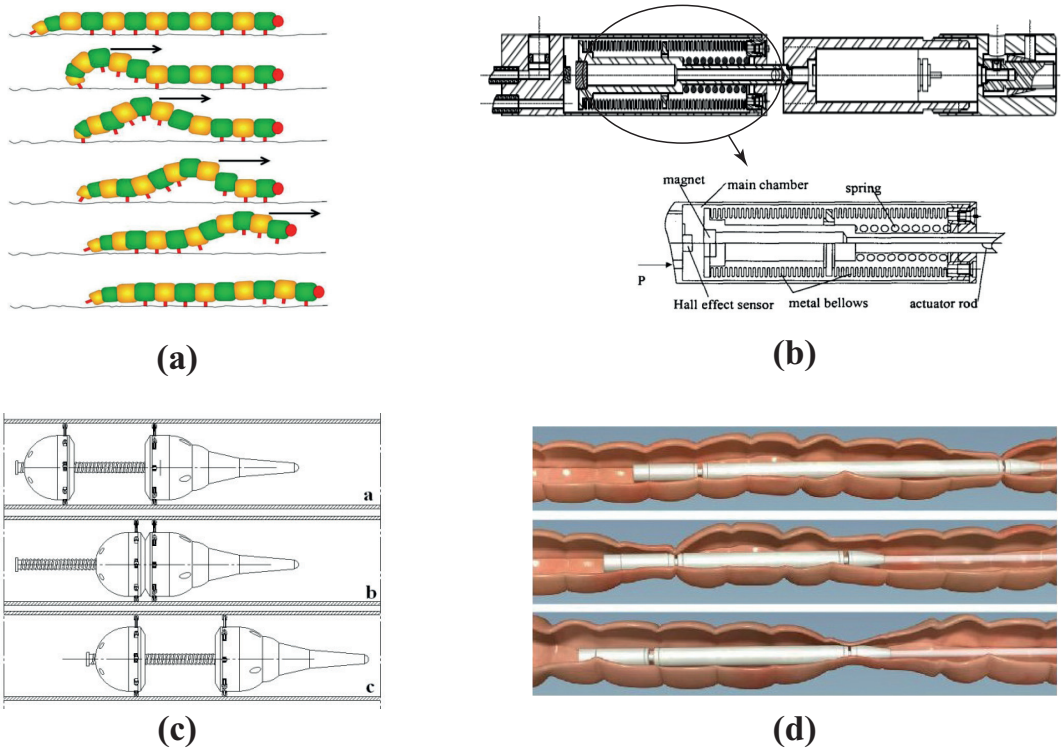
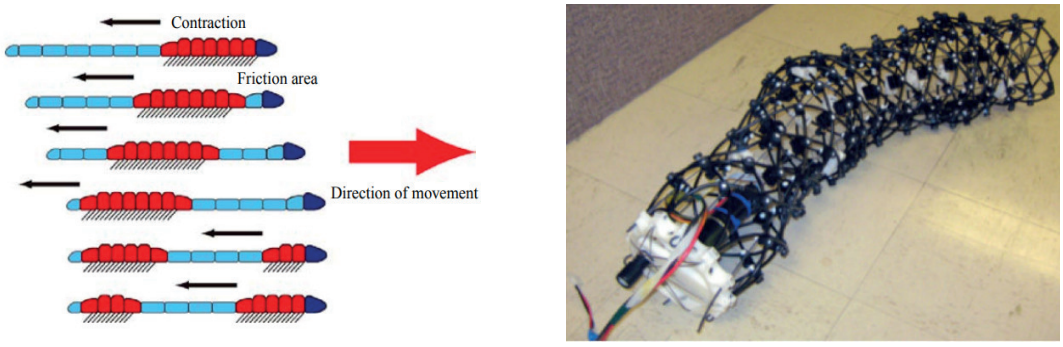


Figure 1.7 – Representation of the: (a) Caterpillar locomotion [SPS14], (b) Robot of Anthierens et al.[ALT+00], (c) Robot of Zhang et al.[ZZSJ10] and (d) Robot for colonoscopy [BIDM+13]

elongation function (Figure 1.7a). The actuation methods vary from jacks to electric motors, including Shape Memory Alloys (SMA). An interesting example is the robot of Anthierens et al.[ACB99, ALT+00], which combines pneumatic bellows for elongation and an electric motor for blocking (Figure 1.7b). This robot was developed in 1999 for steam generators of nuclear power stations. The major limitations of this locomotion type are low forward speed and difficulties in passing obstacles such as intersections, bends and management of diameter changes. In order to simplify the passage of obstacles, Zhang et al.[ZZSJ10] proposed a robot with a passive central axis having a pilot head that uses system compliance to overcome elbows and intersections (Figure 1.7c). This robot uses a DC-motor with a screw and flexible helical axle to move inside pipelines having straight and bent profiles. The caterpillar type locomotion technique generally finds its applications in medical fields such as colonoscopy. The company Endotics, offers a worm type robot that is used for performing colonoscopy [BIDM+13] (Figure 1.7d). This robot uses a joypad controller with which the operator can control the movement and direction of the robot through the intestines. Earthworm type locomotion system, on the other hand, is close to that of the caterpillar, except with the difference that the point of contact with the ground slides along the body of the animal (Figure 1.8a). The advantage of this locomotion technique is a continuous monodirectional advance speed. Boxerbaum et al.[BHS+11] presented and validated this locomotion technique on a prototype (Figure 1.8b). Another interesting earthworm type



(a) **(b)**

Figure 1.8 – Representation of the: (a) Earthworm locomotion [WONY09] and (b) Robot of Boxerbaum et al.[BHS+11]

robot was proposed by Liu et al.[LMS+06]. This robot employs an SMA actuated spring to mimic the locomotion of an earthworm. However, both these earthworm type robots were studied in an open environment. The locomotion principle of an earthworm is interesting and useful for adapting to diameter changes when compared to the caterpillar locomotion. However, the payload carrying capacity for the caterpillar system is better when compared to the earthworm.

Snake type system

Robots that move like snakes mimic their movements from biological snakes. Snakes have four basic types of motion according to the environment viz: Concertina movement, Serpentine or lateral undulation, Rectilinear crawling and a Sidewinding [Pra11]. The representation of these four types is shown in Figure 1.9a. Most of the latest robots imitate the serpentine, caterpillar and sidewinding modes. The concertina is not very much present in literature and it is quite similar to the elongation and blocking phases of inchworm type robots. The first serpentine type locomotion robot was proposed by Hirose et al.[HM04], which is called the Active Cord Mechanism model (ACM-III) (Figure 1.9b). This robot uses a simple pivot between each module and moves along surfaces that are uneven or wind along paths. The ACM robots mainly find its application in disaster management where the robot can try to perform rescue operations. Another interesting application of the serpentine locomotion is the snake-like AmphiBot-III robot of Porez et al.[PBI14]. This robot also uses several pivots between modules and it is employed for underwater inspection application (Figure 1.9c). For the caterpillar or rectilinear motion, the SCALYBOT robot of Marvi et al.[MMRH11] is a good example. This robot uses a deformable parallelogram system to accomplish the locomotion using a Sliding Mode Control (SMC) hydraulic actuator. The sidewinding locomotion can be classified as a 3D movement and it requires at least 2 degrees of freedom between each module which allows for complex movements. An example is the robot of Rollinson et al.[RBC12], wherein the sidewinding locomotion is validated (Figure 1.9d). Apart from that, this robot also performs pole climbing

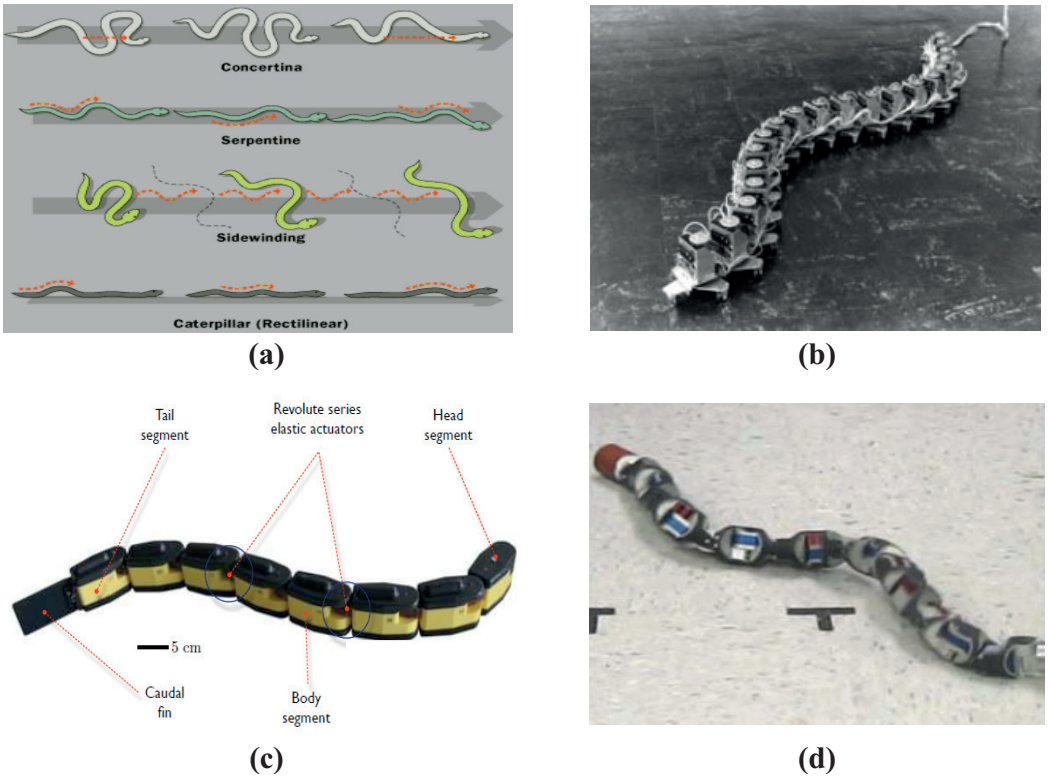


Figure 1.9 – Representation of the: (a) Snake locomotion types [Pra11], (b) ACM-III robot of Hirose et al.[HM04] , (c) AmphiBot-III snake like robot [PBI14] and (d) robot of Rollinson et al.[RBC12]

and rolling through cyclic motions. With reference to the operational context of the robot, serpent type robots have complex controls as they are hyper-redundant architectures. However, their inter-modular linkages can be of greater interests.

Legged type system

The idea of reproducing the approach of animals with legs dates back to the invention of Leonardo da Vinci, who synthesized the first anthropomorphic mechanism [STM07]. In the field of robotics, the general system comprises spider-type or multiple legged type robots in the literature. Neubauer et al.[Neu94] proposed an alternative to classical locomotion systems in a pipeline where the robot resembles the model of a spider. However, this work remained at the theoretical level but it was able to outline the issues of support points and passages over obstacles. This allowed Rimon et al.[RSS01] to develop a four-legged planar robot (Figure 1.10a). This robot is capable of moving inside two-dimensional tunnels having complicated profiles through the actuation of a simple DC-servo motor. Gasparetto et al.[GVS08] proposed the modeling of a spider leg for simulating a digital model for any environment (Figure 1.10b). Apart from the spider type leg systems, robots having multiple legs that are articulated are also present

in literature. Quirini et al.[QWMD07] and Valdastri et al.[VWIQ+09] proposed a 12 legged articulated robot which is used for performing colonoscopy. This robot employs a slot-follower mechanism driven by a screw to actuate the legs and it operates for tube diameters ranging between 11–25 mm. The numerical model of this robot is shown in Figure 1.10c. Another interesting robot that was developed for piping inspection is the MORITZ robot (Figure 1.10d) of Zagler et al.[ZP03]. This robot uses articulated leg mechanisms to work inside pipelines having straight and bent profiles with diameters between 600–700 mm. However, this robot appears to be huge and deviates from the operational context. Robots with leg mechanisms are an interesting choice for having better contacts with pipeline walls during locomotion. Their adaptability to diameter changes in pipelines might be a concern.

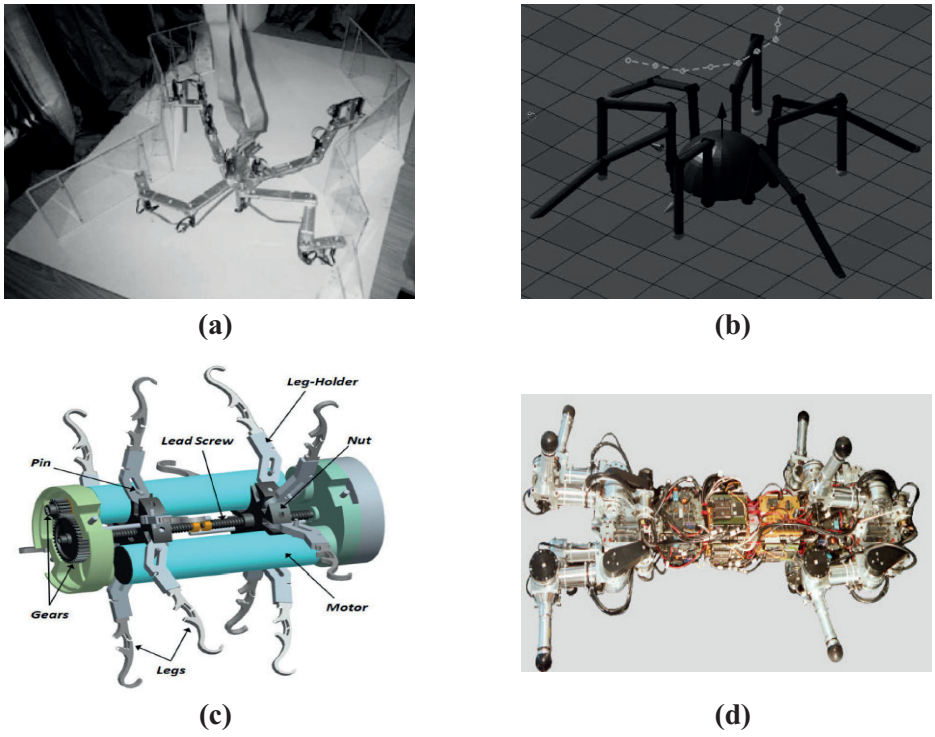
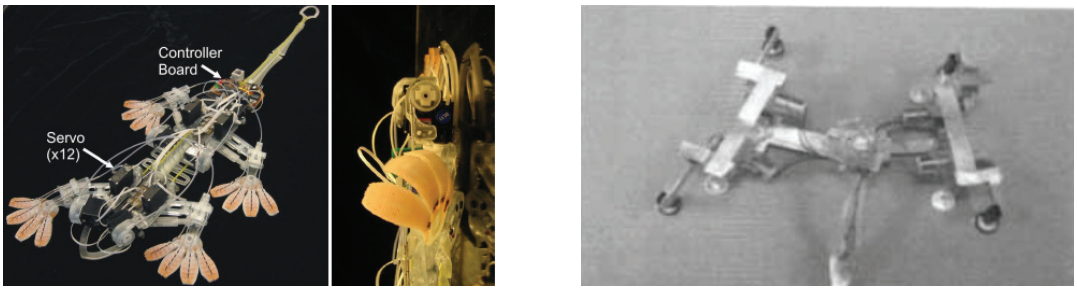


Figure 1.10 – Representation of the: (a) Robot of Rimon et al.[RSS01], (b) Robot of Gasparetto et al.[GVS08] , (c) Robot of Valdastri et al.[VWIQ+09] and (d) MORITZ robot [ZP03]

Gecko type system

Gecko refers to the family of small lizards. Their locomotion principle finds greater interest in the field of robotics, thanks to their contact criteria to a surface as well as rapid displacements and climbing. Usually, the legs are covered with microscopic spikes, which exploits the strengths of the van der Waals forces [SNK+05]. Stickybot robot of Kim et al.[KST+07] is an interesting example that inspires movement from gecko's to climb smooth vertical surfaces such as glasses and tiles (Figure 1.11a). This robot employs an anisotropic adhesive to have good contact with the surfaces and uses a cable drive mechanism through servo motor for locomotion. Another

interesting example is the robot of Carlo et al.[CM06], which inspires the gecko locomotion for climbing (Figure 1.11b). This robot uses Silly Putty for adhesion and SMA actuators for movement. The locomotion principle of gecko is interesting to the operational context but the contact criteria will be a big problem as sticky surfaces cannot assure good adhesion with pipeline walls having deposits or irregularities.



(a) **(b)**

Figure 1.11 – Representation of: (a) the Stickybot robot [KST+07] and (b) the robot of Carlo et al.[CM06]

1.2.3 Synthesis: Locomotion techniques

From the list of mechanical and bio-inspired locomotion techniques that were studied in this section, a synthesis is being done. This synthesis is carried out based on the following factors:

1. Sizing/Scaling: This factor determines if the mechanism can be sized/scaled down for the operational context.
2. Complexity: This factor determines how complex the mechanism can be for the operational context. In other words, it determines if a mechanism requires additional actuation units to be mounted on the robot during locomotion.
3. Diameter variation: This factor is essential to determine if the locomotion principle can adapt to sudden diameter changes without losing any contact with pipeline walls as well as for various pipeline orientations.
4. Contact with walls: This factor determines how effective the mechanism can adhere itself to the walls of the pipeline under any circumstances such as the presence of deposits or irregular pipeline walls.

These factors are identified with respect to the operational context in order to arrive at a design that can be suitable for pipeline diameters less than 150 mm with varying profiles. A scoring is given from 1 to 5, with 5 being the best. The synthesis for the locomotion techniques studied in this section is provided below in Table 1.1. The synthesis table gives an overview of the advantages and disadvantages of each locomotion technique based on scoring given from 1 to 5, with 5 being the best. With respect to the operational context, a suitable locomotion technique

Scoring from 1 to 5		Size & Scaling	Complexity	Diameter variations	Contact with walls
Mechanical systems	**Wheel type**	4	4	3	4
	Track type	2	3	4	3
	Helical/Screw-type	4	4	2	4
	Flexible feet type	4	3	3	1
Bio-inspired systems	**Worm type**	5	4	4	5
	Snake type	3	3	4	4
	Legged type	4	4	1	5
	Geko type	4	3	4	1

Table 1.1 – Scoring for the locomotion techniques studied in literature with respect to the operational context

can be chosen from Table 1.1. However, it is also possible to arrive at a technique that can combine one or more systems with reduced complexities, thereby exploiting the advantages of the employed techniques.

1.3 Articulation units for robots: A literature review

In this section, a literature review on the various types of articulation units employed in robots is being carried out. An articulation unit can be defined as a mechanism that permits to have a reconfigurable or a flexible system. In other words, the articulation unit is responsible for an assembly of rigid bodies to move to the desired position by adapting the structure of entire assembly in an environment. These units can be of simple mechanical elements such as springs, universal joints, flexible couplings, etc. These units can either be passive or active or a combination of both, depending on the application of the robot. For the operational context, an articulation unit for the design of a piping inspection robot is essential for passage through bends and intersections. The various passive and active articulation units in the field of robotic applications will be explored.

Universal joint

Universal joint or universal coupling is a mechanical element which is used for transmission of power or torque between shafts that intersect at a constantly varying angle [Jia19]. They find their applications in the transmission system of automobiles and also in the field of robotics. A classic example is their application to the Stewart-Gough platforms, where universal joints are employed in the base and end-effector platforms [BTT13]. These joints remain in the passive mode and adapt their positions according to the actuation of prismatic joints (Figure 1.12a). Apart from the Stewart-Gough platforms, universal joints also find their applications in other parallel manipulator architectures such as the 3-UPU. In the field of inspection robots, universal

joints play an essential role in overcoming bends or junctions. Usually, these joints are active and are actuated through DC-servo motors. An example is the robot of Lim et al.[LO09], which employs a universal joint actuated through a DC-motor (Figure 1.12b). This joint remains active throughout the locomotion cycle and can twist or turn the entire robot assembly while encountering bends. Another in-pipe inspection robot that employs a double actuated universal joint is the THES-I robot of Hirose et al.[HOMS99]. A central DC-motor controls the actuation of the universal joints to accomplish locomotion as well as to pass through bends or junctions (Figure 1.12c). Universal joints are a good solution for passage through pipe bends owing to their tilt and twist angles. Another possible advantage of these joints is that they can be potentially employed in passive modes without the need for any external actuation.

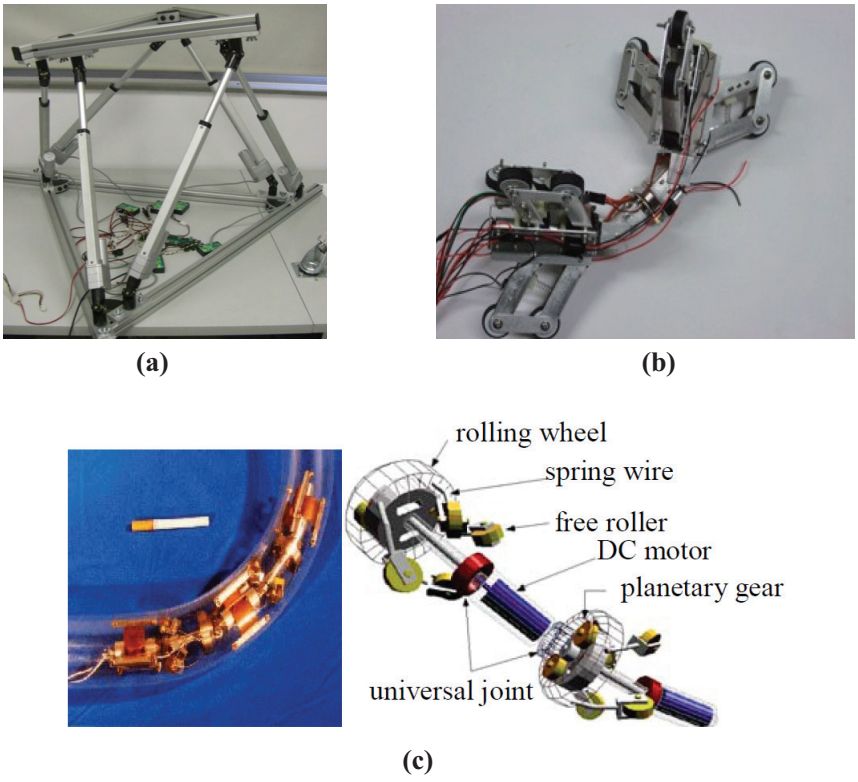
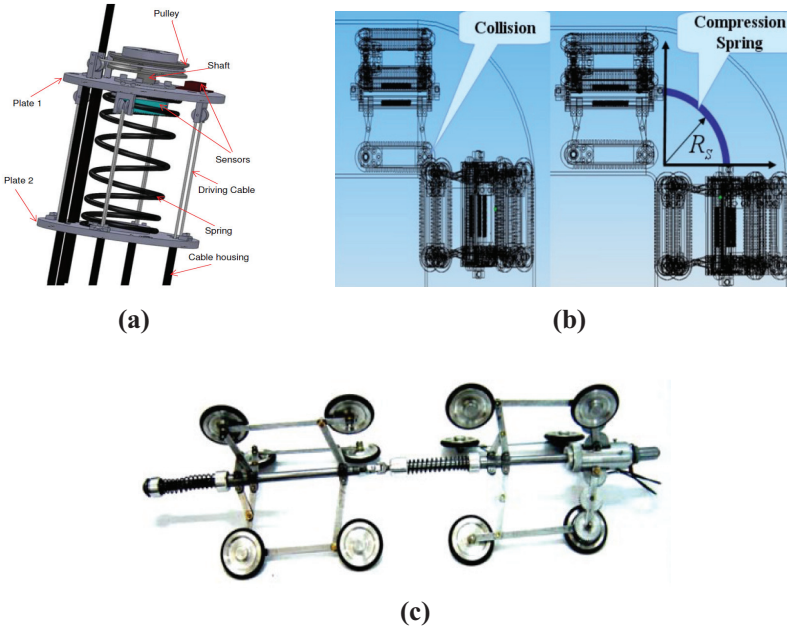


Figure 1.12 – Some applications of universal joints in robotics: (a) the Stewart-Gough platform [BTT13] , (b) the robot of Lim et al.[LO09] and (c) the THES-I robot of Hirose et al.[HOMS99]

Springs

Springs are simple machine elements which are resilient and can undergo a considerable amount of deflection without any permanent deformation [Jia19]. There are many types of springs such as Helical springs, leaf springs and Belleville springs. The most common type of springs that are found in the field of robotics is the Helical spring. Helical springs are wire-like springs and can be of extension, compression, or torsion type, depending on the application. An interesting example is the robot of Gao et al.[GXS+10], which is used to mimic the neck

motion of a human through a spring structure. The prototype employs a fixed base and a moving platform, both of which are assembled through a compression spring. Using cables and pulleys, the end-effector platform is actuated to obtain a 3-DOF neck motion and 1-DOF jaw movement (Figure 1.13a). The application of springs could also be found in piping inspection robots, where they help the entire robot to overcome passage or bends under passive modes. The robot of Kwon et al.[KLJY08] which was studied in the track based mechanical system (Figure 1.4a) employs a compression spring as an articulation unit to overcome pipe bends. This spring ensures that there exists no collision between connected modules and it pushes or pulls the modules in the event of branches or T-unions. A digital model of the working strategy of this robot is represented in Figure 1.13b. An adaptable wheel type piping inspection robot developed by Tătar et al.[TMA07] is another example where springs are employed as an articulation unit. However, this robot employs an extension/tension spring to pass through pipe bends and junctions (Figure 1.13c). The main advantage of employing springs is that they are lightweight and do not be necessarily be actuated through external motors. However, their ability to withstand the robot weight for various pipeline orientations is an important factor that must be studied while incorporating as an articulation unit.



(a) **(b)**

(c)

Figure 1.13 – Some applications of springs in robotics: (a) the robot of Gao et al.[GXS+10] , (b) the robot of Kwon et al.[KLJY08] and (c) the robot of Tătar et al.[TMA07]

Flexible systems

Flexible systems can be categorized under articulation unit as a type of system which can undergo deformations without inducing significant stresses to the connecting elements. They can assume many forms such as a flexible sleeve coupling, flexible shaft or flexible axles. The

robot of Zhang et al.[ZZSJ10], which was studied in the worm type locomotion system employs a flexible helical axle to overcome pipe bends and junctions. The helical axle employed in this robot is made up of elastomer and coiled spring which helps the robot to achieve a squirm type walk inside the pipeline by operating in passive modes (Figure 1.14a). Another application of flexible system is the robot of Suzumori et al.[SMKH99]. This robot employs a passive flexible rubber link in the center and their either ends are connected to planetary wheel mechanisms (Figure 1.14b). This flexible rubber assists the robot to pass through curved sections. Flexible systems also include some actuated systems such as the pneumatic artificial muscle. These structures also find their application in the field of robotics. Tanaka et al.[THN14] developed an inspection robot for sewer pipes that moves like an earthworm by employing a pneumatic artificial muscle system. This robot employs artificial muscle that is constructed from carbon fibers and low-ammonia natural rubber. From a central compressor, consistent pressure is applied to the muscle units, which helps the robot to accomplish the earthworm locomotion inside pipes (Figure 1.14c). Flexible systems are lightweight and simpler in design for the choice of an articulation unit. However, their ability to withstand loads of robot modules will be a concern.

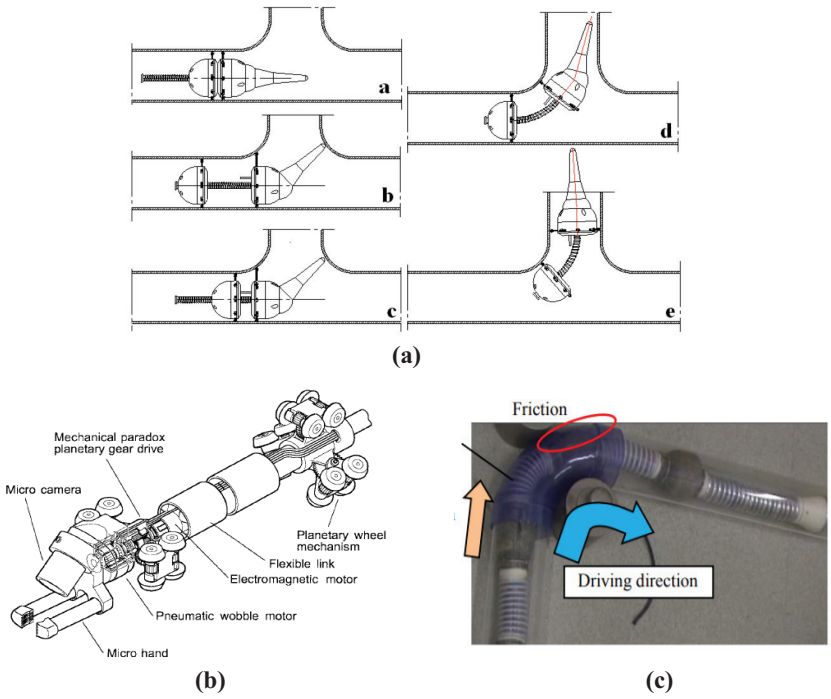


Figure 1.14 – Some applications of flexible systems in robotics: (a) the robot of Zhang et al.[ZZSJ10] , (b) the robot of Suzumori et al.[SMKH99] and (c) the robot of Tanaka et al.[THN14]

Tensegrity structures

Tensegrity structures are deployable mechanisms that comprise strings in tension and bars in compression [SAP+01]. They prove to be an interesting candidate in the field of structural

mechanics and robotics owing to their properties such as low inertia, natural compliance and deployability [FW19]. An interesting application of tensegrity structure is the manipulator developed by Bakker et al.[BMTH15], which can navigate through environments having obstacles. This manipulator employs springs in tension and bars in compression. The manipulator is an under-actuated system wherein cables are actuated from the base to passively tilt the structure and pass through obstacles. Another interesting example is the robot of Fasquelle et al.[FFK+20] where tensegrity mechanisms are assembled in series to mimic a bird neck. This manipulator uses springs in tension that are actuated by motors from base to generate faster shape adaptability similar to that of a woodpecker. The application of tensegrity mechanisms as an articulation unit for piping inspection robots is highly limited in the literature. Tantaworrasilp et al.[TR20] proposed a tensegrity type robot that is capable of performing piping inspection. This prototype uses active prismatic linkages and passive locking mechanisms called restrictor mechanism coupled with tension springs. The prototype can adapt its shape based on pipeline diameters and can propel forward by retracting/extending its length against the pipeline. However, a complete robot testing inside a pipeline has not been performed to validate the contact and locomotion criteria. The modeling of tensegrity mechanisms is complex as the tension of the extension elements must remain positive and static stabilities for the structure must be carefully studied [BAV+17]. Several researches exists in the literature for solving the equations related to the design and stability analysis of tensegrity structures [AG06, BCVR15, WC18]. Tensegrity structures with springs and cables are interesting candidates for articulation units as their actuation strategies do not require motor units to be mounted along with the robot assembly. The actuation for these mechanisms can be carried out from a central station through cables. The representation of the various tensegrity mechanisms from the literature that were studied in this section is shown in Figure 1.15.

1.3.1 Synthesis: Articulation strategies

Similar to the locomotion techniques, synthesis is carried out for the various articulation units that were studied from literature in this section with respect to the operational context. The synthesis is carried out based on the following factors:

1. Complexity: This factor determines how complex or simple is the design of the articulation unit
2. Passive mode: This factor determines if the articulation unit can operate autonomously without the presence of external forces or actuation
3. Active mode: This factor determines if the articulation unit can be controlled externally from actuation units such as a DC-motor
4. Reconfigurability: This factor determines if the articulation unit can adapt according to the pipeline profile such as to bend the robot to overcome a 90° elbow and return back to the original position after crossing the elbow. This factor is also affected by the nature

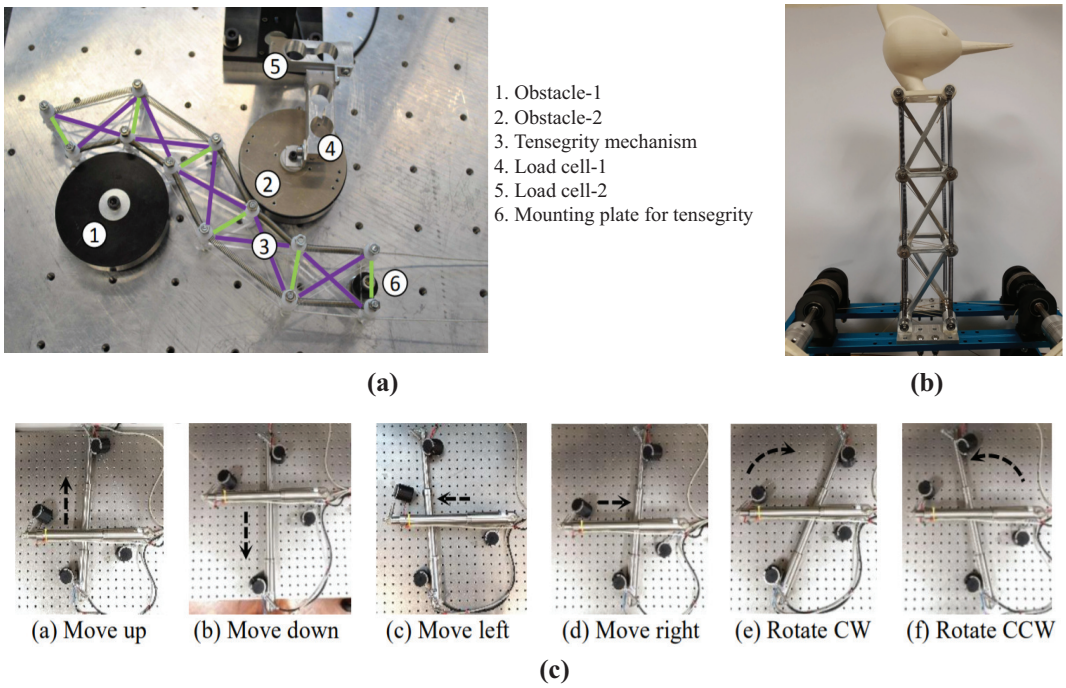


Figure 1.15 – Applications of tensegrity structures in robotics: (a) the robot of Bakker et al.[BMTH15] , (b) the robot of Fasquelle et al.[FFK+20] and (c) the robot of Tantaworrasilp et al.[TR20]

of the articulation unit. If the unit is completely passive, there are chances that the mechanism cannot return to its original position. However, if the unit is active, external actuation can be done to bring the robot back to the initial position.

Based on the above factors, scoring from 1 to 5, with 5 being the best is given to the articulation units studied from literature. The synthesis for the various articulation units studied in this section is provided in Table 1.2. The synthesis from Table 1.2 provides a glimpse of the advantages and drawbacks of each articulation unit based on the scoring. Taking into account the operational context, passive solutions are interesting for overcoming pipe bends without the need for any external actuation. However, a completely passive system is also not a good idea owing to issues such as reconfigurability. An articulation unit that can operate autonomously as well as be controlled from external actuation systems is a good solution, which can possibly be designed by the combination of one or more existing solutions.

1.4 Mathematical concepts and tools

This section analyzes some of the important mathematical concepts and their definitions, which were employed in the thesis. An overview of the working of the mathematical tools associated with software such as the SIORPA library of Maple and the optimization technique features of MATLAB are also being reviewed.

Scoring from 1 to 5	Complexity	Passive mode	Active mode	Reconfigurability
Universal joint	4	3	5	4
Springs	4	5	1	3
Flexible systems	4	5	1	3
Tensegrity structures	4	3	4	4

Table 1.2 – Scoring for articulation units studied in literature with respect to the operational context

1.4.1 Definitions

Varignon's theorem

Definition 1: *Varignon's theorem states that the moment of a force about any point is equal to the sum of the moments of its components about the same point [Hib07].*

Proof: Consider a set of force vectors $\mathbf{f}_1$ to $\mathbf{f}_n$ concentrated on a central point o as shown below in Figure 1.16.

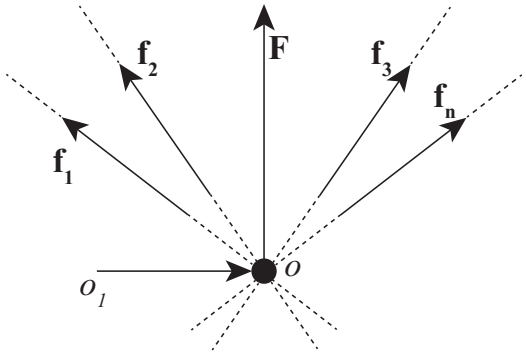


Figure 1.16 – Set of force vectors acting at a point o

The resultant force is given by:

$$\mathbf{F} = \sum_{i=1}^{n} \mathbf{f}_i \tag{1.1}$$

At a point o_1, the moment vector for any force vector $\mathbf{f}_i$ is given by:

$$\tau_{o_i}^{\mathbf{f}_i} = (o - o_1) \times \mathbf{f}_i \tag{1.2}$$

By adding up the torques, the above equation can be written as:

$$\sum_{i=1}^{n} \tau_{o_i}^{\mathbf{f}_i} = (o - o_1) \times \sum_{i=1}^{n} \mathbf{f}_i \tag{1.3}$$

$$\tau_{o_i}^{\mathbf{F}} = (o - o_1) \times \mathbf{F} \tag{1.4}$$

From Eqs. (1.3) & (1.4), it can be proved that the sum of the torque vectors is exactly same as the torque of the sum of the force vectors about the point o_1.

Coulomb's law of dry friction

From the textbook of Popov [Pop17], the definition of Coulomb law can be revisited. From experimental investigations of Coulomb (1736-1806), a frictional force F_R acting between two bodies which are pressed together by a normal force F_N, two properties are being exhibited by the system. They are given by [Pop17]:

- **Static friction:** *In order to set in motion a body lying on an even surface in a state of rest, a critical force, the force of friction F_s, must be overcome. This is force is roughly proportional to the normal force F_N by: $F_S = \mu_s F_N$. The coefficient μ_s is called the coefficient of static friction. It is dependent on the pairing of the contacting materials.*
- **Kinetic friction:** *F_R is the resisting force which acts on a body after the force of static friction has been overcome. Based on experimental findings of Coulomb, the following properties were determined:*
  - *Kinetic friction is proportional to the normal force F_N by: $F_R = \mu_k F_N$*
  - *It exhibits no considerable dependency on the contact area or roughness of the surface*
  - *The coefficient of kinetic friction is approximately equal to the coefficient of static friction: $\mu_k \approx \mu_s$*

Serial & Parallel robots

A serial robot or a serial manipulator is a type of industrial robot in which series of links are connected starting from a base to the end-effector which consists of a tool. A classical example of a serial manipulator is the industrial SCARA robot and anthropomorphic humanoid robot. Serial robots are commonly used in industries owing to their ability to operate inside larger workspaces. The main disadvantage is that every joint of a serial robot must be actuated through its motor unit to perform the desired task. The joints can either be rotary or prismatic. Parallel robots or a parallel manipulator, on the other hand, is a system that has a closed-loop kinematic chain which has an end-effector connected to the base by several independent kinematic chains. Parallel manipulators have numerous advantages such as high accuracy, low inertia of moving parts and high rigidity. However, their limitations include limited workspaces, low dexterity and possibilities of singular configurations within the workspace. Also, there exists Hybrid manipulators, which uses a combination of serial and parallel mechanisms.

Definition 2: *A robot is said to be a serial robot or serial (open-loop) manipulator if its kinematic structure takes the form of an open loop-chain, a parallel manipulator if it is made of a closed-loop chain, and hybrid manipulator if it is consists of both open and closed-loop chains [PD14].*

Forward & Inverse kinematics

Kinematics is a branch of classical mechanics that deals with the study of motion of objects or bodies by not taking into account the forces that cause the motion. It involves the study of displacement, velocity and acceleration and other higher-order derivatives with respect to time or other quantities. In robotics, kinematics can be classified into two types namely: Forward or direct kinematics and Inverse kinematics. In direct kinematics, for given joint parameters, the position and orientation of the end-effector are determined. Solving a direct kinematics equation can also be called as the Direct Kinematic Problem (DKP). For serial robots, the DKP is simpler as the joint parameters can be substituted in the direct kinematics equation to obtain the position of the end-effector. However, in the case of parallel manipulators, the substitution of joint variables in the direct kinematics equation leads to a system of polynomial equations and is usually difficult to solve.

For the inverse kinematics, for a known position of the end-effector, the joint parameters or angles are determined. Solving an inverse kinematics equation can also be called as the Inverse Kinematic Problem (IKP). For serial robots, the determination of solutions to the IKP leads to polynomial equations and involves more computation. For parallel robots, the IKP is usually simpler to solve.

Working mode, Assembly mode and Workspace

Definition 3: *A **Working mode(Posture)** is associated with one solution to the inverse geometric problem of a robot while an **Assembly mode(Position)** is associated with one solution to the direct kinematics problem. The set of points that can be reached by the end-effector is named the **Workspace** [CMRW20].*

Singularities

Singularities or mathematical singularities can be defined as a point or position at which an equation or a surface ceases to be well-behaved in a particular way. Singularities in robotics play an important role to determine the capability of a robot or a mechanism. For a serial robot, assuming that the joint velocity vector is given by $\dot{\mathbf{q}}$ and the cartesian velocity vector is given by $\dot{\mathbf{X}}$, the mapping between these two vectors is made possible by the relation:

$$\dot{\mathbf{X}} = \mathbf{J}\dot{\mathbf{q}} \tag{1.5}$$

In Eq. (1.5), $\mathbf{J}$ is called the Jacobian matrix, which is assembled by partial differentiation of geometric equations with respect to the joint variables. A singular configuration or configurations exist when the Jacobian matrix loses its rank and its determinant becomes zero. For a parallel manipulator, with given vectors of input ($\mathbf{q}$) and output ($\mathbf{X}$) variables, a system of non-linear algebraic equations can be written as [WC97]:

$$F(\mathbf{X}, \mathbf{q}) = 0 \tag{1.6}$$

By differentiating Eq. (1.6) with respect to time leads to the velocity model:

$$\mathbf{A}\mathbf{t} + \mathbf{B}\dot{\mathbf{q}} = 0 \tag{1.7}$$

In Eq. (1.7), $\mathbf{t}$ and $\dot{\mathbf{q}}$ are the joint and cartesian velocity vectors. The matrices $\mathbf{A}$ and $\mathbf{B}$ are the direct and inverse kinematics matrices. Three types of singularities are possible for a parallel manipulator which are given by [GA90, WC97, CW98]:

- **Type-1** or Serial singularities: This condition occurs when the inverse kinematics matrix $\mathbf{B}$ loses its rank and det($\mathbf{B}$) becomes zero. During such singularities, the cartesian or joint velocities completely becomes zero.
- **Type-2** or Parallel singularities: This condition occurs when the direct kinematics matrix $\mathbf{A}$ loses its rank and det($\mathbf{A}$) becomes zero. These singularities can be studied within the workspaces and they are not desirable as the manipulator control might be lost.
- **Type-3** singularities: This condition occurs when both the direct and inverse kinematics matrices $\mathbf{A}$ & $\mathbf{B}$ loses their rank and det($\mathbf{A}$) & det($\mathbf{B}$) becomes zero.

Apart from the above types of singularities, parallel manipulators can encounter other types of singular configurations namely: Constraint singularites and Self-motions.

The concept of constraint singularities was proposed by Zlatanov et al.[ZBG02]. It is a configuration of a parallel mechanism where the number of degrees of freedom is lower than the number of joints in one of the legs. During such singular configuration, the mechanism and the platform will have at least $n + 1$ DOFs [ZBG02]. This singularity is more inherent to the kinematic architecture and does not depend on the actuation mode.

Self-motions are another kind of singularities which are observed in some parallel mechanisms. The study of this kind of singularites was done by Husty [Hus00]. Husty studied and proposed self-motions on a Stewart-Gough platform in which finite mobility of the moving platform was observed when all the actuators were locked. During such motions, the anchor point of the legs followed a spherical path [Hus00]. Self-motions are a very important phenomenon that has to be studied while designing parallel mechanisms because when a robot enters such singular configurations, as there exists many possible poses for the actuated joints, the information about the pose of the mobile platform is lost [BAB+08].

Euler angles

Definition 4: *According to Euler's Rotation Theorem, a rigid body can be guided into any orientation in space by a single rotation. The variable axis and angle of this rotation can be used to parametrize the possible body orientations- an idea that leads to the use of quaternions and Rodrigues parameters to describe the orientation [BZG02].*

Generally, three rotations about coordinate axes are required to bring the body into an

arbitrary orientation. These angles are referred to as the Euler angles. A representation of the Euler angles is shown in Figure 1.17.

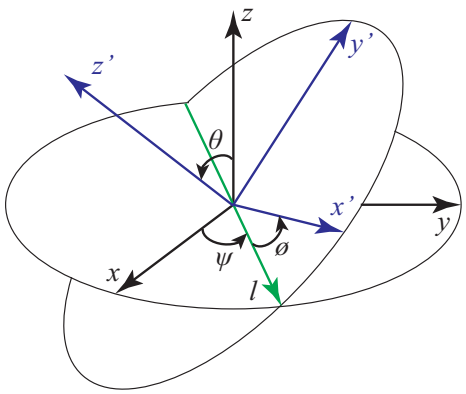


Figure 1.17 – Representation of the Euler angles

In Figure 1.17, xyz represents the fixed frame and $x'y'z'$ represents the rotated frame. The green line l is called the line of nodes which lies on the xy plane. The angle ψ is the angle between x-axis and the line of nodes and it is called the x-convention. It can also be defined between y-axis and the line of nodes, which can be called as the y-convention. The angle θ is the angle between z-axis and z'-axis. The angle ϕ is the angle between the line of nodes and x' axis. For any rotation matrix $\mathbf{R}$, which is at an angle α about a coordinate axis, the matrix $\mathbf{R}$ can assume one of the following forms:

$$\mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}, \quad \mathbf{R}_y(\alpha) = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix}$$

$$\mathbf{R}_z(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.8}$$

For a three dimensional rotation, there are twelve possible sequences for the Euler-angle conventions which are: XZX, XYX, YXY, YZY, ZYZ, ZXZ, XZY, XYZ, YXZ, YZX, ZYX, ZXY. Different authors use different conventions for the Euler-angles. However, the first six combinations are the "proper Euler angles" which correspond to precession, nutation and spin. The last six combinations also fall under the category of Euler angles but are also called the "Tait-Bryan angles". These combinations generally find their applications in aeronautical or automobile industries where the three angles are referred to as pitch, yaw and roll.

Tilt & Torsion angles

Bonev et al.[BZG02, Bon02] proposed the modified Euler-angles, which are otherwise called the "Tilt & Torsion" (T&T) angles for Parallel Kinematic Machines (PKM). Generally, 6-DOF PKM's do not require the rotational degree of freedom about the mobile z'-axis. The conventional ZXZ rotation matrix for the Euler angles can be expressed as:

$$\mathbf{R} = \mathbf{R}_z(\phi)\mathbf{R}_x(\theta)\mathbf{R}_z(\psi) \tag{1.9}$$

However, in the case of T&T angles, only two rotations are employed and the rotation matrix is given by:

$$\mathbf{R} = \mathbf{R}_a(\theta)\mathbf{R}_z(\sigma) \tag{1.10}$$

In Eq. (1.10), θ and σ represents the tilt and torsion angles. The axis a is located with the help of the azimuth angle ϕ. This modified Euler angle set makes the analysis efficient and simpler for the study of PKM's. The representation of the T&T angles is shown in Figure 1.18.

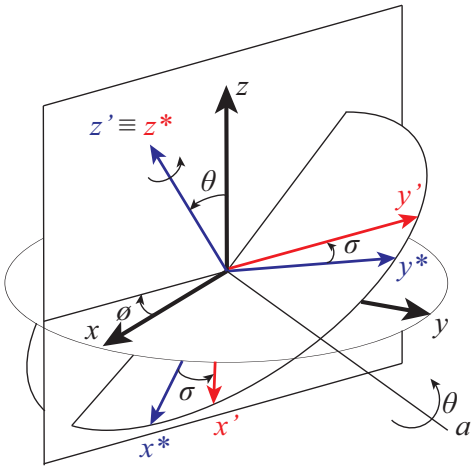


Figure 1.18 – Representation of the Tilt & Torsion angles [BZG02, Bon02]

Groebner bases algebra

The Groebner bases provides an approach to solving a wide range of problems that are expressed in terms of sets of multivariate polynomials having desirable algorithmic properties [Stu05]. The algorithm used for computing this bases is called the Buchberger's algorithm.

Definition 5: *A Groebner basis* $\boldsymbol{G}$ *for a system of polynomials* $\boldsymbol{A}$ *is an equivalence system that possesses useful properties, for example, that another polynomial f is a combination of those in* $\boldsymbol{A}$ *if and only if the remainder of f with respect to* $\boldsymbol{G}$ *is 0. (Here, the division algorithm requires an order of a certain type on the monomials.) Furthermore, the set of polynomials in a Groebner basis have the same collection of roots as the original polynomials. For linear functions*

in any number of variables, a Groebner basis is equivalent to Gaussian elimination [Wei05].

The Groebner bases is an important tool which is employed by researchers in optimization, robotics, statistics to solve algebraic problems using computer software such as Maple, Mathematica. In parallel manipulators or PKM's, the Groebner bases theory is employed to compute the projections of joint spaces ($\pi_{\mathbb{Q}}$) and workspaces ($\pi_{\mathbb{W}}$) [CMW11]. Let $\mathcal{P}$ be a set of polynomials in the variables $X = (x_1, x_2, ...x_n)$ and $q = (q_1, q_2, ...q_n)$. Moreover, let $\mathcal{V}$ be the set of common roots of the polynomial $\mathcal{P}$, let $\mathbb{W}$ and $\mathbb{Q}$ be the projections of $\mathcal{V}$ on the workspace and the joint space respectively. It might not be possible to represent $\mathbb{W}$ (resp. $\mathbb{Q}$) by polynomial equations. Let $\bar{\mathbb{W}}$ (resp. $\bar{\mathbb{Q}}$) be the smallest set defined by polynomial equations that contain $\mathbb{W}$ (resp. $\mathbb{Q}$). The main goal is to compute the polynomial equations defining $\bar{\mathbb{W}}$ (resp. $\bar{\mathbb{Q}}$). A Groebner basis $\mathcal{P}$ is a polynomial system equivalent to $\mathcal{P}$, satisfying some additional properties. The Groebner bases of a system depends on the chosen ordering on the monomials.

For the projection ($\pi_{\mathbb{W}}$), an ordering is chosen that eliminates q such that the Groebner bases $\mathcal{P}$ contains exactly the polynomials defining $\bar{\mathbb{W}}$.

For the projection ($\pi_{\mathbb{Q}}$), an ordering is chosen that eliminates X such that the Groebner bases $\mathcal{P}$ contains exactly the polynomials defining $\bar{\mathbb{Q}}$.

Cylindrical Algebraic Decomposition (CAD)

Cylindrical Algebraic Decomposition (CAD) is an important tool which is employed by researchers and engineers for symbolic computations. It was introduced by Collins in 1975 for identifying solutions to mathematical problems by quantifier elimination [Col75]. The algorithm was then further refined for reducing computation times. In general, there are two main phases of the CAD algorithm. In the first phase, which is called the *projection* , an operator is repeatedly imposed on polynomials to produce sets of few variables. The second phase, which is called the $lifting$, CAD's of real space is built incrementally by dimension according to the real roots of the polynomials [EBDW14].

Definition 6: *A cylindrical algebraic decomposition (CAD) is: a* ***decomposition*** *of* $\mathbb{R}^n$*, meaning a collection of cells which do not intersect and whose union is* $\mathbb{R}^n$*;* ***cylindrical****, meaning the projections of any pair of cells with respect to a given variable ordering are either equal or disjoint, and* ***(semi)-algebraic****, meaning each cell can be described using a finite sequence of polynomial relations [EBDW14].*

For parallel manipulators, the CAD algorithm is an essential tool to characterize the regions bounded by singularities or inequalities. The CAD algorithm can be employed to construct the feasible workspaces for a parallel manipulator by decomposing into smaller cells. Each cell will have its own polynomial equation and they can be employed to construct the workspaces (resp. joint spaces) and isolate the singularity regions. CAD algorithm can be employed with the help of software such as Maple which is ideal for symbolic computations.

Weierstrass substitution

Algebraic tools are employed for solving constraint equations which are generally algebraic. Usually, these equations are in the form of trigonometric entities. The Weierstrass substitution (named after Karl Weierstrass 1815–1897) or the tangent half-angle substitution is an important trigonometric relation for converting a rational function into an ordinary form. The substitution is given by:

$$t = \tan\left(\frac{\theta}{2}\right) \tag{1.11}$$

In Eq. (1.11), θ is any arbitrary angle. This convention is also helpful in simpliyfing the *sine*, *cosine* and *tangent* functions as:

$$\sin(\theta) = \frac{2t}{1+t^2}\ ,\ \cos(\theta) = \frac{1-t^2}{1+t^2}\ ,\ \tan(\theta) = \frac{2t}{1-t^2} \tag{1.12}$$

The conventions given in Eq. (1.12) are important for solving algebraic problems using the CAD algorithm as they simplify quickly the polynomials and the computations times are reduced.

1.4.2 The SIROPA library: Maple

The SIROPA is a library developed for Maple for analyzing the singularities, workspace and joint space of serial and parallel manipulators and tensegrity structures [JCB+18, CMRW20]. The library is composed of two main parts: the first one employs algebraic tools to solve kinematic equations, including trigonometric conversions in the algebraic form and the second one, SIROPA, provides modeling, analyzing and plotting functions for different kinds of manipulators. The list of all functions that are available in the SIROPA library is represented in Figure 1.19. In this section, some of the important functions that were employed in the thesis will be explored.

CreateManipulator

The first and most important function of the SIROPA library is the **CreateManipulator**, which allows the virtual creation of the desired manipulator for analysis. The definition of the **CreateManipulator** function in SIROPA is provided below in Listing 1.1.

```
1 CreateManipulator := proc (
2     sys[c]          :: list ({algebraic , algebraic=algebraic , algebraic <
3     algebraic})
4     cart[c]         :: list(name),
5     arti[c]         :: list(name),
6     passive[o]      :: list(name),
7     geompars[c]     :: list (name),
8     spec[o]         :: {list ,set}(name=algebraic),
```

```
    plotrange[o]     :: list(name=range),
    points[p]        :: list(name=list(algebraic)),
    loops[p]         :: list(list(name)),
    chains[p]        :: list (list(name)),
    actuators[p]     :: list({list(name),name}),
    model[o]         :: string                    := "No name",
    precision[o]     :: integer                   := 4,
    {
    noradical        :: truefalse                 := false
    }
)
```

Listing 1.1 – The definition of **CreateManipulator** function in SIROPA [JCB+18]

In Listing 1.1, **[c]** represents the compulsory inputs or arguments for the function. The parameter sys accepts the equations governing the manipulator. For a parallel manipulator, these equations are generally the relation between base and the end-effector which gives the IKP. Constraints are also set if necessary for the equations, such as non-zero or non-negative conditions. The parameters cart and arti are the pose variables and the articular variables of the manipulator. The parameter geompars accepts other parameters which are present in sys but are not present in cart or arti. The parameters points, loops and chains define the input coordinates that connect with sys to create the manipulator geometry and closed-chains.

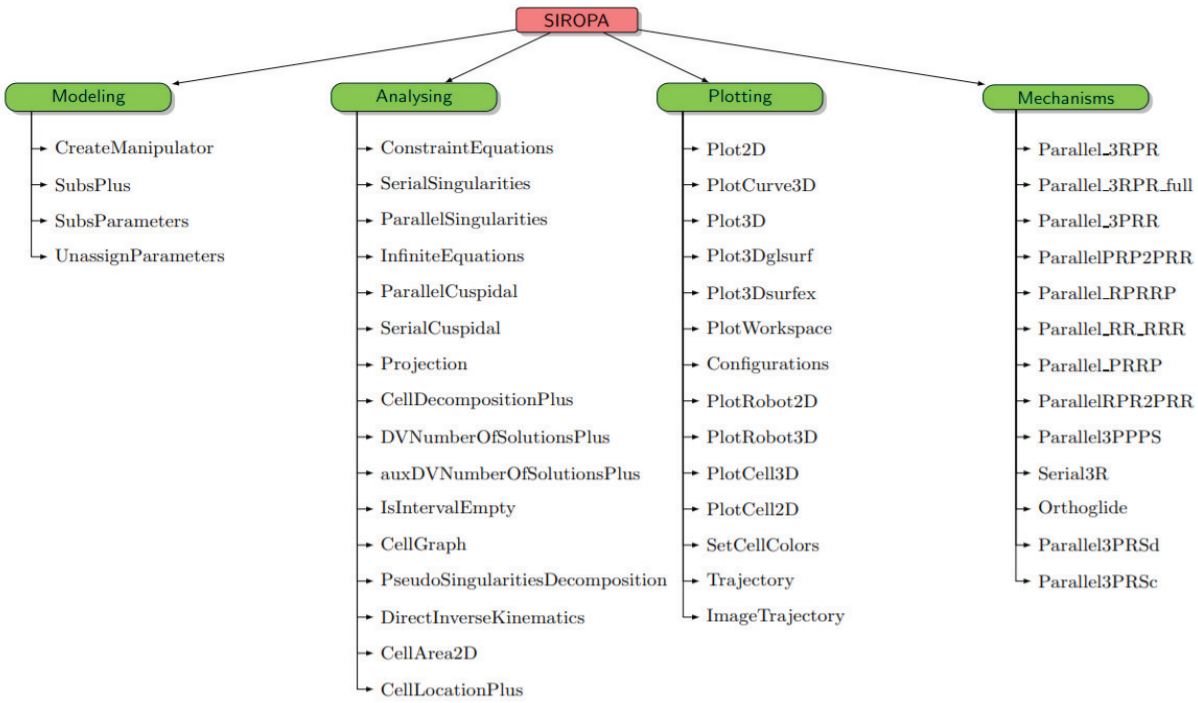


Figure 1.19 – The various functions of the SIROPA library in Maple [JCB+18]

PlotRobot3D

Once the manipulator equations are generated with the **CreateManipulator** function, the **PlotRobot3D** function can be employed to visualize a simple 3D model of the robot. Animation can also be done to view various configurations of the robot over a period of time. The definition of the **PlotRobot3D** function in SIROPA is provided in Listing 1.2. The user can provide some input values of pose variables (resp. articular variables) to visualize and understand the results of IKP (resp. DKP) for the manipulator. For any given inputs, the output values are also displayed to the user along with the determinant values of the Jacobian matrix.

```
PlotRobot3D := proc (
        robot     ::Manipulator,
        spec      ::seq(name=constant),
        k     ::{integer,range,list(integer)}     :=     ..,
        {
        color := [],
        legendvars := subs (map(s->lhs(s)=NULL,[spec]),
        [op(robot:-ArticularVariables), op(robot:-PoseVariables),
        op(robot:-PassiveVariables), 'det(J)']),
        nolegend      ::truefalse     :=     false,
        noconstraints     ::truefalse     :=     false
        }
                                                    )
```

Listing 1.2 – The definition of **PlotRobot3D** function in SIROPA [JCB+18]

ConstraintEquations

The function **ConstraintEquations** is important to create a set of equations with inequalities. Normally, the inverse kinematic equations are transformed into constraint equations by setting minimum and maximum limits. For example; if a manipulator has three equations for the IKP, the number of constraint equations can be six with each IKP having a lower limit and an upper limit. This function is useful for tracing the joint limits. The definition of the **ConstraintEquations** function is provided below in Listing 1.3.

```
ConstraintEquations := proc (robot   ::Manipulator,
                      {
                      constraints   ::truefalse     :=     false
                      }
                      )
```

Listing 1.3 – The definition of **ConstraintEquations** functions in SIROPA [JCB+18]

SerialSingularities, ParallelSingularities & Projection

The functions **SerialSingularities** and **ParallelSingularities** as the name indicates are employed to generate the determinant values of the inverse and the direct kinematics matrices **B**

and **A**. The generated equations are then employed for CAD algorithm to study the workspaces of the manipulator with respect to the pose variables (resp. articular variables). The function **Projection** is employed to project the equations of parallel or serial singularities of the robot on the workspace or the joint space. The definition for the three functions is provided in Listing 1.4. The parameter robot contains the equations that are generated using the **CreateManipulator** function.

```
s1      := SerialSingularities( robot          ::Manipulator    )
s2      := ParallelSingularities( robot        ::Manipulator    )
s1_cart := Projection(s1,       robot:-PoseVariables        )
s2_cart := Projection(s2,       robot:-PoseVariables        )
s1_art  := Projection(s1,       robot:-ArticularVariables       )
s2_art  := Projection(s2,       robot:-ArticularVariables       )
```

Listing 1.4 – The definition of **SerialSingularities** **ParallelSingularities** and **Projection** functions in SIROPA [JCB+18]

CellDecompositionPlus

The **CellDecompositionPlus** is the important function of the SIROPA library that has been exploited in this thesis. This function decomposes the parameter space of a parametric polynomial system into cells in which the original system has a constant number of solutions [Jha16]. The function works exactly similar to the **CellDecomposition** function of **RootFinding[Parametric]** except with the difference that it handles trigonometric expressions and utilizes the tangent half-angle substitution for faster computations. The CAD algorithm is carried out in Maple with the help of this function. The definition of **CellDecompositionPlus** is provided below in Listing 1.5.

```
CellDecompositionPlus := proc (
        equ       ::list(algebraic),
        ineq      ::list(algebraic),
        vars      ::list(name),
        pars      ::list(name)    :=     [ op(indets([equ,ineq],name)
                                         minus {op(vars)}) ],
        {
        nofactor      ::truefalse     :=     false,
        gbfactor      ::truefalse     :=     false,
        norealrootstest   ::truefalse     :=     false
        }
        )
```

Listing 1.5 – The definition of **CellDecompositionPlus** function in SIROPA [JCB+18]

The parameter robot accepts only the equations defined in **CreateManipulator** function. The parameter ineq handles list of trigonometric expressions or polynomials with inequalities that ensure that the equations are greater than zero. The parameter vars can contain the articular variables (resp. pose variables) and pars can contain the pose variables (resp. articular

variables). For example, for a given manipulator, the parallel singularities or the serial singularities equations can be passed to ineq to generate the cells defining the workspaces or joint spaces. The execution time for the **CellDecompositionPlus** function depends on the complexity of the system as well as the configuration of the computer. The results of **CellDecompositionPlus** are generated as cells where each cell contains the polynomial/trigonometric entities of the given input parameters. These cells are then traced with the help of **PlotCell2D** or **PlotCell3D** depending on the number of output parameters. These plot functions requires the output cells of **CellDecompositionPlus** for a specified range of the pose variables or the articular variables.

The SIROPA library also offers many other functions that are useful for algebraic computations. A detailed explanation for the rest of the functions shown in Figure 1.19 that were not discussed and examples of some manipulator architectures can be found in [JCB+18] and [SIR].

1.4.3 Optimization techniques using MATLAB

This section gives a glimpse about the optimization tools that were employed in this thesis. Optimization technique refers to the selection of a best possible solution or elements from set of available alternatives. Optimization is an important and powerful tool which finds its application in the field of engineering, statistics, molecular biology and much more.

Definition 7: *A **mathematical optimization problem**, or just **optimization problem**, has the form*

$$\begin{aligned} &\text{minimize} \quad f_0(x) \\ &\text{subject to} \quad f_i(x) \leq b_i, i = 1..m \end{aligned} \tag{1.13}$$

*Here the vector $x = (x_1, ...x_n)$ is the **optimization** variable of the problem, the function $f_0 : \mathbf{R}^n \to \mathbf{R}$ is the **objective function**, the functions $f_i : \mathbf{R}^n \to \mathbf{R}$, i = 1,...m, are the (inequality) **constraint functions**, and the constants b_1,...b_m are the limits, or bounds, for the constraints. A vector x^* is called **optimal**, or a **solution** of the problem (1.13), if it has the smallest objective value among all vectors that satisfy the constraints: for any z with $f_1(z) \leq b_1$,...,$f_m(z) \leq b_m$, we have $f_0(z) \geq f_0(x^*)$ [BBV04].*

Sometimes, optimization problems does not necessarily require constraints and such problems are referred to as Unconstrained optimization problems. Generally, optimization techniques can be classified into Deterministic and Heuristic approaches. Deterministic approach solves an optimization problem in a structured manner, thereby leading to efficient solutions with respect to design variables and constraints [GB09]. However, they require well-defined mathematical equations and require more computational times to arrive at a solution. Heuristic approach, on the other hand, operate on randomness. They require lesser computational times but the accuracy of their results are not guaranteed. The metaheuristic or evolutionary algorithms is a type of heuristic approach that do not require mathematical definitions such as gradients to

find optimal solutions. They operate on a population based approach where more than one solution can be determined and they are best suited for multi-objective optimization problems. Optimization problems can be solved with the help of softwares such as MATLAB, Python, Maple. Some of the MATLAB optimization functions that were employed in this thesis will be reviewed in this section.

Deterministic approach: "fmincon" function

For solving constrained optimization problems that are non-linear, the function *fmincon* of MATLAB can be employed. This function operates on a gradient based approach where a local (sometimes global) minima or maxima for an input objective function is determined by an iterative procedure [fmi]. The syntax for the *fmincon* function is provided below in Listing 1.6.

```
[x, fval] = fmincon(fun, x0, A, b, Aeq, beq, lb, ub, nonlcon, options)
```

Listing 1.6 – The "fmincon" syntax in MATLAB

The first input parameter of fmincon is *fun*. This is the objective function that has to be minimized or maximized. A separate function can be written and passed as a handle if necessary. For maximization, a negative sign is included in the output variable of the objective function. The parameter x_0 is the initial guess for the optimization problem and it is the point from which the function attempts to initiate the problem. The initial guess is carefully chosen as values out of range might lead to convergence issues. The parameters *A*, *B*, *Aeq* and *beq* are linear inequality and equality constraints. They are not employed frequently unless a linear constraint is present in a non-linear optimization problem. It is important to put braces in the form of '[ ]' if these constraints are not present. The parameters *lb* and *ub* are the lower and upper bounds for the design variables. They can be scalars or vectors depending on the nature of the input design variable. The bounds are also the search zones for finding optimal values of the design variables. The function *nonlcon* is the non-linear constraint function which is written as a separate function and passed as a handle to fmincon. The *nonlcon* function has inequality and equality constraints as outputs. The parameter *options* is used to control parameters of optimization such as function tolerances, display of iteration and algorithm type [MAT]. The output variables after performing optimization are *x* and *fval*. The variable *x* contains the optimized values and *fval* is the value of objective function for the output variable *x*. The function *fmincon* is best suited for multi-variable mono-objective problems. Solving optimization problems with *fmincon* should be given careful attention because when the complexity of the functions increases, convergence issues might occur while performing gradient problems. For any given input function subject to bounds and constraints, the *fmincon* function can find only one solution to the optimization problem. This function is not suitable for solving multi-objective optimization problems.

Metaheuristic approach: Genetic algorithm

For solving multi-objective functions or if multiple solutions are necessary to be found for an optimization problem, the metaheuristic or evolutionary algorithms of MATLAB can be employed. The function *ga*, which stands for Genetic algorithm is ideal for solving optimization problems by a metaheuristic approach [ga]. This function operates on randomness over a defined or default population range to identify optimal solutions. The syntax for the *ga* function is provided in Listing 1.7.

```
[x,fval] = ga(fun,nvars,A,b,Aeq,beq,lb,ub,nonlcon,options)
```

Listing 1.7 – The "ga" syntax in MATLAB

The parameters of function ga is similar to fmincon, except with a difference that the second input parameter for ga expects the number of variables to be optimized. The genetic algorithm does not require an initial guess as they operate on randomness. The definition of objective functions and constraints are set in the same way as fmincon. The *options* parameter of ga is however different where the user can set tolerances for constraints or set population range for identifying potential solutions. For solving multi-objective optimization problems, the function gamultiobj replaces ga in Listing 1.7. The user can generate Pareto front plots using this function where a swarm of solutions can be obtained with a compromise between the objective functions which is limited to three. The execution times of genetic algorithm depends on the population size chosen by the user. As the algorithm operates on randomness, a global minimum is never guaranteed for the evolutionary algorithms. Hence, the user has to choose the ideal or optimal value from the solutions obtained after optimization.

Apart from the *fmincon* and *ga* functions, MATLAB offers various other optimization functions, which can be explored in the optimization toolbox of MATLAB and in the website of Mathworks.

1.5 Conclusions

This chapter dealt with the literature review of piping inspection and description of some mathematical concepts and tools that are employed in this thesis. By classifying locomotion principles of piping inspection robots viz: Mechanical type and Bio-inspired type, a literature review of the existing robots that fall under these categories were studied. A synthesis table for the locomotion techniques was created in line with the operational context. Followed by that, various articulation strategies that are present in literature were explored and a synthesis table for the operational context was also presented. Some of the important mathematical concepts which are employed in this thesis were defined and explained. Finally, some mathematical tools and functions associated with the SIROPA library of Maple and optimization tools of MATLAB software were presented.

CHAPTER 2

Rigid bio-inspired piping inspection robot

Problem statement:

1. *What are the different ways for an inspection robot to accomplish locomotion inside a pipeline?*
2. *What are the forces required to ensure good contact with the walls of a pipeline having various orientations and diameters?*

2.1 Introduction

In industries such as nuclear, chemical and sewage, it is difficult for a human being to perform inspection or interventions. Manual intervention could not only lead to long-term radiation effects but also pose a serious threat to human life. Piping inspection robots play an essential role in such scenarios as they reduce human effort and impose lesser risks to life. Generally, the locomotion principles of these robots can employ mechanical systems or can be inspired from animals. Kassim et al. [KPN+06] proposed a distinction between these two categories where the mechanical systems can use wheels and pulleys [Tak00], telescopic system [Mas78], impact modules [HHY+01] or natural peristalsis [IS97] to accomplish the locomotion. On the other hand, the bio-inspired systems can mimic their locomotions from animals such as earthworms [DSS61], snakes [ITH88], millipedes [Uts79], lizards [TT97] or an octopus [GCITS88]. Presently, several piping inspection robots have been developed. Nayak et al.[NP14] proposed a screw-type robot

that is capable of working inside 127–152 mm diameter pipelines. Zhang et al.[ZZSJ10] developed a flexible inchworm type robot that can adapt itself to varying diameters of pipes. Kwon et al.[KLJY08] developed an inspection robot that employs a caterpillar module with four bar linkages to work inside 100 mm diameter pipelines with bends and junctions. The design and development of inspection robots for pipelines having diameters less than 150 mm is a big challenge. Besides, these piping inspection robots encounter some key problems, which are:

- Movement of the robot inside straight or curved sections of pipe while towing a payload.
- Accurate positioning of the robot on the zone to be intervened.
- Effective communication between onboard sensors and central station.
- Performing the desired mechanical activity such as drilling by withstanding the load of the mounted device.

As a part of project with AREVA, a novel bio-inspired piping inspection robot that mimics the locomotion of a caterpillar was proposed and developed by Henry et al.[HCP+14] and Chablat et al.[CVB18]. This robot is capable of addressing the issue of design of inspection robots for pipelines having diameters less than 150 mm. Using electrical actuators with leg mechanisms and spindle drive units, this robot accomplishes the motion of a caterpillar and can work inside 40–94 mm diameter pipelines. Three architectures were studied for optimization viz: slot-follower, four-bar and six-bar mechanisms in [HCP+14] for the choice of leg mechanism for this robot. By following a heuristic optimization [RW17] technique in MATLAB, which aimed at minimizing the space occupied by the mechanism and maximizing the force transmission factor, the slot-follower mechanism was chosen for the robot [HCP+14]. However, a detailed force analysis was not performed on the robot during clamping and locomotion conditions. This bio-inspired proves to be ideal for the operational context as it can adapt to varying pipeline diameters as well as operate within pipelines having diameters less than 150 mm. This chapter thus focuses on the numerical and experimental validations of this bio-inspired robot. Under static conditions, there exists frictional forces between the robot legs and the pipeline walls. Coulomb's law of static friction is used to estimate the normal forces that exists during clamping. Using Maple and MATLAB, the actuator forces of the motors used in the legs of the robot are estimated. However, in order to estimate the forces on the central actuator, a dynamic approach is necessary. Forces and torques on robotic links can be estimated using Lagrangian or Newton-Euler approaches. The former involves more computation times and appears more complex. A Newton-Euler approach is thus followed in order to estimate the forces on actuators during locomotion. A comparison is then made between both the force models. Using DC-motors, spindle drives, leg mechanisms and standard fasteners, the entire robot is realized at LS2N. Experiments are then carried out on this prototype, where the peak results of the static force models are set as threshold limits for ensuring tight clamping with the pipeline walls. The real-time forces induced on the actuators under locomotion are then interpreted using a Force-control algorithm with the help of a BeagleBone black micro-computer inside a 74 mm diameter straight pipe.

2.2 Locomotion Principle and Kinematic Equations

The locomotion principle of the robot is inspired from a caterpillar. The motion of a caterpillar is subdivided into three modules: a central blocking module and two elongation modules with one at the front and the other at the rear. A classical way to mimic this locomotion can be accomplished with the help of pneumatic bellows for blocking and electric motors for elongation [ACB99, ALT+00]. However, while moving inside nuclear power plant pipelines there exists dust particles which limits the use of pneumatic systems. Hence, electrical actuators are used in the robot. A comparative study by Henry et al. [HCP+14] proposes a mechanism that is capable of simulating the locomotion of a caterpillar, which is mimicked by using three electrical actuators. An outline of the robot inside a pipeline of radius r is represented below in Figure 2.1.

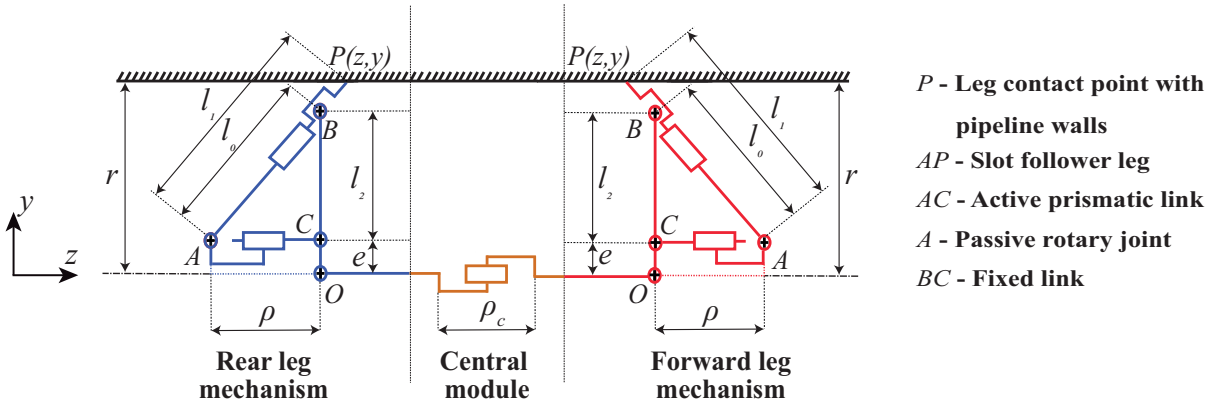


Figure 2.1 – Outline of the bio-inspired robot representing the actuators and leg mechanisms inside a pipeline of radius r.

In Figure 2.1, O represents the reference point for the leg mechanisms. Active spindle drives are used in the front and rear modules to attain a prismatic motion. The stroke lengths of the prismatic link AC is represented by ρ. Three sets of legs, each on the front and rear modules are mounted at angles of $2\pi/3$ radians. The leg mechanisms ensure tight contact with the walls of the pipeline under static and dynamic phases of the robot and they can adapt their stretch based on the diameter changes of pipeline. The central module in Figure 2.1 also uses an active spindle drive to accomplish the elongation/retraction process and its stroke length is represented by ρ_c. The slot-follower leg mechanism is mounted at point A (passive rotary joint) of the actuator. The leg can slide along point B and l_0 represents the length of AB. Each leg will have its contact at point P on the pipeline and l_1 represents its length. Owing to the motor unit used for the spindle drive, the legs are mounted at an offset of distance e from the central axis. l_2 represents the distance between points B and C. This robot accomplishes the locomotion of caterpillar in six-steps, as represented in Figure 2.2. A hyper-static or over-constrained posture exists when both sets of legs of the robot are in contact with the walls. The coordinates $(z_p\ ,\ y_p)$ of point P in Figure 2.1 can be written to obtain the closed-loop equation, which is given by:

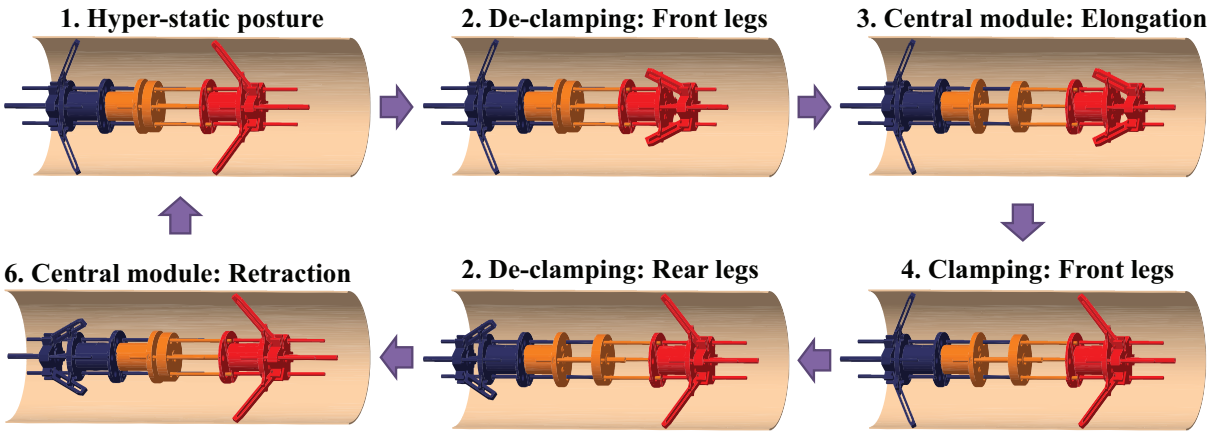


Figure 2.2 – Demonstration of the six locomotion steps to attain caterpillar motion by the robot

$$y_p = \frac{l_2 l_1}{l_0} + e \tag{2.1}$$

$$z_p = \frac{\rho l_1}{l_0} - \rho \qquad \text{where} \quad l_0 = \sqrt{l_2^2 + \rho^2} \tag{2.2}$$

The direct and inverse kinematic problems (DKP and IKP) for the leg mechanism can be derived using Eqs. (2.1) and (2.2). For the DKP, for a known stroke length ρ and dimensional parameters of leg, the radius of the pipeline r or y_p can be estimated. For the IKP, for a known diameter or radius of the pipeline and dimensional parameters of the leg, the actuation length ρ can be estimated. The equations are derived using Maple and they are given by:

$$\text{DKP:} \quad y_P = \frac{l_2^2 e + \rho^2 e + l_1 l_2 \sqrt{l_2^2 + \rho^2}}{l_2^2 + \rho^2} \tag{2.3}$$

$$\text{IKP:} \quad \rho = \frac{l_2 \sqrt{l_1^2 - e^2 + 2 e y_P - y_P^2}}{y_P - e} \tag{2.4}$$

Using the kinematic equations, the Jacobian matrix can be calculated, which is used to estimate the forces. The relation between contact forces and the actuator forces is given by:

$$F_a = \mathbf{J}^T F_p \tag{2.5}$$

The estimation of actuator forces F_a is discussed in the upcoming sections.

2.3 Force models of the robot

For the calculation of forces on the motor during static and dynamic phases, two possible models are considered, which are depicted in Figure 2.3. In Model-A, the rear leg modules are clamped with the pipeline walls whereas in Model-B the front leg modules are clamped with the pipeline walls. These two models are employed for the force modelling of the robot.

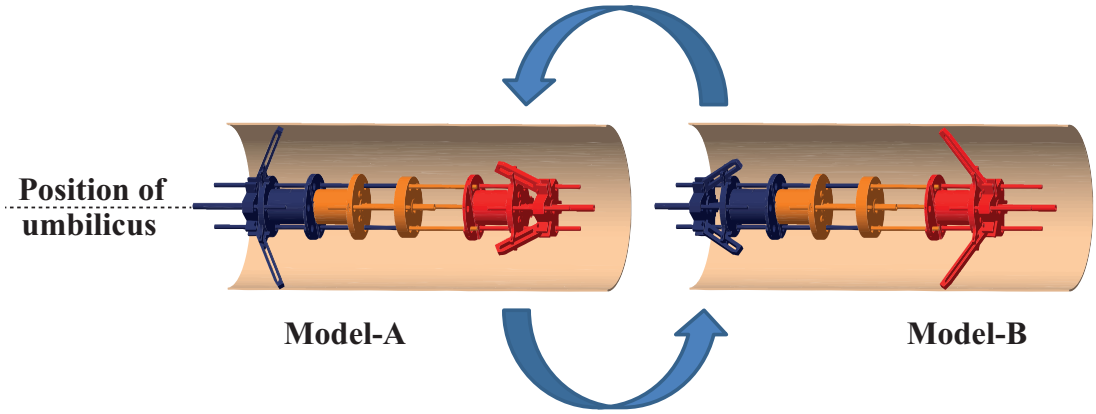


Figure 2.3 – Two models of the robot considered for the force analysis

2.3.1 Static force model

The static algorithm mainly aims at estimating the contact forces between the robot and the pipeline. With the contact forces estimated, the forces induced on the actuators of the front (Model-B) and rear modules (Model A) can be estimated. For the static force model, the robot is considered to be rigid and it can work only through straight pipelines having varying diameters. During the non-hyperstatic phases of clamping, the robot behaves similar to a cantilever beam where on one side three legs have contact points with the pipeline walls and on the free end the self-weight of robot generates a moment. The actuator forces and torques are affected by the orientation of the pipeline and the orientation of the robot about its own axis. The representation of these two orientations are shown below in Figure 2.4.

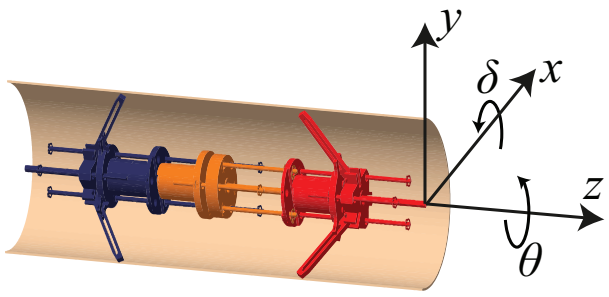


Figure 2.4 – Representation of the rotation angles of pipeline (δ) and robot (θ).

At every contact point, there exists three main forces between the legs and pipeline, which are the:

- N—Normal forces
- T_l—Tangential longitudinal forces
- T_r—Tangential radial forces

The representation of these forces at the contact points with the pipeline walls is shown below in Figure 2.5.

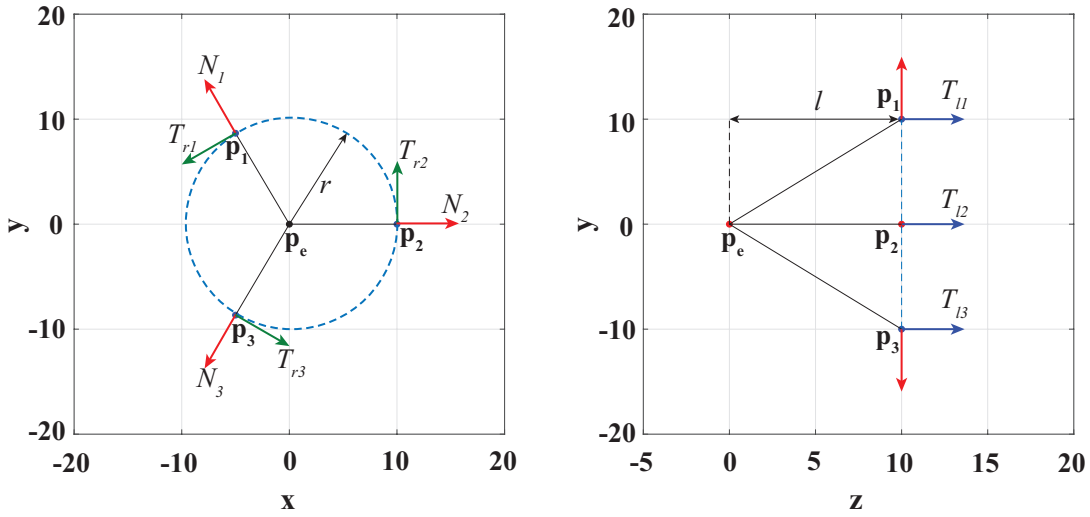


Figure 2.5 – Schema of the contact forces during clamping as seen in $x-y$ (left) and $y-z$ (right) planes [CVB18]

In Figure 2.5, $\sum_{i=1}^{3} \mathbf{p}_i$ indicates the vector coordinates of the contact points of the leg mechanism. The free end of the robot or the position of center of gravity (CG) on the z-axis is represented by vector $\mathbf{p}_e$, which is located at a distance l from the clamped end. The coordinates of $\mathbf{p}_e$ is given by:

$$\mathbf{p}_e = [0, 0, 0]^T \tag{2.6}$$

The coordinates of the vector at the contact points of the leg mechanism are given by:

$$\mathbf{p}_i = \mathbf{R}_z(\theta)\mathbf{n}_i \tag{2.7}$$

$$\text{where} \quad \mathbf{R}_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \& \quad \mathbf{n}_i = \begin{bmatrix} r\cos\left(\frac{2\pi(i-1)}{3}\right) \\ r\sin\left(\frac{2\pi(i-1)}{3}\right) \\ l \end{bmatrix} \quad \text{with } i = 1, 2, 3$$

In Eq. (2.7), $\mathbf{n}_i$ are the coordinates of $\mathbf{p}_i$ in the local reference frame of the contact points of the legs. $\mathbf{R}_z$ is the rotation matrix of robot about the z-axis. About the vector $\mathbf{p}_e$, there exists a wrench equation which consists of forces($\mathbf{f}_i$) and moments($\mathbf{m}_i$) and it is given by:

$$\tau_e = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{m}_i \end{bmatrix} \tag{2.8}$$

$$\text{where} \quad \mathbf{f}_i = [f_{ix}, f_{iy}, f_{iz}]^T \quad \& \quad \mathbf{m}_i = [m_{ix}, m_{iy}, m_{iz}]^T \quad \text{with } i = 1, 2, 3$$

According to the static equilibrium conditions, the total sum of forces and moments are zero, which is given by:

$$\sum_{i=1}^{3} f_i = 0 \quad \& \quad \sum_{i=1}^{3} m_i = 0 \tag{2.9}$$

The tangential radial and longitudinal forces can be estimated directly from the reaction forces at the contact points of the legs with the pipeline walls. The force vector at the contact points is given by:

$$\mathbf{f}_i = \begin{bmatrix} T_{ri} \cos\left(\frac{2\pi(i-1)}{3} + \frac{\pi}{2}\right) \\ T_{ri} \sin\left(\frac{2\pi(i-1)}{3} + \frac{\pi}{2}\right) \\ T_{li} \end{bmatrix} \quad \text{with } i = 1, 2, 3 \tag{2.10}$$

At the contact points, there exists no moments and the wrench equation is comprised of only resultant forces and is represented by ${}^{i}\tau_i$. About the vector $\mathbf{p}_e$, the wrench equation ${}^{i}\tau_0$ can be calculated using the Varignon's theorem and it is given by:

$${}^{i}\tau_0 \quad = \quad {}^{i}\tau_i + (\mathbf{p}_i \times \mathbf{f}_i) \qquad \text{with} \quad i = 1, 2, 3 \tag{2.11}$$

By using Coulomb's law of dry friction, the normal forces can be estimated from the tangential radial and longitudinal forces. The equation is given by:

$$N_i \geq \frac{\sqrt{T_{ri}^2 + T_{li}^2}}{\varphi} \quad \text{with } i = 1, 2, 3 \tag{2.12}$$

Generally, the material used for nuclear or chemical pipelines is steel. For the legs of the robot, bronze is used as it has good mechanical properties such as self-lubrication and wear resistance. As the clamping forces could lead to deformations, a metallic alloy is considered over other materials such as ceramics or rubber. The coefficient of friction represented by φ in Eq. (2.12) is taken as 0.4 [CVB18]. With the individual normal forces estimated, the global normal force N_{total} can be estimated by summing the normal forces of each leg. N_{total} also indicates the magnitude of contact forces F_p. The magnitude of actuator force F_a can be estimated by using Eq. (2.5), which is modified and rewritten as:

$$F_a = \eta_f N_{total} \tag{2.13}$$

In Eq. (2.13), η_f is the force transmission factor that can be generated using the Jacobian of the DKP of the leg mechanism whose equation is given by:

$$J = \eta_f = \frac{\rho l_1 l_2}{(l_2^2 + \rho^2)^{\frac{3}{2}}} \tag{2.14}$$

In Eq. (2.11), the wrench model has dependencies on the position of CG and in Eq. (2.14), the factor η_f has dependencies on the radius of the pipelines r. When the pipeline is horizontal or angled, the robot behaves similar to a cantilever beam and the distance between the clamped

ends to the position of CG is essential to determine the forces induced on the actuators. This analysis will be exploited in the upcoming sections.

2.3.2 Dynamic force model

The static force model was sufficient to understand the clamping forces and leg actuator forces. For estimation of the torques on the central actuator as well as the leg actuators under locomotion, the dynamic model has to be validated in order to ensure the stability of the robot while moving inside horizontal and angled pipelines. The dynamic model for the bio-inspired robot was initially analyzed by Boyer et al.[BHPC16] as a part of an internal and confidential project with AREVA. The robot is a multi-body mobile-based system with closed loops. It moves relative to a fixed Galilean frame. Like the kinematics problem, there exists a direct dynamic model and an inverse dynamic model. The former is used for estimating the accelerations in the links of the robot with the help of wrench equations. The inverse dynamic model is used for estimating torques and wrenches using accelerations and velocities of the links. In this context, the inverse dynamic model is employed in order to estimate the torques generated on the motors during locomotion. The model represented in Figure 2.3 will be also employed for dynamic force modelling. Each model is composed of 29 bodies as represented in Figure 2.6 with 9 non-zero masses, 3 active links (red arrows), 18 passive links (blue arrows) and 8 clamping's (Ei). The details of the key body components of Figure 2.6 and the corresponding material used is provided below in Table 2.1.

Body No.	**Description**	**Material**
Body-1	Left leg actuator	Steel
Body-4	Left leg-1	Bronze
Body-8	Left leg-2	Bronze
Body-12	Left leg-3	Bronze
Body-15	Central & right leg actuators	Steel
Body-18	Right leg-1	Bronze
Body-22	Right leg-2	Bronze
Body-26	Right leg-3	Bronze
Body-29	Umbilicus(Cables)	Copper

Table 2.1 – Description of key body parts of the robot from Figure 2.6

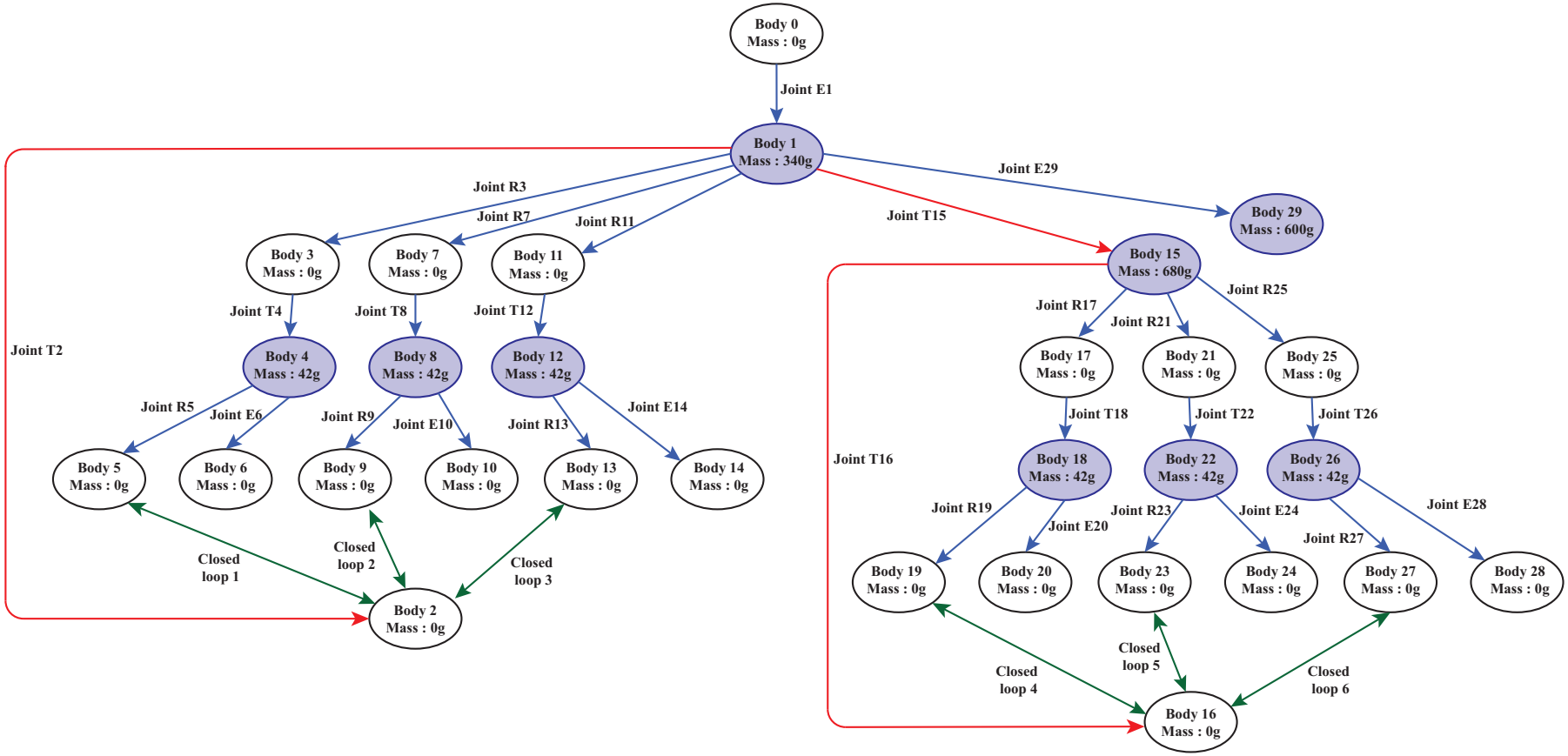


Figure 2.6 – Representation of components of the robot and their masses for the dynamic force model [BHPC16]

Some of the common methods to calculate the dynamic forces are the Lagrange approach and the Newton-Euler approach. In the Lagrange approach, the torques are computed based on the difference between kinetic and potential energy. The equation for computing the torque is given by [Kha10]:

$$\tau = \mathbf{I}(q)\ddot{q} + \mathbf{H}(q, \dot{q}) \tag{2.15}$$

In Eq. (2.15), $\mathbf{I}$ is the inertia matrix of the robot and $\mathbf{H}$ represents the Coriolis, centrifugal and gravity forces/torques. The problem with this approach is when the robot has higher number of degrees of freedom, the computational costs are very high and they consume a lot of time. Khalil proposed the recursive Newton-Euler algorithm (NE) [Kha10] for rigid robots which has been efficient with lesser computational times. Two algorithms are defined within the recursive NE viz: Forward and Backward recursions. In the forward recursion, the link velocities, accelerations and wrench on each links are estimated from link 1 to n. In the backward recursion, the reaction wrenches are estimated from link n to the robot base. The NE algorithm helps in estimating the torques by not necessarily computing the $\mathbf{I}$ and $\mathbf{H}$ terms of the Lagrange equation and the torque is given by:

$$\tau = f(q, \dot{q}, \ddot{q}) \tag{2.16}$$

The homogenous transformation matrix ${}^{i}\mathbb{T}_{j}$ as a function of the six geometric parameters (γ_j, b_j, α_j, d_j, θ_j, r_j) is given by:

$${}^{i}\mathbb{T}_{j} = \begin{bmatrix} {}^{i}\mathbf{R}_{j} & {}^{i}\mathbf{r}_{j} \\ 0_{1\times3} & 1 \end{bmatrix} \tag{2.17}$$

In Eq. (2.17), ${}^{i}\mathbf{R}_{j}$ represents the (3×3) rotation matrix and ${}^{i}\mathbf{r}_{j}$ represents the (3×1) position vector. In the forward recursion [Kha10], the velocity components can be estimated by the equation:

$$\eta_j = \mathrm{Adj}({}^{i}\mathbb{T}_{j})\eta_i + \dot{q}_j {}^{j}\mathbb{A}_{j} \tag{2.18}$$

The vector η_j is a kinematic screw vector of frame j of size (6×1) which consists of the linear and angular velocity components. The equation is given by:

$$\eta_j = \begin{bmatrix} {\mathbf{v}_j}^{T} & {\omega_j}^{T} \end{bmatrix}^{T} \tag{2.19}$$

The acceleration component is obtained by differentiating Eq. (2.18) and it is given by:

$$\dot{\eta_j} = \mathrm{Adj}({}^{i}\mathbb{T}_{j})\dot{\eta_i} + {}^{j}\zeta_j + \ddot{q}_j {}^{j}\mathbb{A}_{j} \tag{2.20}$$

$$\text{where } {}^{j}\zeta_j = \begin{bmatrix} {}^{j}\mathbf{R}_i[{}^{i}\omega_i \times ({}^{i}\omega_i \times {}^{i}\,\mathbf{r}_j)] + 2\sigma_j({}^{j}\omega_i \times \dot{q}_j {}^{j}a_j) \\ \bar{\sigma_j} {}^{j}\omega_i \times \dot{q}_j {}^{j}a_j \end{bmatrix}$$

The component ${}^{j}\mathbb{A}_{j}$ which is of size (6×1) is called as the transposition vector for velocities and

accelerations and it is given by:

$$^{j}\mathbb{A}_{j} = \begin{bmatrix} 0 & 0 & \sigma_j & 0 & 0 & \bar{\sigma_j} \end{bmatrix}^{T} \tag{2.21}$$

For a revolute joint, $\sigma_j = 0$ and for a prismatic joint σ_j=1. The acceleration values are then used in the dynamic wrench equation which consists of forces and moments. The forward recursive equations are thus given by:

$$^{j}\mathbb{F}_{j} = \begin{bmatrix} ^{j}\mathbf{F}_{j} \\ ^{j}\mathbf{M}_{j} \end{bmatrix} = \ ^{j}\mathbb{M}_{j}\dot{\eta_j} + \beta_j \tag{2.22}$$

$$\text{where } ^{j}\mathbb{M}_{j} = \begin{bmatrix} \mathbf{M}_j\mathbf{I}_3 & -^{j}\mathbf{M}\hat{\mathbf{S}_j} \\ ^{j}\mathbf{M}\hat{\mathbf{S}_j} & ^{j}\mathbf{J}_{j} \end{bmatrix} \ \& \ \beta_j = \begin{bmatrix} ^{j}\omega_j \times (^{j}\omega_j \times^{j} \mathbf{MS}_j) \\ ^{j}\omega_j \times (^{j}\mathbf{J}_j \ ^{j}\omega_j) \end{bmatrix}$$

The terms $\mathbf{M}_j$, $\mathbf{MS}_j$ and $^{j}\mathbf{J}_j$ are the inertial components of link j. Using the reaction forces and torques, the backward recursion is carried out. The torque induced on the motor is finally calculated by the equation:

$$\tau_j = {^{j}\mathbb{F}_{j}}^{T} \ ^{j}\mathbb{A}_{j} + F_{sj}\text{sgn}(\dot{q_j}) + F_{vj}\dot{q_j} \tag{2.23}$$

In Eq. (2.23), F_{sj} and F_{vj} represents the Coulomb and viscous friction parameters. These forward and backward recursive equations are implemented in MATLAB for estimating the torques and forces induced on the actuators under locomotion.

2.4 Results of force models

In reality, inside an industrial pipeline, there exists many bends or curves or deposits as demonstrated in Figure 1. The material used for pipeline elbows are usually stainless steel of grade Z401R and the connecting elements are stainless steel of grade Z2 CN18 10 [BHPC16]. These materials are non-magnetic and due to continuous operation of facilities, there exists deposits on the inner walls of the pipeline (Figure 1e). For the numerical simulations, a straight pipeline of 2.5 m length with inner diameter range from 54 to 74 mm is considered. A Maxon motor GP 16 S (ϕ 16) spindle drive unit coupled with an EC-max (ϕ 16) brushless motor [Max] in used in each individual module of the robot for actuation. The spindle drive offers a maximum static axial load of 500 N. The key specifications of the spindle drive and the DC-motor unit are provided in Tables 2.2 and 2.3. The optimized dimensions of slot-follower leg mechanism is determined with respect to the spindle drive used for actuation in [HCP+14] by using Genetic algorithm [URCCW10] in MATLAB. The objective functions aims at minimizing the space occupied by the legs of the robot and maximizing the force transmission factor. The optimized dimensions for the legs used in the robot are provided in Table 2.4. Using the spindle motors, the leg mechanisms, coupling screws and standard fasteners such as circlips, studs, nuts

Parameters	**Value**
Nominal voltage	12 V
Nominal current (I_c)	0.456 A
No load speed	13500 rpm
No load current (I_n)	60.2 mA
Nominal speed	5840 rpm
Efficiency (η_m)	54 %
Torque constant (K_T)	7.82 mNm/A
Speed constant (N_v)	1220 rpm/V
Terminal resistance	15.7 Ω

Table 2.2 – Specifications of the Maxon EC-motor (Part No. 283828) [Max]

Parameters	**Value**
Spindle screw diameter and pitch (p)	ϕ5 x 2 mm
Reduction ratio (G)	455:1
Max. static axial load	500 N
Max. spindle length	102 mm
Efficiency (η_s)	63 %

Table 2.3 – Specifications of the Maxon Spindle drive (Part No. 424749) [Max]

Parameters	**Dimensions** (mm)
l_1	57
l_2	7
e	11
ρ	8.5–45.5

Table 2.4 – Optimized dimensions of the slot-follower leg mechanism [CVB18].

and bolts, the entire robotic system is assembled. For the electronic unit, the ESCON 36/3 EC servo controller system offered by Maxon is used, which provides speed and torque control. The entire locomotion sequence is controlled by this control unit over the DC-Motors and the spindle drives through the umbilicus (robot cables). The digital model of the robot is realized using CATIA software and it is represented in Figure 2.7. The digital model of the robot and a simple simulation in CATIA can be found in the link provided at the bottom of this page [1].

2.4.1 Results of Static Force Analysis

The actuator forces of front and rear modules for horizontal and vertical orientations of pipeline are determined. With the help of Maple, the vector equations of the tangential radial

1. Digital model of the robot and a simple simulation in CATIA : Click here

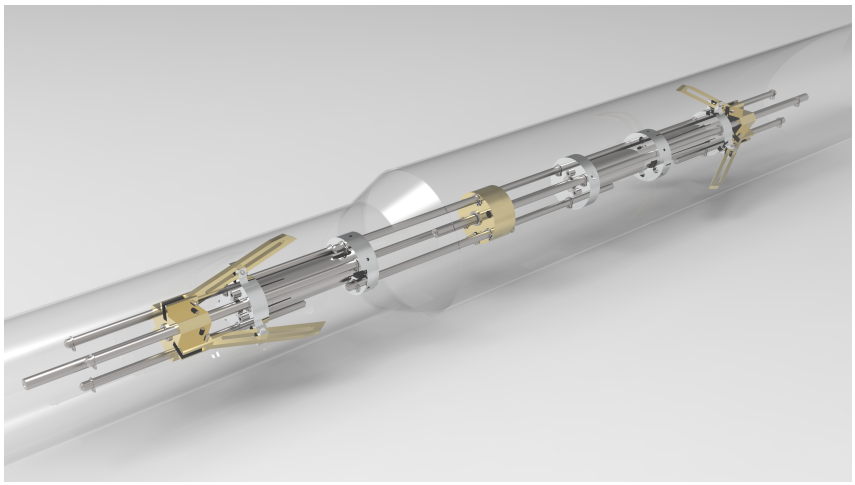


Figure 2.7 – Digital model of the robot in CATIA software [CVB19]

and longitudinal forces are extracted and they are given by:

$$\mathbf{t}_{li} = [T_{l1}, T_{l2}, T_{l3}]^T = \begin{bmatrix} \frac{-2l\cos(\delta)w\sin(\theta)}{r} \\ \frac{-5\left(\sqrt{3}\,\sin(\theta)+3\,\cos(\theta)\right)l\cos(\delta)w}{26r} \\ \frac{-5\left(\sqrt{3}\,\sin(\theta)+3\,\cos(\theta)\right)l\cos(\delta)w}{26r} \end{bmatrix} \tag{2.24}$$

$$\mathbf{t}_{ri} = [T_{r1}, T_{r2}, T_{r3}]^T = \begin{bmatrix} \frac{-2w}{3} \\ \frac{w}{3} \\ \frac{w}{3} \end{bmatrix} \tag{2.25}$$

In Eqs. (2.24) and (2.25), w indicates the self-weight of the robot which is around 6.57 N without taking into account the cables. Two static phases are possible during the locomotion cycle of the robot. The first case is when one set of legs is clamped with pipeline walls and the central actuator remains in a fully retracted phase. The second case is when one set of legs is clamped with pipeline walls and the central actuator reaches its full extension. The value of l, which is essential to estimate the moments induced at the clamped ends can be identified by the CG positions of the robot for the two cases of central actuator position from CATIA software. The values of l for the two cases are identified for 54 and 74 mm diameter pipelines and the corresponding values are provided in Table 2.5. For the estimation of forces, it is assumed that the pipeline is either horizontal ($\delta = 0$ or π radians) or vertical ($\delta = \pi/2$ or $3\pi/2$ radians) because the robot is a rigid model. The longitudinal and radial forces are estimated using Eqs. (2.24) and (2.25) for the two orientations of pipeline and the four CG positions in Table 2.5. Followed by this, using Eqs. (2.11) to (2.13), the normal forces as well as the actuator forces F_a can be estimated. The value of η_f for the two diametrical ranges under study can be taken from Figure 2.8 which is plotted with the help of Eq. (2.14).

Position of Central actuator	**Radius of the pipeline** (mm)	**Position of CG from clamped end** (mm)
Fully retracted	27	128
Fully extended	27	149
Fully retracted	37	123
Fully extended	37	144

Table 2.5 – CG position (from CATIA) from the clamped legs of robot for fully retracted and extended positions of central actuator inside 54 and 74 mm diameter pipelines [CVB18]

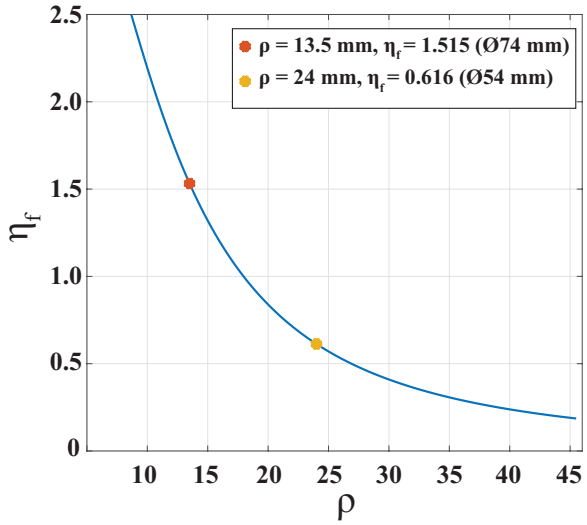


Figure 2.8 – Force transmission factor (η_f) with respect to ρ

The results of the actuator forces for the two orientations of pipeline and four CG positions are then estimated using the values from Table 2.5 and Figure 2.8. The results are represented in Figure 2.9. As shown in Figure 2.9, the forces are affected directly by the position of CG and the orientation of pipeline. For horizontal orientation of pipeline, varying range of actuator forces could be observed for both diameters. This is caused mainly by the longitudinal force component, which is a function of the orientation of robot (θ). When the pipeline is vertical, no moment exists, as the longitudinal forces depends on the cosinus of the pipeline orientation (δ), which becomes zero at $\pi/2$ and $3\pi/2$ radians. On the contrary, the radial forces continues to exist and they always remain a constant (T_{r1}=4.3 N, T_{r2}= T_{r3}=-2.3 N), as they depend only on the self-weight w of the robot. This phenomenon could be observed on the plots for vertical orientations of the pipeline. For a given diameter, the highest amount of force is observed with the fully extended position of the central actuator for δ at 0 or π radians. As the actuator force F_a depends on the factor η_f, which is a function of the pipeline radius, higher forces are observed for the 74 mm diameter pipeline. The force range values for Figure 2.9 are provided in Table 2.6.

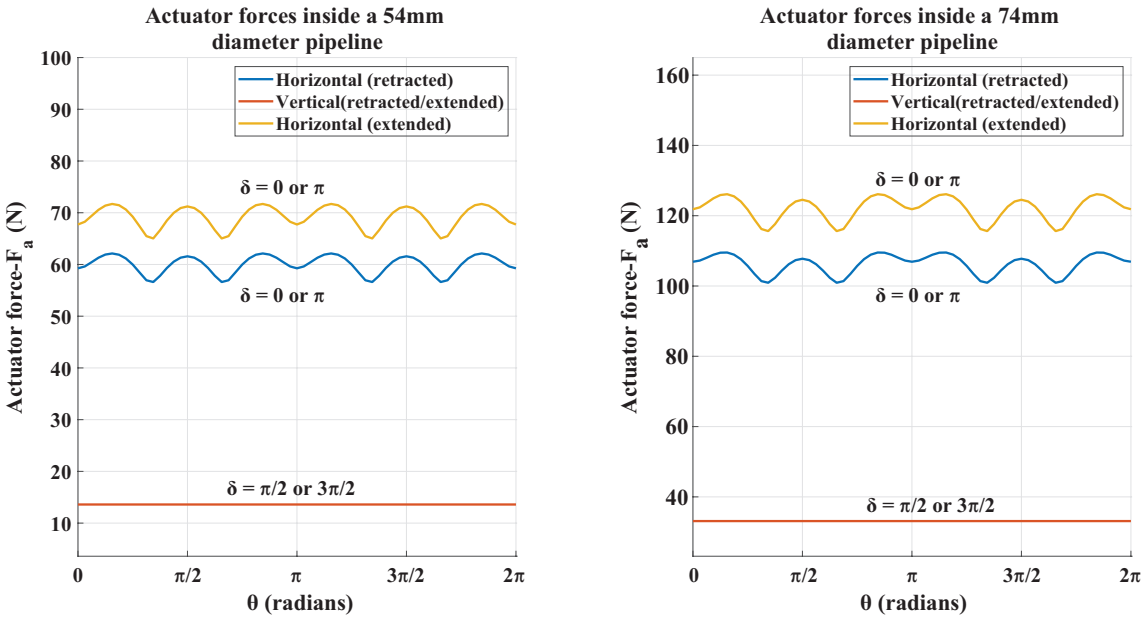


Figure 2.9 – Front/Rear module actuator forces F_a inside 54 mm (left) and 74 mm(right) diameter pipelines for horizontal (δ= 0 or π radians) and vertical (δ= $\pi/2$ or $3\pi/2$ radians) orientations

Position of Central actuator	**Pipeline radius** (mm)	**Actuator force** (N) **Horizontal pipe**	**Actuator force** (N) **Vertical pipe**
Fully retracted	27	57.6–62.1	13.6
Fully extended	27	66.5–71.7	13.6
Fully retracted	37	101.9–109.5	35.1
Fully extended	37	117.6–126.1	35.1

Table 2.6 – Front/Rear module actuator forces in Figure 2.9 inside horizontal and vertical orientations of pipeline for the CG positions of Table 2.5

2.4.2 Results of Dynamic Force Analysis

The simulation of the locomotion of the robot is performed using MATLAB. For motion planning, three profiles are taken into study. The law of Bang-Bang is faster but it is discontinuous in acceleration. In order to overcome this issue, the Trapezoidal law can be used, which ensures a constant velocity profile where the acceleration becomes zero [KD04]. In order to have better continuity of the profile in velocity and acceleration, the polynomial equation of order 5 is employed. The velocity and acceleration profiles are given by the equations [KD04]:

$$\dot{q}(t) = 30\left(\frac{t^2}{t_f^3}\right) - 60\left(\frac{t^3}{t_f^4}\right) + 30\left(\frac{t^4}{t_f^5}\right) \tag{2.26}$$

$$\ddot{q}(t) = 60\left(\frac{t}{t_f^3}\right) - 180\left(\frac{t^2}{t_f^4}\right) + 120\left(\frac{t^3}{t_f^5}\right) \tag{2.27}$$

In Eqs. (2.26) and (2.27), t_f is the total simulation time. This value is not chosen arbitarily but instead fixed in such a way that the velocity and acceleration reach their saturation points [KD04]. The acceleration equation, Eq. (2.27) is used to estimate the forces and torques induced on the actuators. The results of displacement, velocity and acceleration for the three actuators are represented in below in Figure 2.10.

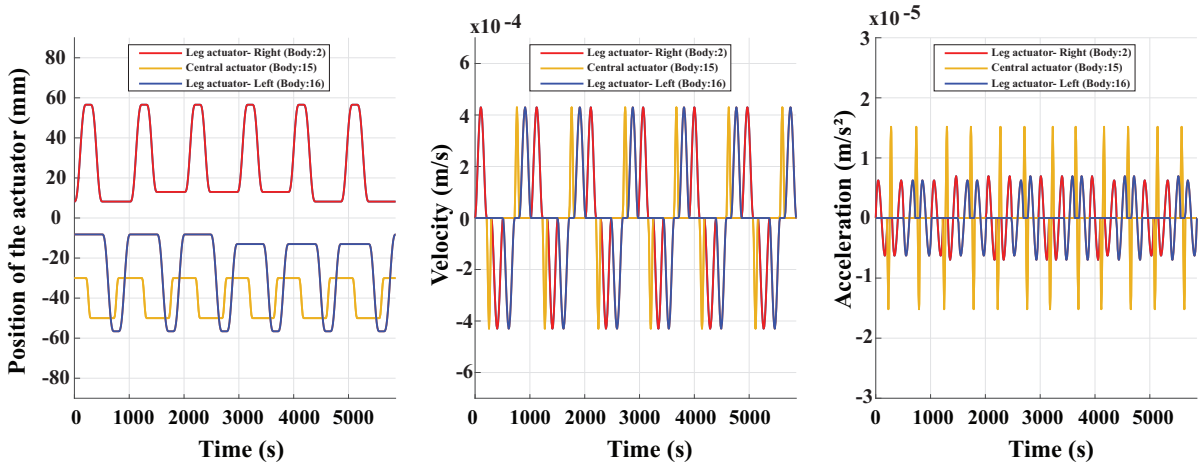


Figure 2.10 – Displacement, Velocity and Acceleration profiles of the actuators during a locomotion cycle

The blue and red lines represent the left and right leg actuators. Inside a 74 mm diameter pipeline, they have an extension of 40 mm but when they encounter a diameter change, the extension is found to be reduced to 35 mm. This could be observed easily on the displacement plots. The central actuator (Orange line), irrespective of the diameter change extends always to a distance of 20 mm as they have no clamping modules attached to them. The maximum velocity attained by the system is found to be 0.43 mm/s, which is caused by the reduction unit used in the Maxon motors. The output of the acceleration as well as the velocity is then employed in the NE algorithm in MATLAB to estimate the motor forces when traveling inside horizontal and vertical pipelines. The results are computed and represented in Figures 2.11 and 2.12 for Model-A and in Figures 2.13 to 2.14 for Model-B. From the results, it could be observed that in contrast to the static model, higher actuator forces are observed in the central leg actuator while moving inside a vertical pipeline. This is because, in a vertical pipeline, the effect of gravity is much higher when compared to the horizontal orientation. The constant lines on the curves indicate the presence of gravity and the peaks represent the dynamic effects caused by the moving masses. The actuator forces on both the leg modules remain the same but appear in the opposite way for both models. In both models, the central actuator force range appears to be the same and is close to 0 N when the pipeline is horizontal. In the case of vertical orientation of the pipeline, the actuator forces on the leg modules remain the same for both the models like the horizontal pipeline. However, the forces on the central actuator are higher for the vertical orientation. For Model-A, these forces range between 8.07–8.1 N. However, for

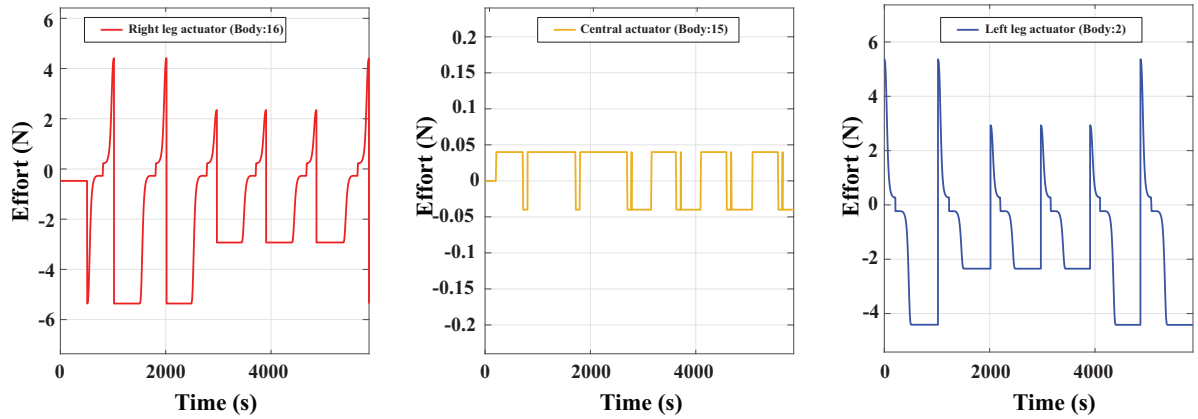


Figure 2.11 – Forces induced on the right leg actuator (left), central actuator (middle) and left leg actuator (right) inside a horizontal pipeline for Model-A

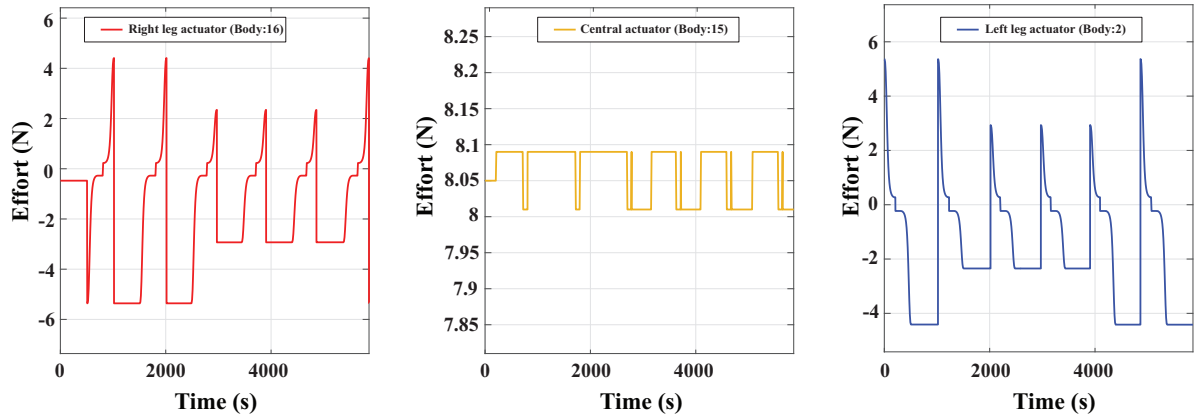


Figure 2.12 – Forces induced on the right leg actuator (left), central actuator (middle) and left leg actuator (right) inside a vertical pipeline for Model-A

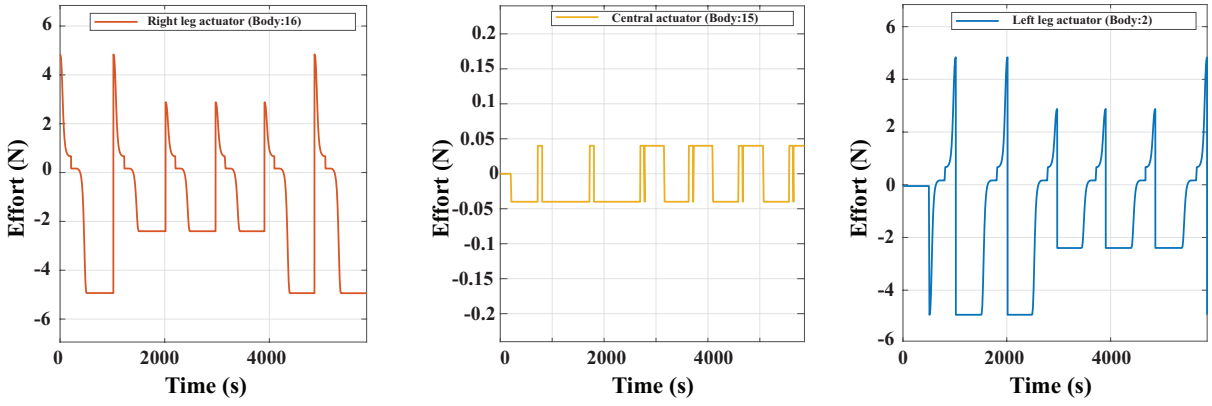


Figure 2.13 – Forces induced on the right leg actuator (left), central actuator (middle) and left leg actuator (right) inside a horizontal pipeline for Model-B

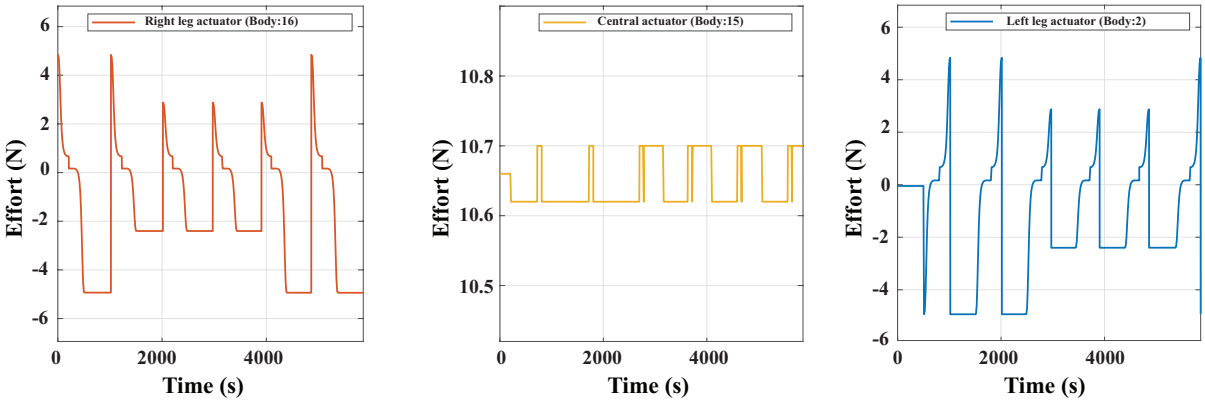


Figure 2.14 – Forces induced on the right leg actuator (left), central actuator (middle) and left leg actuator (right) inside a vertical pipeline for Model-B

Model-B, the central actuator forces range between 10.66–10.7 N. This is mainly caused by the presence of the weight of the umbilicus along gravity for Model-B. Based on simulation results, it has also been observed that when the pipeline has an orientation of π radians, the central actuator forces remain the same as inside a pipeline which is at 0 radian for both the models. Also, the forces remain the same for $\pi/2$ and $3\pi/2$ radians pipeline orientations. The forces on the left leg and the right leg actuators tend to have the same range of values irrespective of the orientation of the pipeline for both the models and they are affected only by the radius of the pipeline.

2.4.3 Discussions

From the results of both the force models, it could be observed that the static forces on the actuators of the leg mechanisms are comparatively higher than the dynamic model. One of the major factors is the low velocity of the motors used in each module. The dynamic forces are purely influenced by the velocity as the time derivative which provides the acceleration is employed to determine the motor forces and torques during locomotion. These forces appeared to be around 10 times smaller than the static actuator forces of the leg mechanisms. However, with the dynamic model, it was possible to identify the forces on the central actuator, which could not be identified in the static force model. The results coming from static force models will be employed for the experimental validation of the prototype for achieving static clamping phases. It must be noted that the static force model did not take into account the weight of the cables and other electromechanical factors.

2.5 Experimental validation of the prototype

In this section, experimental validation is done on the prototype of the bio-inspired robot developed at LS2N. In the previous sections, it was possible to identify the actuator forces

through static and dynamic models by considering two models. Besides, it was possible to evaluate the performance of the motors, which is necessary to achieve the robot's mobility. Generally, inspection robots employ one or more electronic boards for their control and movements when moving through various trajectories of pipelines. Some of the well-known devices are the Raspberry Pi [UH14], Arudino [Smi17] and BeagleBone (BB) black [**?**]. The choice of these boards or micro-computers purely depends on the intended application. Leccese et al. [LCT14] studied on the comparison of these three micro-computers. Arduino is usually used for programming with low-level C++ languages and it can also be used to create an interface between sensors. An interesting example is the vibro-impact capsule type inspection robot prototype of Yusupov et al. [YL16] where an Arduino board is used to propel the robot inside a Poly-vinyl chloride(PVC) test pipe. Raspberry Pi and BB Black, on the other hand, works efficiently for high-end programming languages such as Python and MATLAB. These micro-computers operate on Linux platform with access to the internet. Raspberry Pi is intended for applications with GUI and cannot be compatible for interfacing with external sensors. However, it has compatibility for camera interfacing which makes it efficient for moving inside dark pipelines. The robot of Abdellatif et al. [AMM+18] employs a Raspberry Pi for robot control and detection of cracks inside a dark environment. A BeagleBone (BB) black is employed for the control of the rigid bio-inspired robot. The peak results of a given pipeline diameter of the static force model are employed in the control algorithm of the robot to attain static phases such as clamping or de-clamping during the locomotion. The motion profile of the locomotion cycle is similar to a "bang-bang" concept [KD04], which is optimal with respect to the execution times. In parallel, it is possible to reconstruct the current position of the robot by knowing the travel time and speed.

2.5.1 Interfacing the robot with BB black and ESCON controllers

The BB black offers 92 pins which comprise of the Pulse-Width Modulation (PWM), General-Purpose Input/Output (GPIO), Analog to Digital converters (ADC) and reconfigurable digital ports [BB]. Each Maxon motor is interfaced with the BB black through their ESCON 36/3 servo controller by three pins. One pin is connected to the PWM which is used to operate the duty cycle of a motor in a particular direction. The other pin is connected to a GPIO which is used to enable or disable the motor functioning in a Boolean way. The third pin is connected to the ADC to read the motor voltage. The PWM of the BB provides a maximum voltage of 3.3 V and it is used to operate the motors with control over the speed. For the idle stage, 50% duty cycle is set as a reference where the current is 0 A. For the safe running of motors, 80% of the duty cycle of the PWM is only used. At 80% duty cycle the motor performs either clamping or retraction operation with clockwise rotation and at 20% duty cycle the motor performs de-clamping or elongation operation with anti-clockwise rotation. The interfacing between BB black and one set of EC motor and its controller is provided in Figure 2.15. In Figure 2.15, the **J2** cable of each ESCON controller is connected to its EC motor unit. The **J1** cable of each controller is

connected to a power source of 12 V. The cables **J5** and **J6** are connected to the PWM, GPIO and ADC pins of BB black. The PWM, GPIO and ADC pin address on BB black for each motor and its servo controller is listed below in Table 2.7.

Motor	**PWM (Duty cycle)**	**GPIO (Enable/Disable)**	**ADC (Results)**
Motor-1	P9_14 (DigIN 1)	GPIO_31 (DigIN 2)	AIN0 (AnOUT 1)
Motor-2	P8_13 (DigIN 1)	GPIO_47 (DigIN 2)	AIN1 (AnOUT 1)
Motor-3	P9_31 (DigIN 1)	GPIO_68 (DigIN 2)	AIN2 (AnOUT 1)

Table 2.7 – BB black pin address for each motor and its servo controller

Before the start of experiments, the DC motors and ESCON controllers are calibrated in a computer using a USB cable through port **J7** as per the specifications provided in Tables 2.2-2.3. The output voltages from PWM which are generally analog is passed to the ADC to display the outputs to the user. The ADC of BB black, however, supports a maximum voltage of 1.8 V. Thus, the output analog voltage before passing to ADC of BB black is calibrated in the ESCON software in accordance with the nominal current of the motor. The BB black also offers Enhanced Quadrature Encoder Pulse (eQEP) pins which can be used to determine the position/displacement of robot at any instance of locomotion. However, this feature is not included in the current control loop of the robot.

2.5.2 Control methodology: Force control

Two algorithms can be classified for the control of a robot namely: Displacement control and Force control. The former is used for a robot that works in an open environment and moves through a known distance. When the diameter of any instance of a locomotion step is unknown, the force control algorithm is employed where through voltage control, the necessary force is applied to the motors in order to clamp the robot legs with pipeline walls. The force control algorithm is employed for the bio-inspired robot and this algorithm is used to control the locomotion of the robot inside an environment having unknown diameters. By applying the necessary clamping force from the power source to the motors, the robot can adapt its legs and clamp itself to pipeline walls. Initially, interfacing between BB black and each motor as well as its servo controller is done as per Figure 2.15. The BB black is connected to a computer and this board operates on Debian OS 7.9. The activation of pins specified in Table 2.7 is done using the Linux terminal of BB black micro-computer. When a PWM duty cycle is applied to the motor, the spindle drive enables the leg mechanisms to clamp or de-clamp or allow the central motor to elongate or retract. The motors rotate at a constant speed during its duty cycles. A simpler form of the pseudo-code for one forward motion is provided in Algorithm 1 [2].

2. BB black pin activation and control of the bio-inspired robot written in C : GitHub link

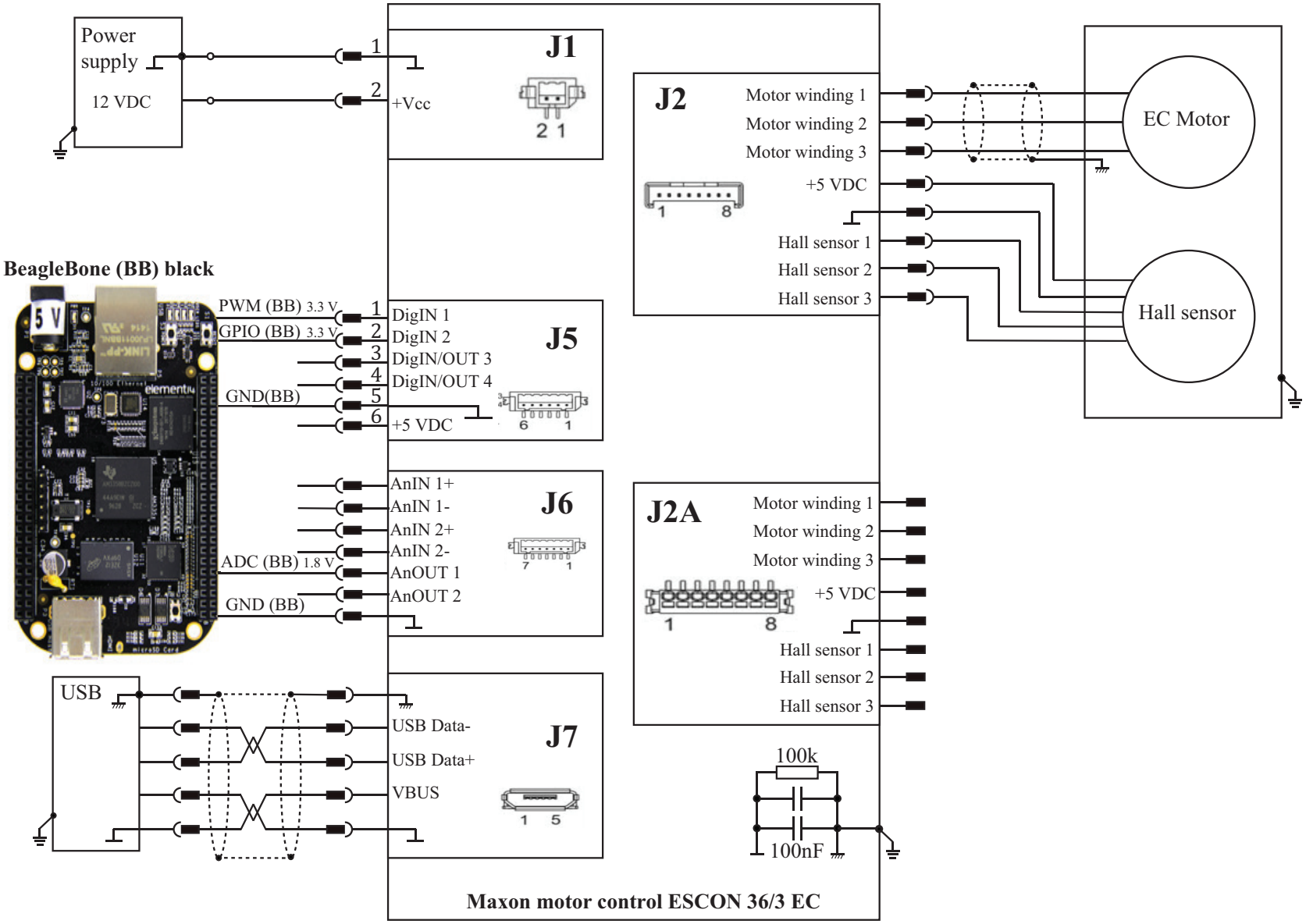


Figure 2.15 – Layout of the interface between BB black to one EC-motor and its servo controller

At the end of one loop, the robot remains in an over-constrained posture wherein both the leg mechanisms are clamped with the walls of the pipeline. The prototype of the robot developed at LS2N, France is represented below in Figure 2.16.

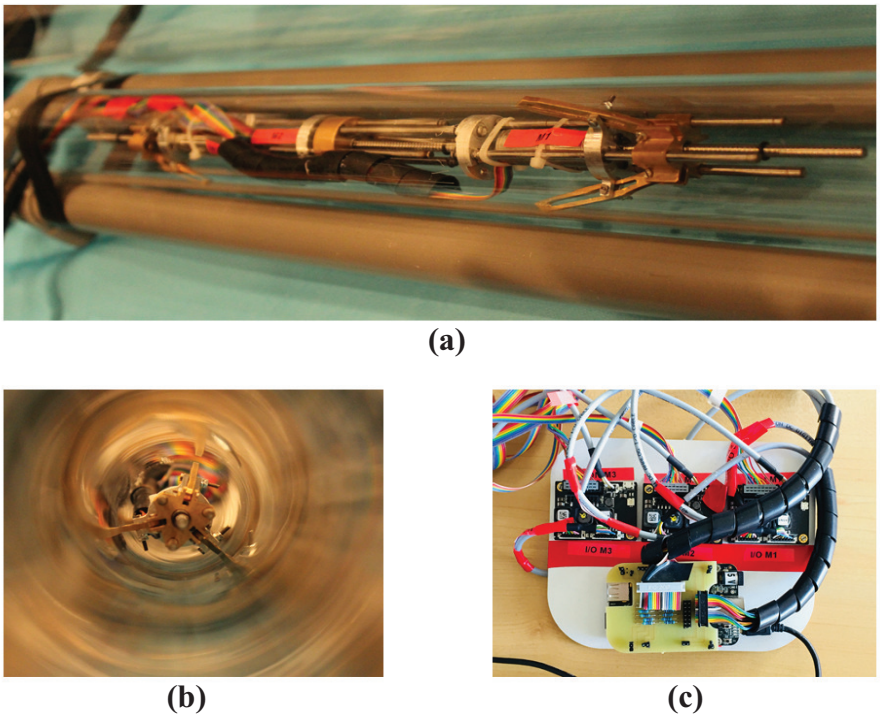


Figure 2.16 – Representation of the (a) Robot inside a test pipeline , (b) side view of the robot and (c) one of the three ESCON 36/3 servo controllers

The following operations are performed using C program for the locomotion of the robot, which is written as per Algorithm 1:

- Forward motion (twice)
- Reverse motion (twice)

In order to ensure that there is no continuous running of motors after an operation is finished, threshold limits are being set for the clamping/retraction as well as de-clamping/elongation phases. The upper threshold T_U is set at 1.5 V which is 83% of the maximum voltage of ADC and the lower threshold is set at 0.42 V. These values are determined based on experimental trials. Also, these values are ideal when operating inside variable sections as the diameter of any locomotion step is unknown. When the limits are reached, the torque reaches its stall point and the PWM is set to the idle phase.

2.5.3 Results of experiments for a locomotion cycle

After the interfacing between controllers and BB black is done, the parameters defined in Table 2.8 are calibrated for the ESCON controllers before the algorithm is initiated.

Algorithm 1 Forward locomotion sequence: Force control algorithm

Initialization
$Hyper - static\ posture$
while $ADC\ of\ M1 < T_U$ **do** ▷ T_U- Upper threshold: Operation of motor till stall torque
    Front leg modules clamping: M1 rotation at constant speed
end while
while $ADC\ of\ M2 < T_U$ **do**
    Central module retraction: M2 rotation at constant speed
end while
while $ADC\ of\ M3 < T_U$ **do**
    Rear leg modules clamping: M3 rotation at constant speed
end while
$Advancement\ of\ robot$ ▷ Two forward motions ensured by a for loop
for $i = 1 - 2$ **do**
    while $ADC\ of\ M1 > T_L$ **do** ▷ T_L- Lower threshold: Operation of motor till stall torque
        Front leg modules de-clamping: M1 rotation at constant speed
    end while
    while $ADC\ of\ M2 > T_L$ or $(t < 2s)$ **do** ▷ Estimation of stroke as a function of speed
        Central module elongation: M2 rotation at constant speed
    end while
    while $ADC\ of\ M1 < T_U$ **do**
        Front leg modules clamping: M1 rotation at constant speed
    end while
    while $ADC\ of\ M3 > T_L$ **do**
        Rear leg modules de-clamping: M3 rotation at constant speed
    end while
    while $ADC\ of\ M2 < T_U$ **do**
        Central module retraction: M2 rotation at constant speed
    end while
    while $ADC\ of\ M3 < T_U$ **do**
        Rear leg modules clamping: M3 rotation at constant speed
    end while
end for
return $ADC\ values$

PWM Duty	**ADC voltage** (V)	**Nominal current** (A)	**Motor speed** (rpm)
20%	0	-0.46	10800 (Counter-clockwise)
50% (idle)	0.9	0	0
80%	1.8	0.46	10800 (Clockwise)

Table 2.8 – ESCON controller calibration before initiation of Force-control algorithm

The experiment is carried out inside a 2.5 m length PVC transparent tube of 74 mm diameter as depicted in Figure 2.16a. The digital voltage values generated by the ADC of the BB are extracted and the results are plotted with the help of MATLAB. With the voltage values obtained

during each step of the experiment, the current induced on the actuators can be estimated by the equation:

$$I = \frac{V_t I_c}{V_{idle}} - I_c \tag{2.28}$$

In Eq. (2.28), V_t indicates the voltage at a particular instant of time and I_c is the nominal current of motor which is provided in Table 2.2. V_{idle} is the ADC voltage at 50% PWM duty cycle, which is taken as 0.9 V from Table 2.8. However, the output power of the DC-Motor is affected by the power loss generated due to the current and resistance of the windings. The actual current generated from the motor is thus given by the equation:

$$I_a = I - I_n \tag{2.29}$$

In Eq. (2.29), I_n is the no-load current and this value is subtracted from Eq. (2.28) to compensate the power loss. With the help of Eq. (2.29), the output torques and corresponding forces can be calculated. The actual current generated from motors after power loss consideration is shown below in Figure 2.17.

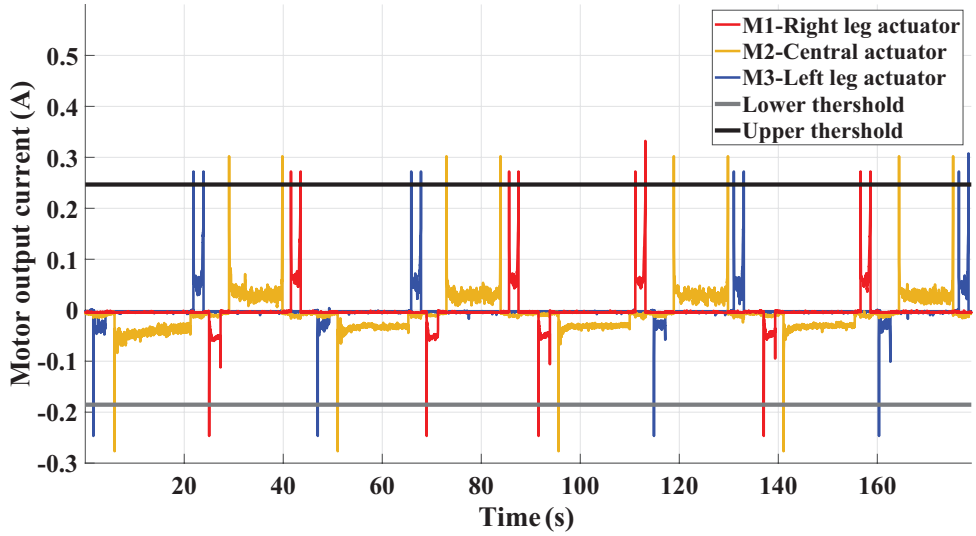


Figure 2.17 – Output current of EC-motors from experiments.

In Figure 2.17, the red lines indicate the front leg module, the orange line indicates the central elongation module and the blue lines indicates the rear leg module. The threshold voltages are indicated in black and gray lines. The total time for performing two forward and reverse operations by the robot is around 179 s. At 89.5 s, two forward operations are completed and at 179 s, the robot returns to the starting point. Higher noises were observed from the output voltages generated by the ADC. This is caused by some bending phenomena observed on the free end of the robot when the other end is clamped and also due to numerical errors. The frequency of the results was measured using MATLAB. With the help of Savitzky–Golay [sgo] filtering technique in MATLAB, the frequency from the ADC output is matched with the

desired frequency to reduce the noise generated in the voltage. This filtering technique employs the linear least-squares technique for smoothening signals without distorting it. The output torques from the motors are then estimated from the smoothened current data by the equation:

$$\tau_{Mi} = I_a \; K_T \; G \; \eta_m \; \eta_s \quad \text{with} \quad i = 1, 2, 3 \tag{2.30}$$

In Eq. (2.30), the torque constant (K_T), reduction ratio (G) and the efficiencies (η_m and η_s) are taken from Tables 2.2 and 2.3, respectively. The operating force on the spindle drive of the motor can be estimated with the help of torque values obtained from Eq. (2.30) and it is given by:

$$F_{Mi} = \frac{2\pi\tau_{Mi}}{p} \quad \text{with} \quad i = 1, 2, 3 \tag{2.31}$$

In Eq. (2.31), p represents the screw pitch of the spindle drive, which is provided in Table 2.3. The output forces in the spindle drive can be calculated using Eq. (2.31) by performing the force control algorithm inside horizontal and vertical orientations of pipeline. The results are represented in Figures 2.18 and 2.19. Video link for the experiment performed on the bio-inspired robot is provided in the bottom of this page [3]. Since the velocity of the robot is very low (0.43 mm/s [CVB19]), the lower threshold limit is set to a value lower than the peak limits of static force model such that the de-claming phase happens for a shorter duration. This can be observed in Figures 2.17 to 2.19. As the static forces on central module is lower, this module extended/retracted to its maximum, which facilitated the robot to cover larger distances during the locomotion cycle. The range of forces in Figures 2.18 and 2.19 for the three individual actuators are provided in Table 2.9.

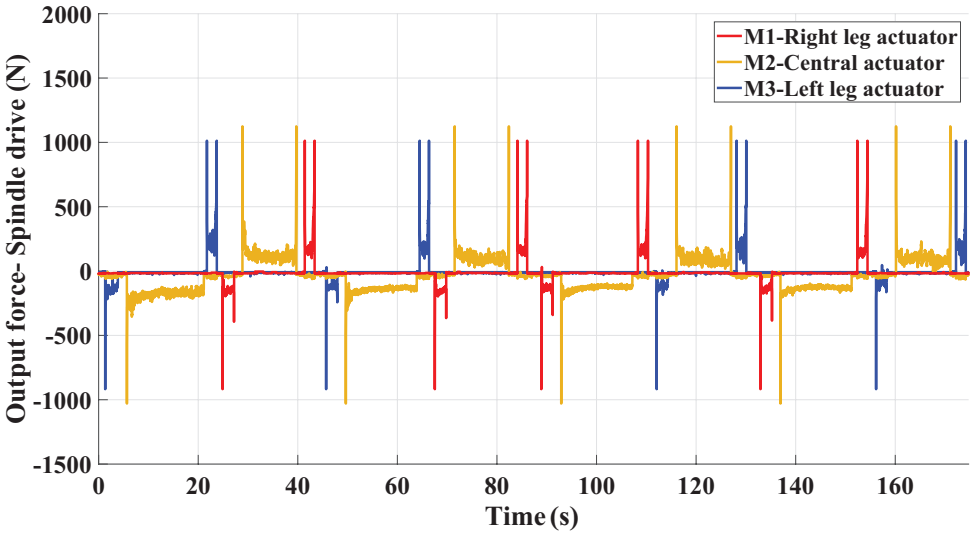


Figure 2.18 – Experimental output forces in the spindle drive for horizontal orientation (δ= 0 or π radians) of pipeline.

3. Video link for the experiment performed on the bio-inspired robot : Click here

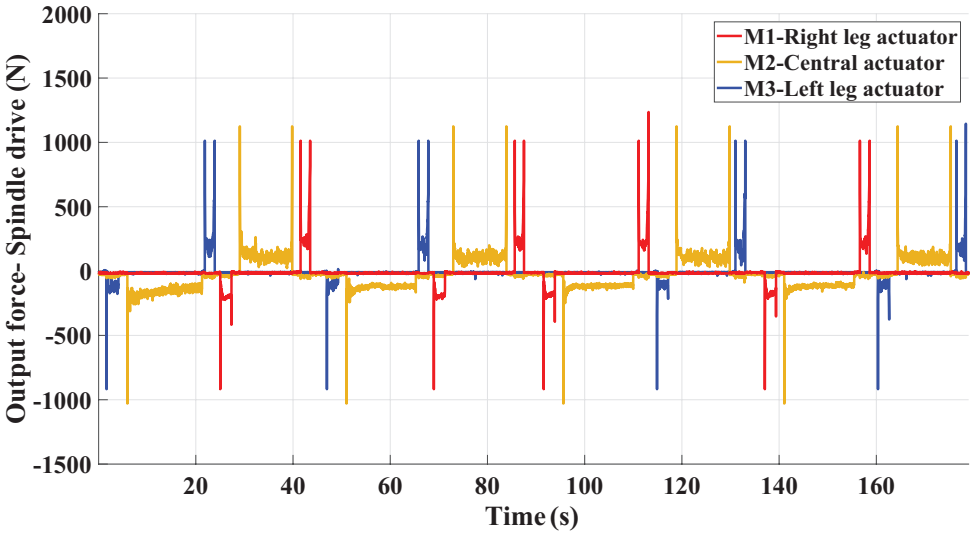


Figure 2.19 – Experimental output forces in the spindle drive for vertical orientation ($\delta = \pi/2$ or $3\pi/2$ radians) of pipeline.

Orientation of pipeline	**Phase**	**Initial force** (N)	**Final force** (N)	**Force under operation** (N)
Horizontal	Clamping (M1 & M3)	1000	1000	170–250
Horizontal	Declamping (M1 & M3)	900	0	80–150
Vertical	Clamping (M1 & M3)	1000	1000	160–330
Vertical	Declamping (M1 & M3)	900	0	100–180
Horizontal & Vertical	Elongation (M2)	1000	0	110–160
Horizontal & Vertical	Retraction (M2)	1100	1100	100–150

Table 2.9 – Experimental output forces of the spindle drive for Figures 2.18 and 2.19.

Higher forces of 1000–1100 N could be observed at the start of clamping, elongation and retraction modules for both orientations of the pipelines. This is because a constant velocity profile has been used in the algorithm, which contributed to higher starting torque when a PWM duty cycle is applied. The inertial effects also contributed to these peak values. At the end of clamping operation in motors M1 and M3, higher forces could be observed (Figures 2.18 and 2.19) over the de-clamping phases. This is caused by the leg masses and their flanges as they impose more loads on the motor when they try to establish contact with the walls of the pipeline. For the vertical orientation of pipeline, the forces under operation of clamping/de-clamping phases are found to be on the higher side over the horizontal orientation, especially in Motor M3. This is because, motor M3 is bolted along with M2 and during vertical pass, the influence of gravity impose more load on the motors, especially during the clamping phase. The umbilicus also contributes some forces significantly during vertical orientation of pipeline. It is interesting to note from the results of Figures 2.18 and 2.19 as well as Table 2.9 that the central

actuator forces almost remains the same for both orientations of the pipeline. However, under operation, the forces on the central actuator is slightly higher for the horizontal orientation during elongation phase because the robot behaves similar to the cantilever beam assumption of the static force model.

At the end of the clamping phase, the contact force F_p at the legs could be calculated with the help of Eq. (2.13). Considering a 74 mm diameter tube for the experiments, a clamping force of around 630 N is required to establish a tight contact between the legs of the robot and the PVC test pipe. These forces tend to exist on the motor until the de-clamping phase begins, where these forces become zero. The results are depicted below in Figure 2.20.

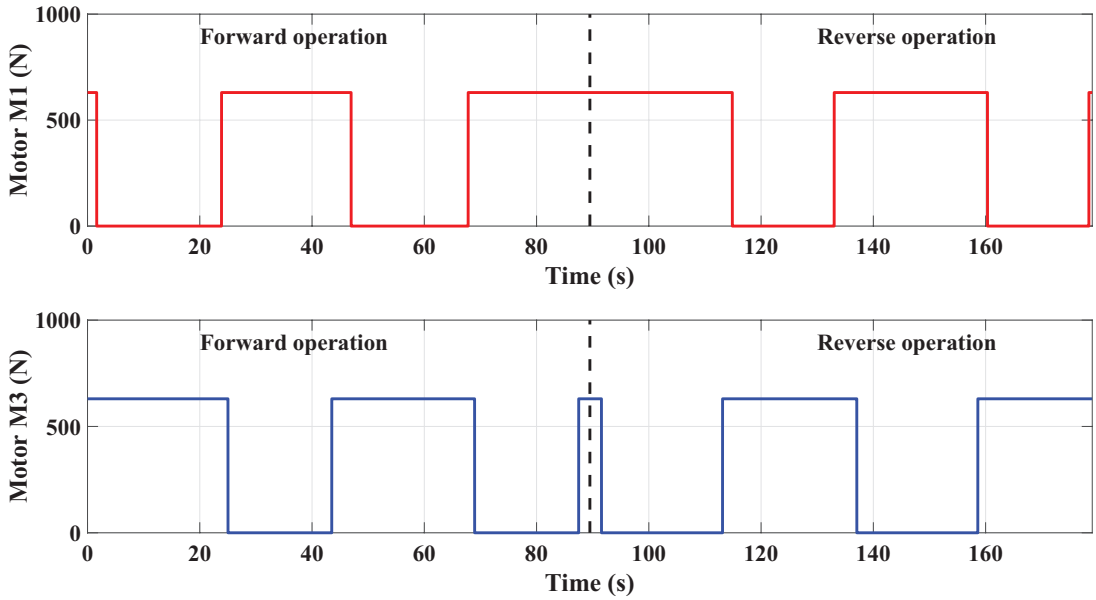


Figure 2.20 – Clamping forces (F_p) between legs of the robot and PVC test pipe from experiments.

In Figure 2.20, it could be observed that the clamping forces are higher when compared to the numerical model. Several factors contributes to this deviation. The numerical model considers a frictional contact between steel pipeline and bronze legs but the prototype was tested inside a PVC tube. Also, with the numerical model, factors such as the umbilicus weight and electromechanical parameters were not taken into account. These reasons could have potentially contributed to higher force values in the experiments over the numerical model.

2.6 Conclusions

In this chapter, the static and dynamic force models on a bio-inspired piping inspection robot proposed by Henry et al.[HCP+14]. Owing to the low velocity of the robot, a static model was sufficient to understand the clamping as well the static actuator forces of the robot. It was also possible to identify the key properties of motors that are required for the robot's locomotion

cycle. The peak results coming from the numerical model facilitated the control of static phases on the prototype during experiments. In the numerical analysis, it was not possible to validate the electromechanical factors or frictional coefficients. Using a prototype, these factors were examined and it was possible to isolate some differences from numerical analysis such as voltage fluctuations caused by the working condition of EC motors. The robot was able to overcome the issue of working inside pipelines having diameters less than 150 mm as well as with varying diameters. The experiments were conducted for a 74 mm diameter pipe, however, it is also possible to carry out within a 54 mm diameter, which can then be compared with the static model. The results of experiments proved to be on the higher side owing to factors that were not considered in numerical analysis . One major disadvantage of the robot is that it is a rigid model and it cannot be employed for curved pipe profiles represented in Figure **??**b to **??**d. The spindle drive unit used in each module of the robot is currently oversized as it has a screw length of 102 mm. This excess screw length might restrict the robot to overcome a pipeline having bends or curvatures if the robot is made flexible. In the upcoming chapter, the initial problem that will be addressed is the design modification of this robot into a flexible system.

CHAPTER 3

Design and analysis of Tensegrity mechanisms

Problem statement:

1. *What are the key design issues that must be addressed for the design of an articulation unit which can facilitate a robot to overcome pipe bends and junctions under passive and active modes?*
2. *What are the tilt limits for the mechanism?*
3. *What are the optimal design parameters of the mechanism for having a stable configuration under static modes?*

3.1 Introduction

This chapter focuses on the design modification of the rigid bio-inspired robot [VCB19] studied in Chapter-2 into a flexible system. The main design criteria for a piping inspection robot are locomotion, contact type with pipeline walls, cable management and articulation unit to pass through bends. Roh et al.[RC05] proposed a classification of in-pipe inspection robots viz: Pig type, Wheel type, Caterpillar type, Wall-press type, Walking type, Inchworm type and Screw type. This classification provides a glimpse of the contact type with the pipeline walls and the locomotion principle of the robot. However, detailed research on the design of an articulation system for bends and cable management inside complicated pipeline trajectories have not been carried out. This chapter addresses in detail the key design issues that must

be addressed by a piping inspection robot for passing through bends and junctions. Generally, articulation units or a mechanism could be classified viz: Active and Passive systems. Some existing researches that employ an active articulation unit include the THES-I robot of Hirose et al.[HOMS99] which uses a universal joint actuated by a DC-Motor and the electro-pneumatic robot of Anthierens et al.[ACB99] that employs a metallic bellow system to pass through bends. Some interesting passive systems include the robot developed by Kwon et al.[KLJY08], where a compression spring is employed to push and pull motor modules while working through T-sections and bends. Zhang et al.[ZZSJ10] also proposed a robot that uses a passive flexible helical axle to pass through pipe bends. Most of these robots employ articulation units that are either passive or active, but not a combination of both. A tensegrity mechanism, with three tension springs and a universal joint, is proposed as an articulation unit for the rigid bio-inspired robot studied in Chapter-2. While encountering a pipe bend at 90°, the tensegrity mechanism can work in a passive mode. In the event of a junction or a T-union, cables that pass through the springs of the mechanism are actuated to tilt along a certain direction. The geometrical equations for the tensegrity mechanism are generated by correlating to a parallel manipulator of type 3-S$\underline{P}$S-U. The analysis of the positioning of the end-effector is carried out in two ways. For the first analysis, the Tilt & Torsion (T&T) theory is employed to study the joint-limits of the mechanism through case study of three different postures. Based on the tilt limits, the best posture is identified and the static stability of this mechanism is verified. For the second analysis, the singularities and workspace analysis are carried out for the Euler angles of the universal joint. This analysis permits to determine the maximum tilt limits for the mechanism, especially under passive modes. Both analysis are carried out using the SIROPA library of Maple [JCB+18] in which the workspaces for the mechanism are calculated by algebraic methods [CMW11].

The architecture of the 3-S$\underline{P}$S-U tensegrity mechanism is then modified into a 4-S$\underline{P}$S-U mechanism by the addition of another tension spring since the tilt limits for the 3-S$\underline{P}$S-U mechanism were found to be less for operation during passive modes. The static analysis of both mechanisms is reverified for the Euler angles. Through an optimization technique, the design parameters of the system are modified to obtain a stable configuration under the presence of an external load. A simple prototype of the 3-S$\underline{P}$S-U mechanism is realized at LS2N with initial and modified design parameters for understanding the stability of the mechanism. The singularities and workspace analysis are carried out for the modified design parameters to understand the maximum tilt limits and a comparison is being made.

3.2 Design issues and architecture of the flexible robot

In general, for the design of a piping inspection robot, three main factors must be taken into account viz: rectilinear motion, passive compliance and active compliance. The existing architecture of the robot presented in Chapter-2 addresses the first criteria as it holds good inside vertical and horizontal pipelines. It is essential to address the other two factors for the

robot, in order to pass through bends and junctions. Passive compliance could be imagined for a robot that moves through pipe bends at 90° or less. The articulation unit, which drives the robot through such bends need not be necessarily active. Examples of such robots include the squirm pipe robot proposed by Zhang et al. [ZZSJ10] and the caterpillar type robot proposed by Kwon et al. [KLJY08]. For active compliance, it is essential for the articulation unit as well as the contact mechanism with the pipeline walls to be actuated and adaptable. Robots passing through T-branches and junctions need to adopt active compliance. However, the direction of movement inside a T-branch or junction is usually controlled through software programming and sensors. An interesting example is the robot proposed by Ryew et al. [RBR+00] which employs a double active universal joint (actuated by two motors and two clutches) that remains active while passing through junctions. This robot also uses a wheel system with a pantograph mechanism that adapts according to bends and straight profiles by ensuring contact with the pipeline walls. Another important factor for the articulation unit is the maximum angle at which it can bend while encountering a curvilinear profile. Depending on the centerline radius (CLR) of the pipe bend, the articulation unit can facilitate the robot to bend up to a certain angle, which is referred to as the "Tilt". The general representation of Passive compliance, Active compliance and Tilt limits for a piping inspection robot are depicted below in Figure 3.1.

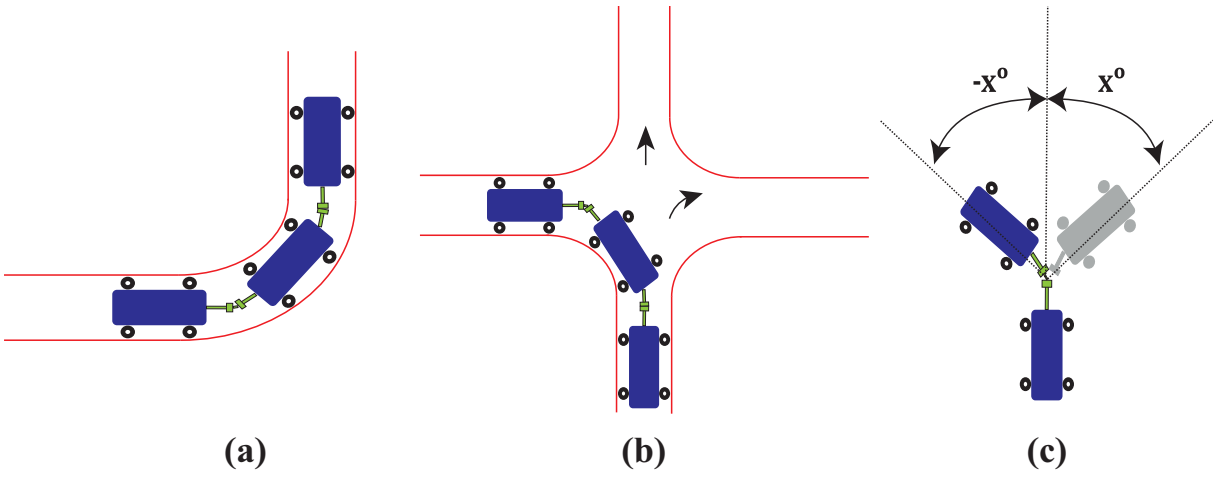


Figure 3.1 – Representation of (a) Passive compliance, (b) Active compliance and (c) Tilt limits for a piping inspection robot [VC19b]

3.2.1 Architecture of the new piping inspection robot

For addressing the compliance factors and the tilt limits, the existing architecture of the bio-inspired robot from Figure 2.1 is modified into a flexible robot and it is represented in Figure 3.2. The caterpillar module represented by $OCAB$ is retained in the modified design to ensure contact with the pipeline walls during static and dynamic phases. $EFGH$ represents a spring-loaded four-ball wheel mechanism. This mechanism will have contact with the pipeline walls at all instances of locomotion. Also, this mechanism assists the robot to smoothly enter

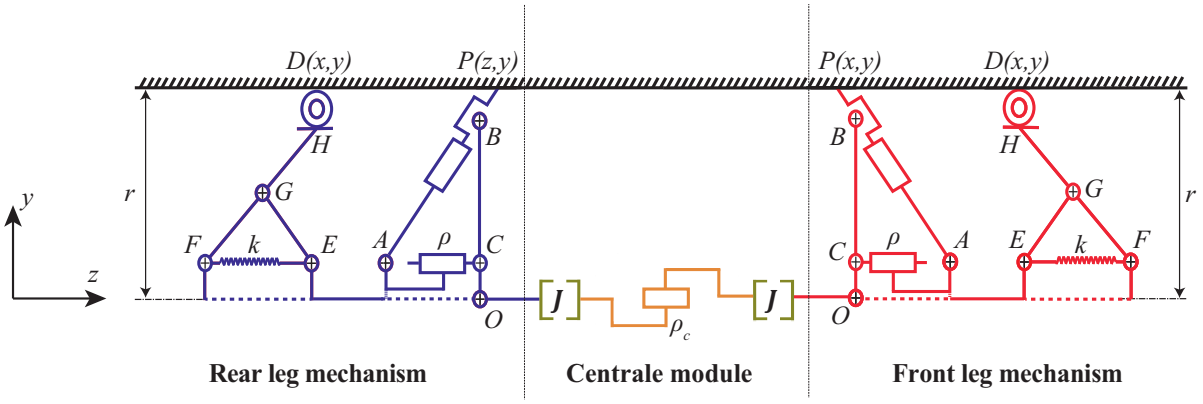


Figure 3.2 – Outline of the modified bio-inspired robot inside a pipeline of radius r

and exit pipe bends and junctions. The parameters ρ and ρ_c represents the clamping and central module actuation units. The joint J represents the new articulation unit that will be introduced in the robot. In the existing prototype, the front and central modules were bolted together but in the proposed design all the modules are split into independent systems. As the robot moves through a pipe bend, the modified architecture resembles an "Elephant Trunk". By ensuring tight contact with the pipeline walls using the caterpillar module, the force is transmitted from the front to the rear end, thereby allowing the robot to pass through or overcome a pipe bend.

3.2.2 Design of a tensegrity mechanism

In order to make the existing robot flexible, an articulation unit in the form of a tensegrity mechanism is proposed and introduced between each module. This mechanism can facilitate the robot to pass through 90° pipe bends or junctions by tilting up to a certain angle. In the existing design, the motor units were coupled rigidly using flanges. By the addition of another flange, a universal joint and three tension springs are introduced between the flanges. The universal joint remains passive and the issue of passive compliance can be addressed by this joint and the springs for a 90° pipe bend. Through each spring, cables are introduced which are connected to one of the flanges of the mechanism. In the event of active compliance, these cables are actuated from a central control unit. This will facilitate the robot to tilt in a certain direction while encountering a junction or a T-union. The cables can be actuated from a DC-Motor through pulleys and some preloading forces are generally applied to have a stable configuration of the mechanism. The architecture of the tensegrity mechanism is represented in Figure 3.3. The tensegrity mechanism resembles the robot analyzed by Yigit et al.[YB17], except with a difference that three tension springs are employed over a single compression spring. The three springs are mounted at an angle of 120° between each other. These springs ensure the stability of the mechanism and they control the tilt limits of the universal joint. The other three holes on the flange can be used for routing the wires coming out from the motors used in the caterpillar module. The springs are mounted at radius r_f from the central axis of the flange. In order to

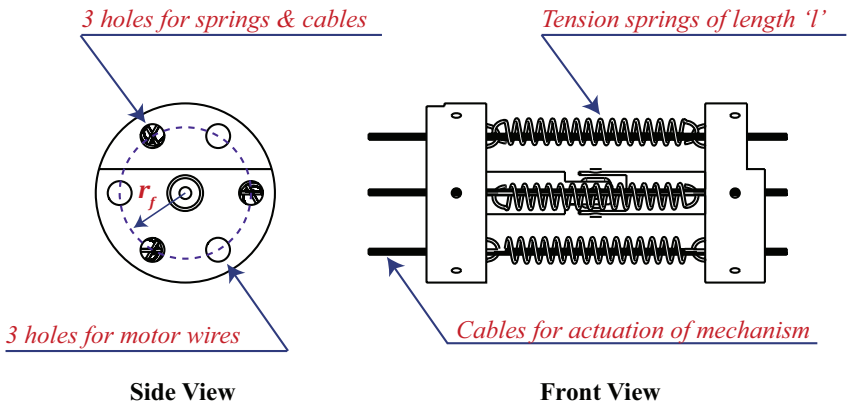


Figure 3.3 – Proposed tensegrity mechanism with universal joint, springs and cables

determine the maximum tilt limits, the radius r_f and the distance l between the flanges have to be considered for analysis, which will be exploited in the upcoming sections.

3.2.3 Geometric equations of the mechanism

For the analysis of the mechanism, it is necessary to derive the equations for the system. The tensegrity mechanism represented in Figure 3.3 is analyzed by correlating to a parallel manipulator of type 3-SPS-U [AS04]. Here S indicates the spherical joint, U indicates the universal joint and P indicates the actuated prismatic joints or the springs. The representation of the tensegrity mechanism at the home-pose and the correlation to a parallel manipulator is represented below in Figure 3.4.

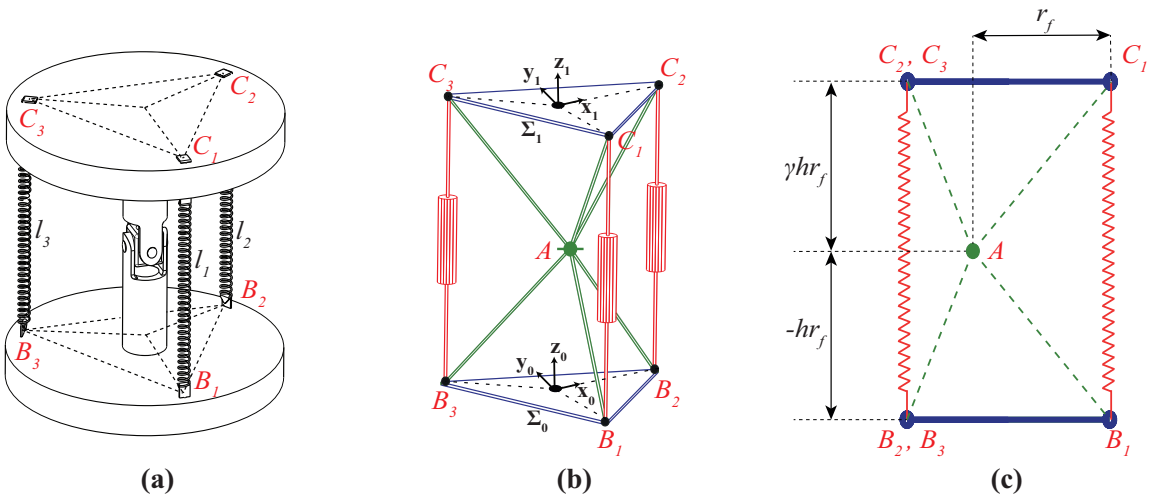


Figure 3.4 – Representation of the (a) Tensegrity mechanism at home-pose, (b) 3D view of the correlation to a 3-SPS-U manipulator and (c) 2D view of the manipulator

The home-pose represented in Figure 3.4a is the position where the tilt angles, as well as the external forces on the mechanism, are zero. The fixed coordinate frame of the base is represented

by $\sum_0$ with its origin at B_0. The spring mounting points on the fixed base are represented by B_1, B_2, B_3 and they form the imaginary equilateral triangle of the manipulator base. The vector coordinates for the base mounting points can be written as:

$$\mathbf{b}_i^0 \;=\; \mathbf{R}_z(\theta)\;\mathbf{n}_i^0, \quad \text{with} \quad i = 1, 2, 3 \tag{3.1}$$

$$\text{where} \;\; \mathbf{R}_z(\theta) \;=\; \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \;\; \& \;\; \mathbf{n}_i^0 = \begin{bmatrix} r_f \cos\left(\frac{2\pi(i-1)}{3}\right) \\ r_f \sin\left(\frac{2\pi(i-1)}{3}\right) \\ -r_f h \end{bmatrix}$$

In Eq. (3.1), $\mathbf{n}_i^0$ are the coordinates of B_i in the fixed coordinate frame $\sum_0$. The rotation matrix $\mathbf{R}_z$ with respect to θ is used to determine the orientation or rotation of the mechanism about the z-axis. For determining the vector coordinates of the end-effector, two sets of analysis are carried out. The first analysis employs the theory of Tilt&Torsion (T&T) for the joint limits of the mechanism. In the second analysis, the singularity analysis of the mechanism is carried out for the Euler angles of the universal joint for identifying the workspaces. These analyses are discussed in detail in the upcoming sections.

3.3 Analysis of joint limits using Tilt & Torsion

In this section, the joint limits of the mechanism are analyzed by parametrizing the orientation of the platform using the theory of Tilt&Torsion (T&T). The T&T theory is employed mainly for parallel manipulators, where the robot platform is mobile and asymmetric [BZG02]. Since the universal joint is passive, there exists no torsion. Thus, the tilt and azimuth angles would be sufficient to calculate the positions of the end-effector. The representation of the tilt and azimuth angles on the tensegrity mechanism are shown in Figure 3.5. In Figure 3.5, β represents the azimuth angle between the x-axis and face of the z-axis. Rotation about M causes a shift of the axis xyz to $x^*y^*z^*$. The angle between z and z^* represents the tilt angle, α. The transformation matrix $\mathbf{R}$ for the tilt and azimuth angles is given by the equation:

$$\mathbf{R} = \mathbf{R}_z(\beta)\mathbf{R}_x(\alpha)\mathbf{R}_z(-\beta) \tag{3.2}$$

$$\mathbf{R} = \begin{bmatrix} \cos(\beta)^2 + \sin(\beta)^2\cos(\alpha) & \cos(\beta)\sin(\beta)(1-\cos(\alpha)) & \sin(\beta)\sin(\alpha) \\ \cos(\beta)\sin(\beta)(1-\cos(\alpha)) & \sin(\beta)^2 + \cos(\beta)^2\cos(\alpha) & -\cos(\beta)\sin(\alpha) \\ -\sin(\beta)\sin(\alpha) & \cos(\beta)\sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{3.3}$$

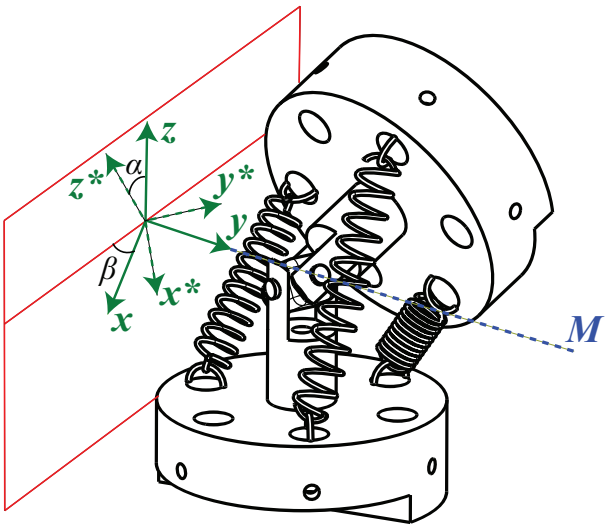


Figure 3.5 – Representation of the Tilt (α) and Azimuth (β) angles on the tensegrity mechanism

$$\text{where } \mathbf{R}_{\mathrm{x}}(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}, \mathbf{R}_{\mathrm{z}}(-\beta) = \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 \\ -\sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The moving coordinate frame of the end-effector is represented by $\sum_1$ with its origin at C_0. The spring mounting points of the end-effector is represented by C_1, C_2, C_3 for the mechanism. In order to obtain the position of the end-effector, Eq. (3.2) is multiplied by the vector coordinates of C_i and the equation is given by:

$$\mathbf{c}_i^1 = \mathbf{R}_z(\theta)\ \mathbf{R}\ \mathbf{o}_i^1, \quad \text{with} \quad i = 1,2,3 \ \ \& \ \ \mathbf{o}_i^1 = \begin{bmatrix} r_f \cos\left(\frac{2\pi(i-1)}{3}\right) \\ r_f \sin\left(\frac{2\pi(i-1)}{3}\right) \\ r_f \gamma \end{bmatrix} \tag{3.4}$$

In Eq. (3.4), $\mathbf{o}_i^1$ are the coordinates of C_i in the mobile reference frame $\sum_1$ of the end-effector. The parameter γ is a scalar which determines the positioning of the end-effector from the origin A. The joint space analysis for the various values of γ will be analyzed in the next section. The length of the mechanism l_i which provides the solution for the Inverse Kinematic Problem (IKP) can be solved by computing the distance between the vector coordinates of the base and the end-effector. The constraint equations are given by:

$$\left\| \mathbf{b}_i^0 - \mathbf{c}_i^1 \right\| = l_i \ , \ \text{where } \mathbf{b}_i^0 = [b_{ix}, b_{iy}, b_{iz}]^T, \mathbf{c}_i^1 = [c_{ix}, c_{iy}, c_{iz}]^T \ , \ \text{with } i = 1,2,3 \tag{3.5a}$$

$$\begin{aligned} &rf^2(\cos(\alpha)(2h^2\gamma + 2\cos(\beta)^2 - 2) - 2h\sin(\alpha)\sin(\beta)(\gamma+1) - 2\cos(\beta)^2 + 2 \\ &+ h^2(\gamma^2+1)) = l_1^2 \end{aligned} \tag{3.5b}$$

$$rf^2(\sqrt{3}\cos(\beta)(-\sin(\beta)\cos(\alpha)+h(\gamma+1)\sin(\alpha)+\sin(\beta))+\cos(\beta)^2(-\cos(\alpha)+1)+$$
$$\cos(\alpha)(2h^2\gamma-\frac{1}{2})+h\sin(\alpha)\sin(\beta)(\gamma+1)+\frac{1}{2}+h^2(\gamma^2+1))=l_2^2 \tag{3.5c}$$

$$rf^2(-\sqrt{3}\cos(\beta)(-\sin(\beta)\cos(\alpha)+h(\gamma+1)\sin(\alpha)+\sin(\beta))+\cos(\beta)^2(-\cos(\alpha)+1)+$$
$$\cos(\alpha)(2h^2\gamma-\frac{1}{2})+h\sin(\alpha)\sin(\beta)(\gamma+1)+\frac{1}{2}+h^2(\gamma^2+1))=l_3^2 \tag{3.5d}$$

3.3.1 Estimation of tilt limits

The tilt range for the tensegrity mechanism must be determined which permits to analyze the geometrical workspace of the mechanism. In Eq. (3.4), the parameter γ in $\mathbf{o}_i$ can assume three values: $\gamma < 0$, $\gamma = 0$ and $\gamma > 0$ [VFCW]. These postures are classified as the Pendulum, the Neutral pose and the Inverse pendulum. The three postures are depicted below in Figure 3.6.

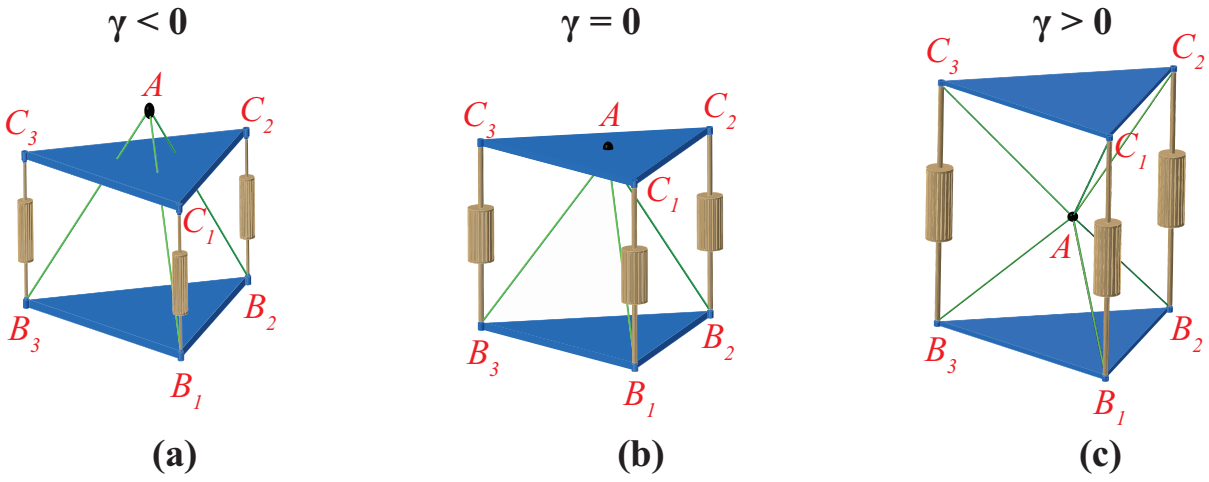


Figure 3.6 – Representation of the (a)Pendulum, (b)Neutral pose and (c) Inverse pendulum postures of the tensegrity mechanism

For the initial analysis, the value of h is set as 1. The tilt limits are estimated by taking into account the joint limits of the mechanism. The equations of the robot and its associated constraints are generated using the SIROPA [JCB+18, MCWR10, CMRW20] library of Maple. The vector coordinates of the base from Eq. (3.1), the end-effector coordinates from Eq. (3.4) and the distance relations Eqs. (3.5b) to (3.5d) that maps the base to the end-effector are used as inputs for the *CreateManipulator* function of SIROPA library. This function virtually constructs the 3-SPS-U manipulator in Maple [JCB+18, CMRW20]. The tilt and azimuth angles are set as the pose variables and the lengths (l_1, l_2 and l_3) are set as the articular variables for computations. The constraint equations are once again used to determine the workspace of the three case-study postures by setting joint limits for the springs. Using the *ConstraintEquations* function of SIROPA library and with $h = 1$, Eqs. (3.5b) to (3.5d) are transformed into six

constraint equations which are given by :

$$\begin{aligned}\mathcal{C}_{1+3i} : & 2r_f^2\cos(\alpha)(\cos(\beta)^2 - 1 + \gamma) - 2r_f^2\sin(\beta)\sin(\alpha)(1+\gamma) - 2r_f^2\cos(\beta)^2 \\ & + r_f^2(\gamma^2+3) = l_j^2 \end{aligned} \tag{3.6}$$

$$\begin{aligned}\mathcal{C}_{2+3i} : & 2\sqrt{3}r_f^2\cos(\beta)(\sin(\alpha)\gamma + \sin(\beta) - \cos(\alpha)\sin(\beta) + \sin(\alpha)) + r_f^2(3-\cos(\alpha)) \\ & + 2r_f^2(\sin(\alpha)\sin(\beta)(1+\gamma) - \cos(\beta)^2(\cos(\alpha)-1) + \gamma(2\cos(\alpha)+\gamma)) = 2l_j^2 \end{aligned} \tag{3.7}$$

$$\begin{aligned}\mathcal{C}_{3+3i} : & -2\sqrt{3}r_f^2\cos(\beta)(\sin(\alpha)\gamma + \sin(\beta) - \cos(\alpha)\sin(\beta) + \sin(\alpha)) + r_f^2(3-\cos(\alpha)) \\ & + 2r_f^2(\sin(\alpha)\sin(\beta)(1+\gamma) - \cos(\beta)^2(\cos(\alpha)-1) + \gamma(2\cos(\alpha)+\gamma)) = 2l_j^2 \end{aligned} \tag{3.8}$$

$$\text{where for } i = \ 0 \text{ and } 1, \ l_j = \ l_{min} \text{ and } l_{max}$$

For γ, three values are assumed: $-1/4$, 0, 1 and they are employed in Eqs. (3.6) to (3.8). The value for r_f is retained as 11 mm from the dimensions of the flange used in the existing prototype [VCB19]. For estimating the geometrical workspaces of the mechanism, the Cylindrical Algebraic Decomposition (CAD) algorithm is employed. The workspace (resp. joint space) analysis classifies the number of solutions of the parametric system associated with the direct (resp. inverse) kinematic problem [Jha16]. This method was proposed and analyzed for parallel robots by Chablat et al.[CMW11]. The main steps are recalled here in which, the workspace, as well as joint space, are decomposed into cells C_1,...C_k, such that:

- C_i is an open connected subset of the workspace
- for all pose values in C_i, the direct (resp. inverse) kinematics problem has a constant number of solutions
- C_i is maximal in the sense if C_i is contained in a set E, then E does not satisfy the first or second condition.

The three main steps that are carried out in this analysis are [CMW11]:

- Computation of a subset of the workspace (resp. joint space) where the number of solutions changes: the *Discriminant Variety*
- Description of the complementary of the discriminant variety in connected cells: the *Generic Cylindrical Algebraic Decomposition*
- Connecting the cells that belong to the same connected component of the complementary of the discriminant variety: *interval comparisons*

From a general point of view, the discriminant variety is defined for any system of polynomial equations and inequalities. Let p_1,...,p_m, q_1,...,q_l be polynomials with rational coefficients depending on the unknowns X_1,...,X_n and on the parameters U_1,...,U_d. The following constructible set is considered:

$$\mathcal{E} = \{\mathbf{v} \in \mathbb{C}^{n+d}, p_1(\mathbf{v}) = 0, ..., p_m(\mathbf{v}) = 0, q_1(\mathbf{v}) \neq 0, ..., q_l(\mathbf{v}) \neq 0\} \tag{3.9}$$

If it is assumed that $\mathcal{E}$ is a finite number of points for almost all the parameter values, a discriminant variety V_D of $\mathcal{E}$ is a variety in the parameter space $\mathbb{C}^d$ such that over each connected open set $\mathcal{U}$ satisfying $\mathcal{U} \cap V_D = \Phi$, $\mathcal{E}$ defines an analytic covering. In particular, the number of points of $\mathcal{E}$ over any point of $\mathcal{U}$ is constant. Now, the following semi-algebraic set is considered:

$$\mathcal{F} = \{\mathbf{v} \in \mathbb{C}^{n+d}, p_1(\mathbf{v}) = 0, ..., p_m(\mathbf{v}) = 0, q_1(\mathbf{v}) \geq 0, ..., q_l(\mathbf{v}) \geq 0\} \tag{3.10}$$

If $\mathcal{F}$ is assumed to have a finite number of solutions over at least one real point that does not belong to V_D, then $V_D \cap \mathbb{R}^d$ can be viewed as a real discriminant variety of $\mathcal{F}$ with the same property: over each open set $\mathcal{U} \subset \mathbb{R}^d$ such that $\mathcal{U} \cap V_D = \Phi$, $\mathcal{E}$ defines an analytic covering. In particular, the number of points of $\mathbb{R}$ over any point of $\mathcal{U}$ is constant [CMW11, JCB+18]. Discriminant varieties can be computed using basic and well-known tools from computer algebra such as the Groebner bases [CLO13]. A general framework for computing such objects is available through the *RootFinding*[*Parametric*] function of Maple. For estimating the geometric workspace for the three case study postures, the CAD algorithm combined with the parametric root finding technique of Maple is employed to estimate solutions for the tilt and azimuth angles [COM11]. This is carried out by using the *CellDecompositionPlus* function of the SIROPA library in Maple [JCB+18, CMRW20]. For isolating the aspects around the home-pose, Eqs. (3.6) to (3.8) are transformed as inequality equations [CW98], which are then used as inputs for *CellDecompositionPlus* in Maple. For the constraint equations, it is also necessary to set the joint limits of l_i, which determines the maximum tilt limits for the mechanism. At the home-pose as represented in Figure 3.4a, the length between $B - C$ are calculated and they are given by: 9 mm, 11 mm and 22 mm for γ= -1/4, 0 and 1. The constraint limits for these lengths under maximum tilt angles are set as $l_{min} = 7$ mm and $l_{max} = 31$ mm for the analysis. The results of the geometric workspace are then generated for the three cases of γ using CAD algorithm in Maple. The solutions for the three postures are represented in Figure 3.7. From the results of CAD algorithm, the number of cells that corresponds to the number of solutions for the plots represented in Figure 3.7 are provided below in Table 3.1.

Architecture	**0 solutions**	**2 solutions**	**Total cells**
Pendulum	134	88	222
Neutral pose	134	88	222
Inverse pendulum	586	136	722

Table 3.1 – Number of cells obtained by CAD algorithm for Figures 3.7(i) to 3.7(iii)

The white zones in Figures 3.7(i) to 3.7(vi) for the three configurations indicate the regions where there exists no solutions. The blue regions indicate feasible postures for the manipulator,

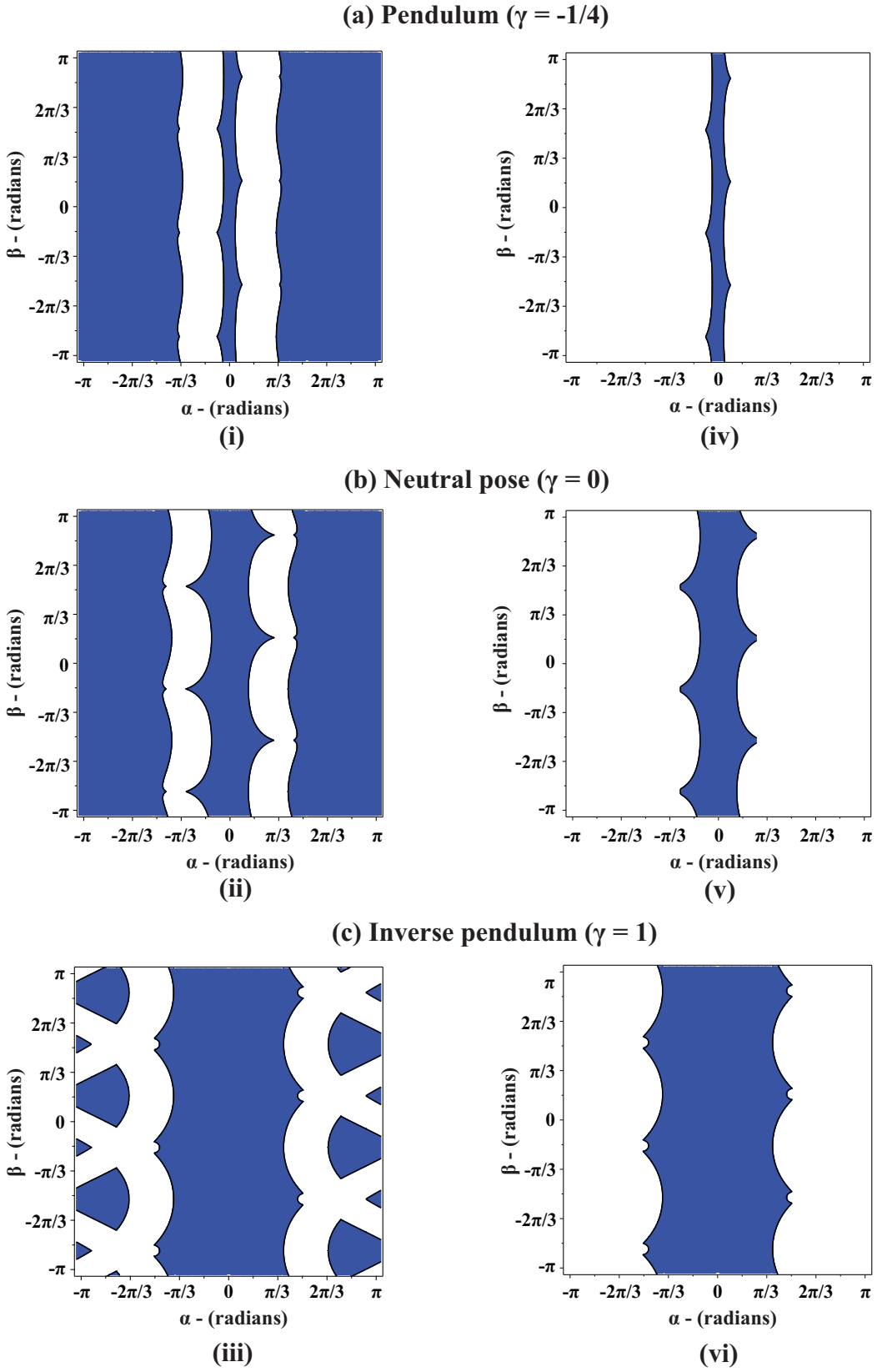


Figure 3.7 – Results obtained by CAD algorithm from Maple with representation of (i),(ii),(iii): Feasible and non-feasbile solutions and (iv),(v),(vi): Extraction of geometrical workspaces around the home-pose for the three case study postures

where there exists two solutions for the Direct Kinematic Problem (DKP). Around the home-pose, the mechanism can tilt up to the values at the boundaries of the blue region. There also exists some blue regions beyond the white zones as seen in Figures 3.7(i) to 3.7(iii). However, in these regions, the orientations of the manipulators are in such a way that there exists collisions between the base and the end-effector. In other words, the base and the end-effector align themselves on the same plane. For the inverse pendulum, the azimuth range beyond the white zones is intermittently distributed for a given tilt angle. For the pendulum and neutral postures, the azimuth ranges have no discontinuities beyond the white zones for a given tilt angle. Using the *PlotRobot3D* function of SIROPA, the postures of the mechanisms in blue regions around the home-pose and beyond the white zones are created and they are represented in Figures 3.8(i) to 3.8(vi). The postures of the mechanisms for the blue regions beyond the white zones are represented in Figures 3.8(iv) to 3.8(vi) where collisions could be observed. Therefore these regions are eliminated and the feasible solutions around the home-pose are extracted. The feasible tilt limits for the three case study configurations around the home-pose are represented in Figures 3.7 (iv) to 3.7 (vi). The corresponding values of the tilt angles with the azimuth angles at the joint limits around the home-pose are provided below in Table 3.2.

Architecture	γ	**Azimuth** (β)	**Tilt** (α)
Pendulum	-1/4		$\pm\pi/18$
Neutral pose	0	$[\pm\pi/2,\pm5\pi/6]$	$\pm\pi/6$
Inverse pendulum	1		$\pm\pi/3$

Table 3.2 – Tilt limits for azimuth values at joint limits around the home-pose for Figures 3.7(iv) to 3.7(vi)

For the case of pendulum, narrow tilt limit ranges could be observed from Table 3.2 and from Figure 3.7(iv). This configuration might not be able to overcome a pipe bend of 90° when it is coupled with the bio-inspired robot. Better solutions are obtained for the neutral posture when compared to the pendulum configuration and this could be observed in Figure 3.7(v). However, in the case of the inverse pendulum, the tilt limit ranges are comparatively higher than the other two configurations. This configuration can thus be employed for the piping inspection robot to overcome bends and junctions. As the inverse pendulum configuration by nature is not stable, static stabilities of the mechanism must be studied in order to ensure that the mechanism does not topple down at the home-pose under the absence of external forces.

3.3.2 Static analysis of the mechanism

With the inverse pendulum configuration being identified as a suitable configuration for the tensegrity mechanism, it is important to analyze its stability. In the event of active compliance, when the tensegrity mechanism is coupled with the robot, the stability of the mechanism can

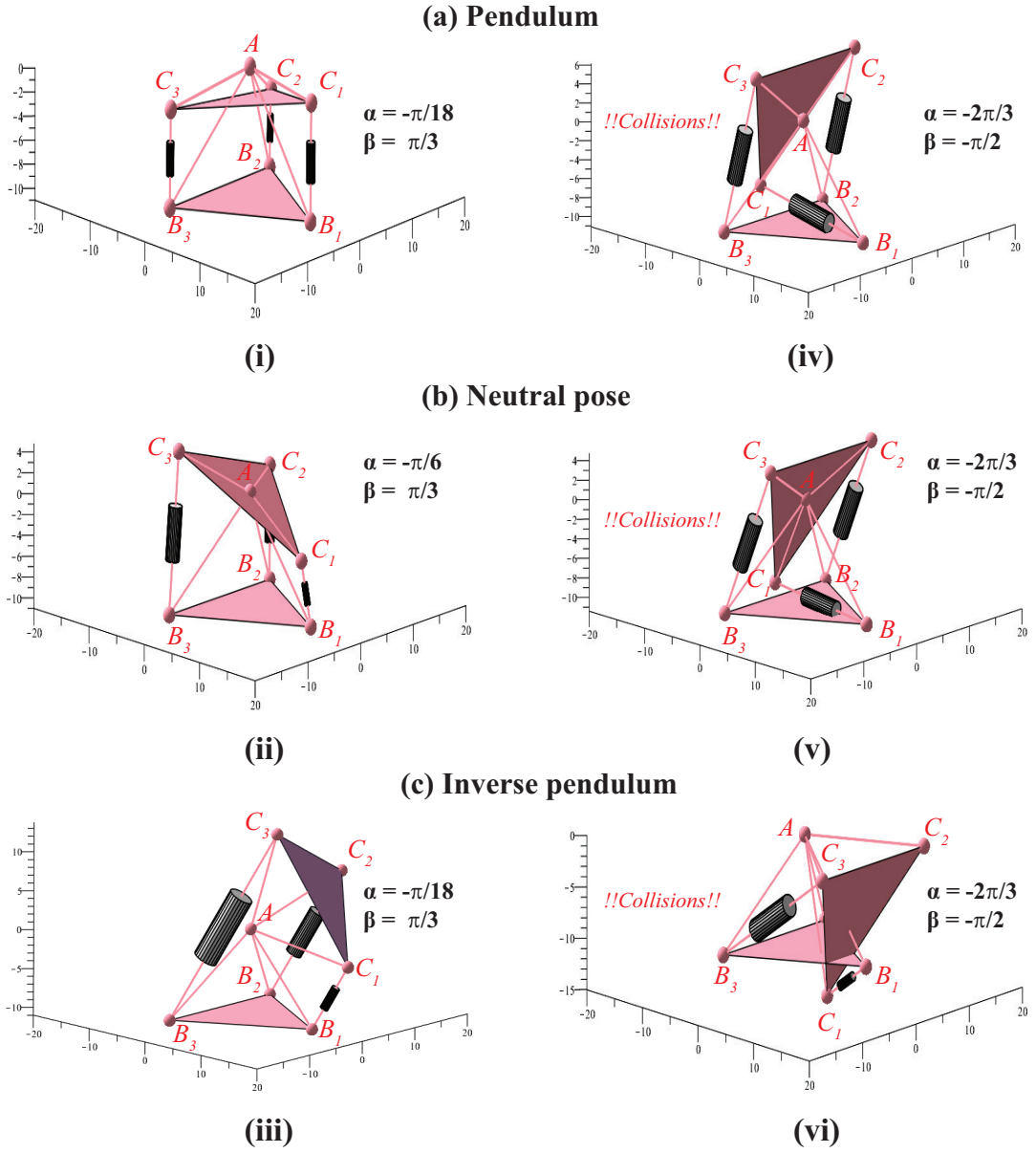


Figure 3.8 – Representation of the manipulator orientations in SIROPA in: (i),(ii),(iii): Blue regions around the home-pose ,(iv),(v),(vi): Blue regions beyond the white zones for the three case study postures

be controlled through cables. This will also facilitate the robot to follow a given direction. For passive compliance, where the cable actuation is not required, the tensegrity mechanism must overcome pipe bends autonomously. In this scenario, the static stability of the mechanism has to be determined. According to Lagrange, for a moving system, the equation of motion [Mei10] is given by:

$$\boldsymbol{\tau} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial T}{\partial \mathbf{q}} + \frac{\partial U}{\partial \mathbf{q}} \quad \text{where} \quad \mathbf{q} = [\alpha, \beta]^T \tag{3.11}$$

In Eq. (3.11), T and U are the kinetic and potential energies of the system. $\boldsymbol{\tau}$ represents the generalized torques on the system. Under static modes, the velocity of the system is zero and there exists no kinetic energies. Thus, in Eq. (3.11) only the potential energy derivative with respect to the tilt and azimuth angles exists. The potential energy of the system is contributed mainly by the springs and cables. For this calculation, the magnitude of applied forces along the three cables are considered as 'F_1', 'F_2', 'F_3' N. Three identical springs of stiffness 'k' N/mm is chosen for the mechanism. The total potential energy, U_{tot} can be estimated by summing up the potential energies of springs and cables. The equations are given by:

$$U_{cable} = \sum_{i=1}^{3} F_i l_i \quad U_{spring} = \sum_{i=1}^{3} \tfrac{1}{2} k (l_i - l_0)^2 \tag{3.12}$$

$$U_{tot} = U_{cable} + U_{spring} \tag{3.13}$$

For the stability analysis, the springs are considered to be massless and their free lengths l_0 are considered to be 0 mm. The potential energy also consists of the gravity term. This term is however not considered as the mass parameter is only known when a real prototype is constructed. By substituting the results of Eq. (3.13) in Eq. (3.11), the forces required to actuate the mechanism can be calculated. Under zero-applied forces or under the presence of a preload, the mechanism can deform due to the springs. Arsenault et al. [AG06] analyzed that for a tensegrity mechanism to be stable under static modes, the second-order derivative or the stiffness of the mechanism 'K' has to be positive. The U_{tot} equation is first derived for the inverse pendulum configuration with $\gamma = 1$ in Maple and it can be expressed as:

$$\begin{aligned} U_{tot} = {} & \sqrt{2} F_1 (r_f^2 ((h^2 + \cos(\beta)^2 - 1)\cos(\alpha) - 2\sin(\beta)\sin(\alpha)h + h^2 - \cos(\beta)^2 + 1))^{1/2} + \\ & \sqrt{2} F_2 (r_f^2 (\sqrt{3}\cos(\beta)(\sin(\alpha)h - \frac{\cos(\alpha)\sin(\beta)}{2} + \frac{\sin(\beta)}{2}) + \cos(\alpha)(h^2 - \frac{\cos(\beta)^2}{2} - \frac{1}{4}) \\ & + h\sin(\alpha)\sin(\beta) + h^2 + \frac{\cos(\beta)^2}{2} + \frac{1}{4}))^{1/2} + \sqrt{2} F_3 (r_f^2 (-\sqrt{3}\cos(\beta)(\sin(\alpha)h + \frac{\sin(\beta)}{2} \\ & - \frac{\cos(\alpha)\sin(\beta)}{2}) + \cos(\alpha)(h^2 - \frac{\cos(\beta)^2}{2} - \frac{1}{4}) + h\sin(\alpha)\sin(\beta) + h^2 + \frac{\cos(\beta)^2}{2} + \frac{1}{4}))^{1/2} \\ & + \zeta \end{aligned} \tag{3.14}$$

$$\text{where } \zeta = 3\, k {r_f}^2 \left(\left(h^2 - \frac{1}{2} \right) \cos(\alpha) + h^2 + \frac{1}{2} \right)$$

At the home-pose condition represented in Figure 3.4, when the applied forces on the mechanism are zero, only the constant term ζ of Eq. (3.14) exists. This term depends on the spring stiffness k, mounting radius r_f, parameter h and the tilt angle α. The stability of the mechanism can be understood by plotting the total potential energy over the tilt range that was estimated for the inverse pendulum configuration. A spring of 0.75 N/mm is taken for this analysis. The

value of r_f is carried over as 11 mm from existing prototype and the value of h is taken as 1. Using these values, the total potential energy of the mechanism is traced against the tilt angle α around the home-pose and it is represented below in Figure 3.9.

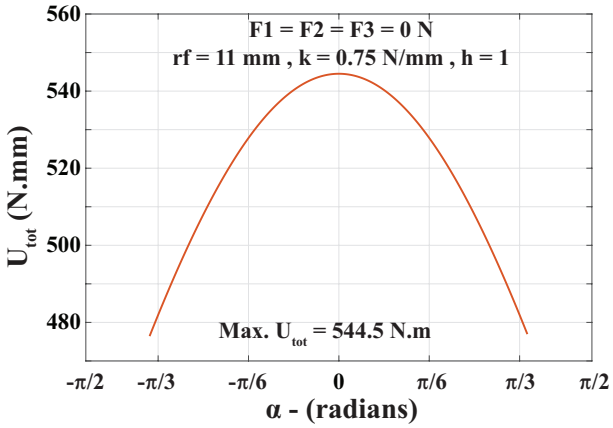


Figure 3.9 – Plot of total potential energy versus the tilt(α) at the home-pose

It could be observed that the mechanism remains unstable and has maximum potential energy in the home-pose when the applied forces are zero. Since U_{tot} has a direct dependency on the applied forces in Eq. (3.14), the system continues to remain unstable when there is an external loading. At the home-pose condition, the total potential energy of the system has no dependencies on the azimuth angle β. In order to determine stable configurations, the second-order derivative of the total potential energy with respect to the tilt angle α, which provides the stiffness of the mechanism K_α must be studied against the design parameters r_f and h at the home-pose. The mechanism stiffness K_α is derived using Maple at the home-pose condition and the equation is given by:

$$K_\alpha = 3kr_f^2 \cos(\alpha)(\frac{1}{2} - h^2) \tag{3.15}$$

From Eq. (3.15), it could be observed that the mechanism stiffness depends on the parameters r_f, h and spring stiffness k. By fixing one or several parameters, it is possible to determine a stable configuration. The stiffness of the mechanism K_α is analyzed for varying values of h and r_f for a spring stiffness $k = 0.75$ N/mm at $\alpha = 0$ radians and this plot is represented in Figure 3.10. It could be observed from Figure 3.10 that as h increases towards 1, the mechanism stiffness transforms from a positive to a negative zone. On the contrary, with an increase in the value of r_f, the mechanism stiffness increases equally along the positive and the negative directions. The choice of parameter h influences the positive zone of K_α. With respect to the dimensions of the flanges used in the prototype of the bio-inspired robot studied in Chapter-2, the value of r_f is retained as 11 mm. Under the absence of external loading, the optimum value of h at which the mechanism can remain stable is identified by solving Eq. (3.15) at the home-pose. If $h > 0.707$, the mechanism is unstable and if $h < 0.707$, the mechanism becomes stable.

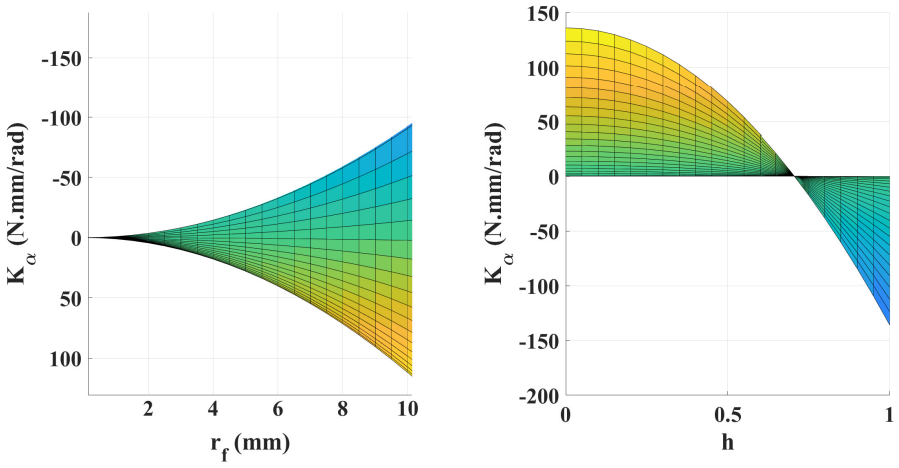


Figure 3.10 – Projection of K_α with respect to r_f (left) and h (right) at the home-pose

The total potential energy of the mechanism for values of h = [1, 0.707, 0.414] is represented below in Figure 3.11.

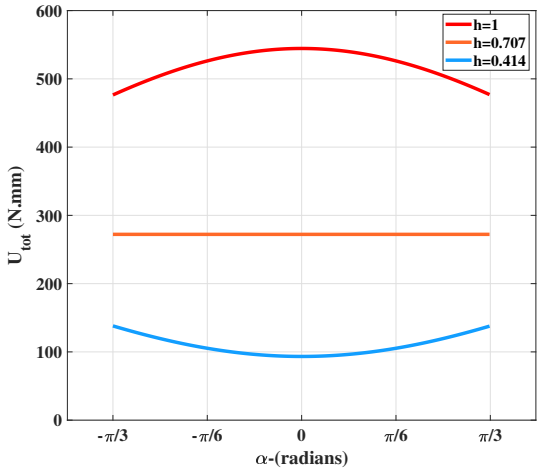


Figure 3.11 – Plot of total potential energy of the mechanism with h = [1, 0.707, 0.414] under the absence of external forces

From Figure 3.11, the flat line at h=0.707 is the transformation line where the potential energy can go from stable to unstable state or vice-versa. It has to be noted that the analysis is carried out by considering the free length of the spring as 0 mm. However, if the free length of the spring is taken into account, the position of flat line shifts from 0.707 to a higher value. The amount of shift depends mainly on the stiffness of the spring chosen. As the tensegrity mechanism is proposed to be integrated along with the bio-inspired robot which will impose an additional load on the mechanism, a value of h= 0.5 from the stable zone is considered. The geometric workspace of the mechanism is recalculated for the new inverse pendulum configu-

ration by using Eqs. (3.6) to (3.8) with the help of CAD algorithm [CMW11] in Maple. The new geometric workspace for this configuration with 320 cells having 0 solutions (White) and 106 cells (Blue) with 2 solutions to the DKP are represented in Figure 3.12a. As there exists collisions between base and end-effector of the mechanism for the blue regions beyond white zones, these regions are eliminated. The geometrical workspace with feasible tilt limits around the home-pose are extracted and the cells are represented in Figure 3.12b.

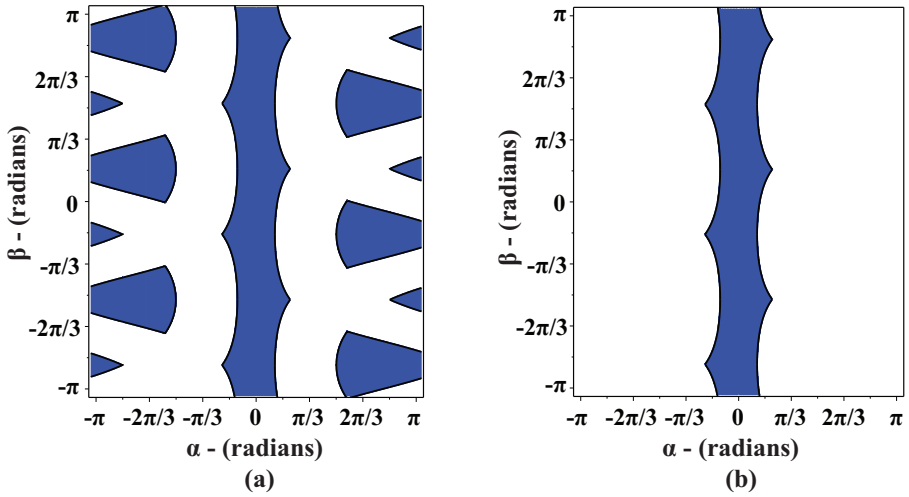


Figure 3.12 – Results obtained by CAD algorithm for the modified inverse pendulum configuration of h=0.5 with (a) Feasible and non-feasible solutions and (b)Extraction of feasible solutions around the home-pose

From the results of Figure 3.12b and CAD algorithm, it is found that the tilt limits for the mechanism have been reduced to $\pm\ \pi/6$ radians. This mechanism can be integrated into the bio-insired robot studied in Chapter-2. Due to the presence of motor modules, there exists some preload on the tensegrity mechanism. With the new feasible tilt range obtained in Figure 3.12, the total potential energy of the mechanism is recalculated at the home-pose without any preloading and with an equal force distribution of $F_1 = F_2 = F_3 = 2$ N. The plots of total potential energy under these conditions are represented in Figures 3.13a and 3.13b. The preloading of 2 N along each spring is considered based on the existing weight of the prototype [VCB19], which is around 6.5 N. Even though the tilt limits have been reduced by half when compared to the initial solutions, a stable configuration has been identified. Under real-time applications, this modified configuration can be potentially stable under the influence of weight of motors during static and dynamic phases of the bio-inspired robot.

3.4 Analysis of workspaces with respect to the Euler angles

The T&T theory gave an insight into the estimation of tilt angles for the joint limits of the springs through the study of three different postures. For studying the singularities of the

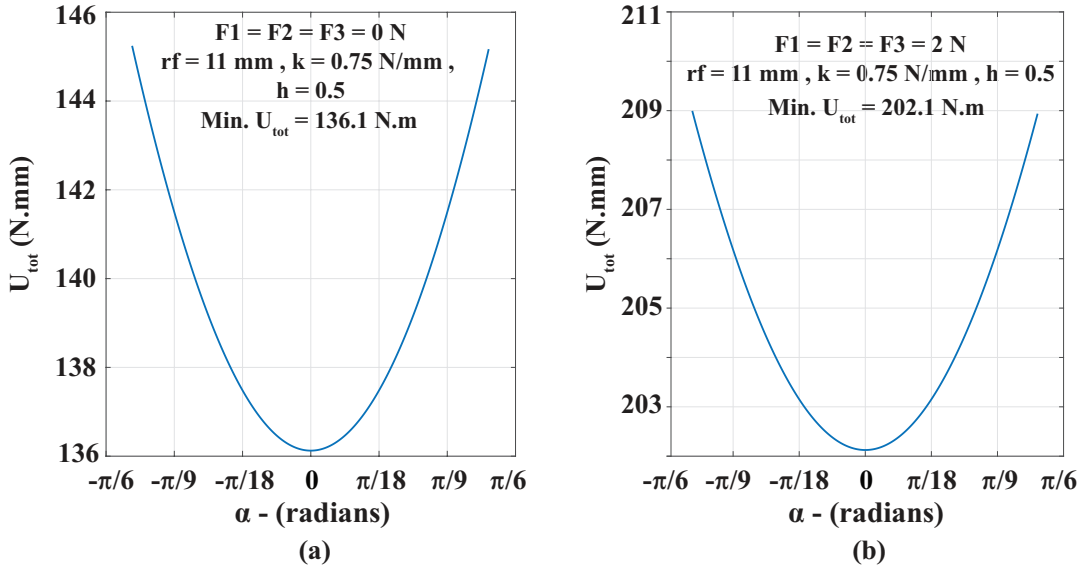


Figure 3.13 – Stability plot of the tensegrity mechanism with h=0.5 under (a) no-load condition and (b) preloading of F_1=F_2=F_3 = 2 N

mechanism, the Tilt & Azimuth angles will not be compatible for mapping the end-effector from the base with respect to the universal joint. At the home-pose represented in Figure 3.4, the Euler angles of the universal joint [SSD12] will be employed to identify the constraint equations of the mobile platform. For identifying feasible workspaces of the mechanism, especially under passive modes, singularities of the mechanism must be studied. The singularity curves and workspaces are affected by the tilt angles of the universal joint. By setting the joint limits for the mechanism, the singularity equations are generated and analyzed by using the SIROPA library of Maple [JCB+18, CMRW20]. For determining the vector coordinates of C_i, the XY Euler angles of the universal joint about point A from Figure 3.4a is employed. The transformation matrix $\mathbf{E}$ about the universal joint is given by:

$$\mathbf{E} = \mathbf{R}_x(\eta)\mathbf{R}_y(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ \sin(\eta)\sin(\phi) & \cos(\eta) & -\sin(\eta)\cos(\phi) \\ -\cos(\eta)\sin(\phi) & \sin(\eta) & \cos(\eta)\cos(\phi) \end{bmatrix} \tag{3.16}$$

$$\text{where } \mathbf{R}_\mathrm{x}(\eta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\eta) & -\sin(\eta) \\ 0 & \sin(\eta) & \cos(\eta) \end{bmatrix}, \mathbf{R}_\mathrm{y}(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 1 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{bmatrix}$$

In Eq. (3.16), η and ψ are the XY Euler angles or tilt angles for the 3-SPS-U tensegrity mechanism. At the home-pose condition represented in Figure 3.4, these tilt angles are zero. In Eq. (3.16), $\mathbf{E} \in SE(3)$ represents the spatial transformation matrix obtained from the Euler

angles of universal joint. This matrix is used to identify the end-effector coordinates and the equation is given by:

$$\mathbf{c}_i^1 = \mathbf{E}\ \mathbf{o}_i^1, \quad \text{with} \quad i = 1, 2, 3 \tag{3.17}$$

In Eq. (3.17), $\mathbf{o}_i^1$ are the coordinates of C_i in the mobile reference frame $\sum_1$ whose vector equation is given in Eq. (3.4). The IKP which determines the length of the prismatic springs is computed by calculating the distance between base and end-effector at the home-pose and working conditions. The constraint equations are given by:

$$2r_f^2(\cos(\phi)(h^2\cos(\eta) - 1) - h\sin(\phi)(\cos(\eta) + 1) + h^2 + 1) = l_1^2 \tag{3.18a}$$

$$2r_f^2(\frac{\sqrt{3}}{2}\sin(\eta)(h\cos(\phi) + h + \frac{\sin(\phi)}{2}) + \cos(\phi)(h^2\cos(\eta) - \frac{1}{4}) + \frac{h\sin(\phi)}{2}(1 + \cos(\eta)) + h^2 - \frac{3\cos(\eta)}{4} + 1) = l_2^2 \tag{3.18b}$$

$$-2r_f^2(\frac{\sqrt{3}}{2}\sin(\eta)(h\cos(\phi) + h + \frac{\sin(\phi)}{2}) - \cos(\phi)(h^2\cos(\eta) - \frac{1}{4}) - \frac{h\sin(\phi)}{2}(1 + \cos(\eta)) - h^2 + \frac{3\cos(\eta)}{4} - 1) = l_3^2 \tag{3.18c}$$

3.4.1 Singularities and joint limits equations

For identifying feasible workspaces for the mechanism, it is important to study the singularities. For a parallel mechanism, the singularity equation is given by the well-known equation [GA90]:

$$\mathbf{A}\mathbf{t} + \mathbf{B}\dot{\rho} = 0 \tag{3.19}$$

where $\mathbf{t}$ represents the angular velocity vector

and $\dot{\rho} = [\dot{l_1}, \dot{l_2}, \dot{l_3}]^T$ represents the joint velocity vector

In Eq. (3.19), $\mathbf{A}$ represents the direct-kinematics matrix or forward Jacobian matrix and $\mathbf{B}$ represents the inverse-kinematics matrix or inverse Jacobian matrix of the mechanism. The pose variables for the mechanism are the tilt angles η and ϕ. The articular variables are the lengths: l_1, l_2 and l_3. For the tensegrity mechanism, the three types of parallel singularities [GA90, WC97, CW98] which was studied in Chapter-1 are verified using Eq. (3.19). From the number of pose variables and articular variables, it could be seen that the matrix $\mathbf{A}$ does not correspond to a $n \times n$ square matrix. For identifying the singularity equations and to construct a square matrix, the 3-SPS-U mechanism is split into three sets of 2-SPS-U architecture which comprises of length pairs $l_1 - l_2$, $l_2 - l_3$ and $l_1 - l_3$ [VC20]. The corresponding direct kinematic matrices $\mathbf{A}$ are created for the three sets by differentiating Eqs. (3.18a) to (3.18c) with respect

to the pose variables. The matrices are given by:

$$\mathbf{A}_{1-2} = \begin{bmatrix} \frac{\partial L_1}{\partial \eta} & \frac{\partial L_1}{\partial \phi} \\ \frac{\partial L_2}{\partial \eta} & \frac{\partial L_2}{\partial \phi} \end{bmatrix}, \ \mathbf{A}_{2-3} = \begin{bmatrix} \frac{\partial L_2}{\partial \eta} & \frac{\partial L_2}{\partial \phi} \\ \frac{\partial L_3}{\partial \eta} & \frac{\partial L_3}{\partial \phi} \end{bmatrix}, \ \mathbf{A}_{3-1} = \begin{bmatrix} \frac{\partial L_3}{\partial \eta} & \frac{\partial L_3}{\partial \phi} \\ \frac{\partial L_1}{\partial \eta} & \frac{\partial L_1}{\partial \phi} \end{bmatrix} \tag{3.20}$$

In Eq. (3.20), L_1, L_2 and L_3 represents the LHS of Eqs. (3.18a) to (3.18c). The parallel singularities are analyzed by calculating the determinant of matrices in Eq. (3.20) and equating them to zero. The determinant values are given by:

$$\begin{aligned} \det(\mathbf{A}_{1-2}), \mathcal{D}_{p1} : 2\sqrt{3}h(h\cos(\eta)^2(\cos(\phi) + h\sin(\phi)) + \cos(\eta)(h\cos(\phi)) - \sin(\phi) - 1 + h\cos(\phi) \\ (h\sin(\phi)\cos(\phi)\cos(\eta))) + \sqrt{3}(\cos(\phi)^2(h^2 + \cos(\eta)) + h\cos(\phi)\sin(\phi)(1 - \cos(\eta)) \\ + \cos(\eta)(3h^2\cos(\eta) - 1)) + 3(\sin(\eta)(h - \sin(\phi) + h\sin(\phi)\cos(\eta)) - 2h^3(\cos(\phi)^2 \\ + \cos(\eta)) + h\sin(\eta)\cos(\phi)(h\sin(\phi) - \cos(\phi) + \cos(\eta) + 1)) = 0 \end{aligned} \tag{3.21}$$

$$\begin{aligned} \det(\mathbf{A}_{2-3}), \mathcal{D}_{p2} : 2h(2h\cos(\eta)^2(\cos(\phi) - h\sin(\phi)) + \sin(\phi)\cos(\phi)(1 + 2\cos(\eta) - 4h^2) - 2h \\ + 2h\cos(\phi)^2(\cos(\eta) + 2) + \cos(\eta)(\sin(\phi) + 2h\cos(\phi))) + \cos(\eta)(1 - \cos(\phi)^2) + \\ 3(\cos(\phi)(\cos(\eta)^2 - 1) + 2h\sin(\phi)(1 - \cos(\eta)^2) + \cos(\eta)(1 - \cos(\phi)^2)) = 0 \end{aligned} \tag{3.22}$$

$$\begin{aligned} \det(\mathbf{A}_{3-1}), \mathcal{D}_{p3} : -2\sqrt{3}h(h\cos(\eta)^2(\cos(\phi) + h\sin(\phi)) + \cos(\eta)(h\cos(\phi)) - \sin(\phi) - 1 + h\cos(\phi) \\ (h\sin(\phi)\cos(\phi)\cos(\eta))) - \sqrt{3}(\cos(\phi)^2(h^2 + \cos(\eta)) + h\cos(\phi)\sin(\phi)(1 - \cos(\eta)) \\ + \cos(\eta)(3h^2\cos(\eta) - 1)) + 3(\sin(\eta)(h - \sin(\phi) + h\sin(\phi)\cos(\eta)) - 2h^3(\cos(\phi)^2 \\ + \cos(\eta)) + h\sin(\eta)\cos(\phi)(h\sin(\phi) - \cos(\phi) + \cos(\eta) + 1)) = 0 \end{aligned} \tag{3.23}$$

Using Eqs. (3.21) to (3.23), it is possible to identify the pose variables and understand the presence of parallel singularities in the mechanism. In a similar way, the inverse kinematics matrix $\mathbf{B}$ is created by differentiating RHS of Eqs. (3.18a) to (3.18c) with respect to the articular variables. Followed by that, the determinant of these matrices are computed and equated to zero for verifying serial singularities. The matrices and the determinant values are given by:

$$\mathbf{B}_{1-2} = \begin{bmatrix} \frac{\partial R_1}{\partial l_1} & \frac{\partial R_1}{\partial l_2} \\ \frac{\partial R_2}{\partial l_1} & \frac{\partial R_2}{\partial l_2} \end{bmatrix}, \ \mathbf{B}_{2-3} = \begin{bmatrix} \frac{\partial R_2}{\partial l_2} & \frac{\partial R_2}{\partial l_3} \\ \frac{\partial R_3}{\partial l_2} & \frac{\partial R_3}{\partial l_3} \end{bmatrix}, \ \mathbf{B}_{3-1} = \begin{bmatrix} \frac{\partial R_3}{\partial l_3} & \frac{\partial R_3}{\partial l_1} \\ \frac{\partial R_1}{\partial l_3} & \frac{\partial R_1}{\partial l_1} \end{bmatrix} \tag{3.24}$$

$$\det(\mathbf{B}_{1-2}), \mathcal{D}_{s1} : l_1 l_2 = 0, \quad \det(\mathbf{B}_{2-3}), \mathcal{D}_{s2} : l_2 l_3 = 0, \quad \det(\mathbf{B}_{3-1}), \mathcal{D}_{s3} : l_3 l_1 = 0 \tag{3.25}$$

In Eq. (3.24), R_1, R_2 and R_3 represents the RHS of Eqs. (3.18a) to (3.18c). From Eq. (3.25), when the determinant of the inverse kinematics matrices are equated to zero, serial singularities

can occur only when the length of one or two prismatic springs becomes zero. This condition is not feasible for the mechanism under study. Thus, there exists no serial singularities in the mechanism. Using Eqs. (3.21) to (3.23), the parallel singularities present in the mechanism are analyzed to extract the feasible workspaces. With the help of SIROPA library, Eqs. (3.21) to (3.23) are generated using *ParallelSingularities* function [JCB+18]. Similarly, the joint limits are set for the mechanism using the *ConstraintEquations* function of SIROPA. By setting h = 1 for the initial analysis and retaining r_f as 11 mm from the existing prototype [VFCW], Eqs. (3.18a) to (3.18c) are transformed into six constraint equations which are given by:

$$\mathcal{G}_{1+3i} : 242\cos(\phi)(\cos(\eta)-1) - 242\sin(\phi)(\cos(\eta)+1) + 484 = l_j^2 \tag{3.26}$$

$$\begin{aligned}\mathcal{G}_{2+3i} : 121(\sqrt{3}\sin(\eta)(\cos(\phi)+\frac{\sin(\phi)}{2}+\sin(\eta)) + \sin(\phi)(\cos(\eta)+1) - \frac{\cos(\phi)}{2} \\ + \cos(\eta)(2\cos(\phi)-\frac{3}{2})+4) = l_j^2\end{aligned} \tag{3.27}$$

$$\begin{aligned}\mathcal{G}_{3+3i} : 121(\sin(\phi)(\cos(\eta)+1) - \sqrt{3}\sin(\eta)(\cos(\phi)+\frac{\sin(\phi)}{2}+\sin(\eta)) - \frac{\cos(\phi)}{2} \\ + \cos(\eta)(2\cos(\phi)-\frac{3}{2})+4) = l_j^2\end{aligned} \tag{3.28}$$

where for $i = 0$ and 1, $l_j = l_{min}$ and l_{max}

The joint limits are essential to identify the singularity boundaries for the mechanism as well as feasible workspaces. The CAD algorithm along with the Groebner base elimination technique of the parametric root finding option of Maple are employed to solve the system of parallel singularities equations. This is carried out by the *CellDecompositionPlus* function of SIROPA. The trigonometric relations are transformed into algebraic expressions by the transformations: $t = \tan(\eta/2)$ and $t = \tan(\phi/2)$. The parallel singularity equations, Eqs. (3.21) to (3.23) and the joint limit equations, Eqs. (3.26) to (3.28) are transformed as inequalities for isolating the aspects around the home-pose.

3.4.2 Workspace obtained from CAD algorithm

For solving the system of equations using CAD algorithm, the joint limits for the 3-SPS-U mechanism are set as $[l_{min}, l_{max}]$ = [10 , 30] mm. The lower limits for the mechanism contributes mainly to the boundaries of the aspects centered around $\eta = \phi = 0$ radians. The initial analysis of workspaces generated by the CAD algorithm is represented in Figures 3.14a and 3.14b. In Figure 3.14a, the orange regions indicate the feasible workspaces for the mechanism where there exists one solution to the DKP. The white regions are the zones where there exists no solutions. Around the home-pose, a larger workspace could be observed. Beyond the singularity zones, it is

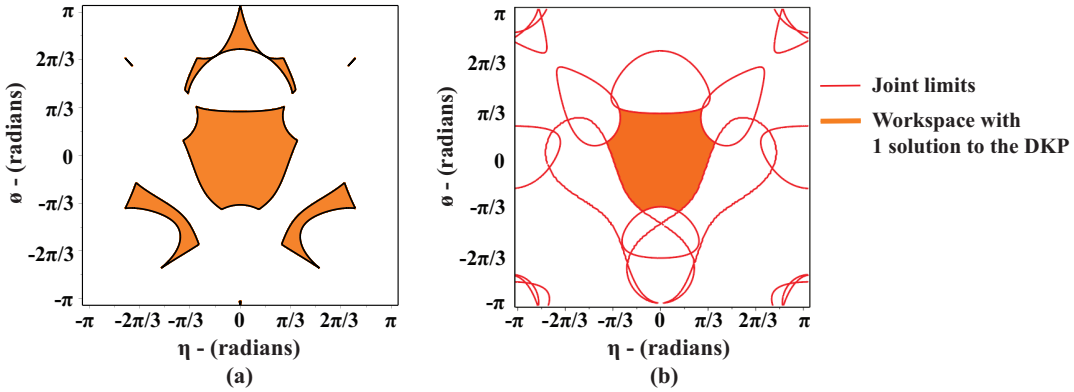


Figure 3.14 – Representation of (a) Singularities and workspace zones for the mechanism at l_{min} = 10 mm and (b) extraction of feasible workspace around home-pose with the joint limits at l_{min} = 10 mm

also possible to identify solutions for the DKP. These regions are not feasible as there are chances that the mechanism might be locked in singular poses while crossing over boundaries, especially while working under passive modes. Thus, the workspace around the home-pose is isolated and they are represented in Figure 3.14b. The joint limits equations are represented by red colored lines for the three springs and they also form the boundaries of the workspace around the home-pose. It could also be observed that the singularity boundaries for the workspace around the home-pose do not have a regular geometry. The singularity free workspace for the mechanism is bounded by a triangular region for the 3-SPS-U mechanism. The minimum length to which the springs can reach is found by substituting $\phi = \eta$ in one of the three lower limit constraint equations. This substitution is done in Eq. (3.26) for the length l_1 and the resulting equation is solved to identify the value of η, which is found to be $-\pi/3$ radians. The closest edge of the triangle from $[\eta, \phi] = [-\pi/3, -\pi/3]$ radians is located at $[\eta, \phi] = [0, -\pi/3]$ radians. At this position of the triangle, one or two of the springs reach their minimum position with no singular posture. Using Eqs. (3.18a) to (3.18b), the values of l_1, l_2 and l_3 are found to be 30 mm, 13.5 mm and 13.5 mm at $[\eta, \phi] = [0, -\pi/3]$ radians. The length of 13.5 mm at $[\eta, \phi] = [0, -\pi/3]$ radians is the minimum limit to which the prismatic springs can reach to avoid parallel singularities. For determining the remaining edges of the workspace triangle, one of the lengths (l_2 or l_3) is fixed at 30 mm while the other two lengths ($l_1 - l_3$ or $l_2 - l_1$) are fixed at 13.5 mm. The system of equations can be written as:

$$\mathcal{L}_1 : f_1(\eta, \phi) = l_{min}^2 \ , \ \mathcal{L}_2 : f_2(\eta, \phi) = l_{max}^2 \ , \ \mathcal{L}_3 : f_3(\eta, \phi) = l_{min}^2 \tag{3.29}$$

$$\mathcal{L}_1 : f_1(\eta, \phi) = l_{min}^2 \ , \ \mathcal{L}_2 : f_2(\eta, \phi) = l_{min}^2 \ , \ \mathcal{L}_3 : f_3(\eta, \phi) = l_{max}^2 \tag{3.30}$$

$$\text{where } [l_{min}, l_{max}] = [13.5, 30]$$

It could be seen that a system with three equations and two unknowns is obtained. As the tensegrity mechanism has redundancy for a given tilt angle, the equations with equal length pairs are subtracted from each other and solved with the third length and they are given by:

$$\mathcal{L}_{1-3} : f_1(\eta, \phi) - f_3(\eta, \phi) = 0 \ , \ \mathcal{L}_2 : f_2(\eta, \phi) = l_{max}^2 \tag{3.31}$$

$$\mathcal{L}_{1-2} : f_1(\eta, \phi) - f_2(\eta, \phi) = 0 \ , \ \mathcal{L}_2 : f_3(\eta, \phi) = l_{max}^2 \tag{3.32}$$

By solving Eqs. (3.31) and (3.32) separately, eight possible solutions for the pose variables are obtained for each set of equation. By eliminating the complex solutions and the solutions beyond singularity boundaries, the values of pose variables that form the remaining two edges of the workspace triangle are found to be $[\eta, \phi] = [\pm\pi/3, 0.57]$ radians. The joint limits for the mechanism are modified as $[l_{min}, l_{max}] = [13.5 \ , \ 30]$ mm and the workspace is recalculated using the CAD algorithm. The results of the CAD algorithm with the modified joint limits are represented below in Figure 3.15a.

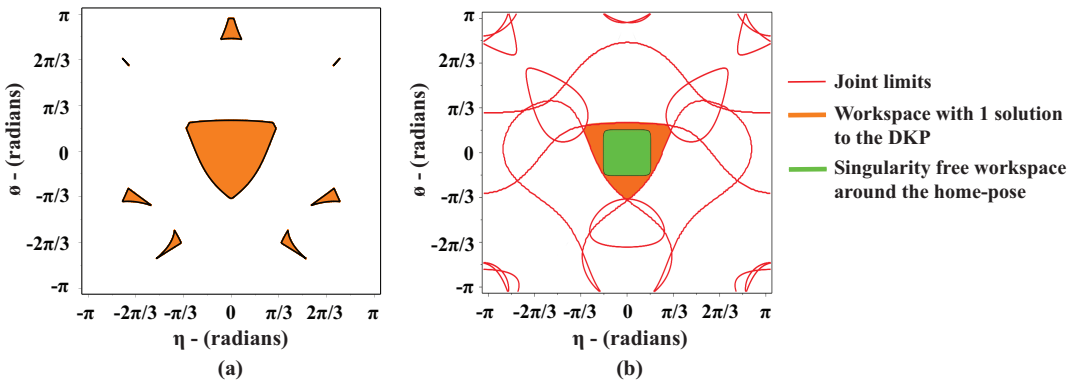


Figure 3.15 – Representation of (a) Singularities and workspace zones for the mechanism at $l_{min} = 13.5$ mm and (b) extraction of feasible workspace and singularity free workspace around home-pose with the joint limits at $l_{min} = 13.5$ mm

It could be observed that a singularity free workspace in the form of a Reuleaux triangle is obtained. The solutions obtained beyond the singularity boundaries are considerably smaller than the previous analysis and they are neglected. The joint limit equations (red colored lines)along with the feasible workspace around the home-pose are represented in Figure 3.15b. The total number of cells obtained by CAD algorithm for the initial and modified joint limits from Figures 3.14 and 3.15 are provided in Table 3.3. The postures of the tensegrity mechanism is constructed using the *CreateManipulator* function of SIROPA library in Maple. By using the *PlotRobot3D* function, the postures of the mechanism under working conditions and singular configurations are represented in Figures 3.16a to 3.16d. In the event of active compliance, it is possible to gain control over the mechanism through cable forces for operating within the workspace around the home-pose by avoiding singularities. However, in the event of passive compliance, the mechanism must be autonomous to overcome pipe bends at 90° when coupled

Architecture	**Joint limits** (mm)	**0 solutions**	**1 solution**	**Total cells**
3-SPS-U	$[l_{min}, l_{max}] = [10\ ,\ 30]$	2852	284	3136
	$[l_{min}, l_{max}] = [13.5\ ,\ 30]$	3362	176	3538

Table 3.3 – Number of cells obtained by CAD algorithm for the 3-SPS-U mechanism under initial and modified joint limits

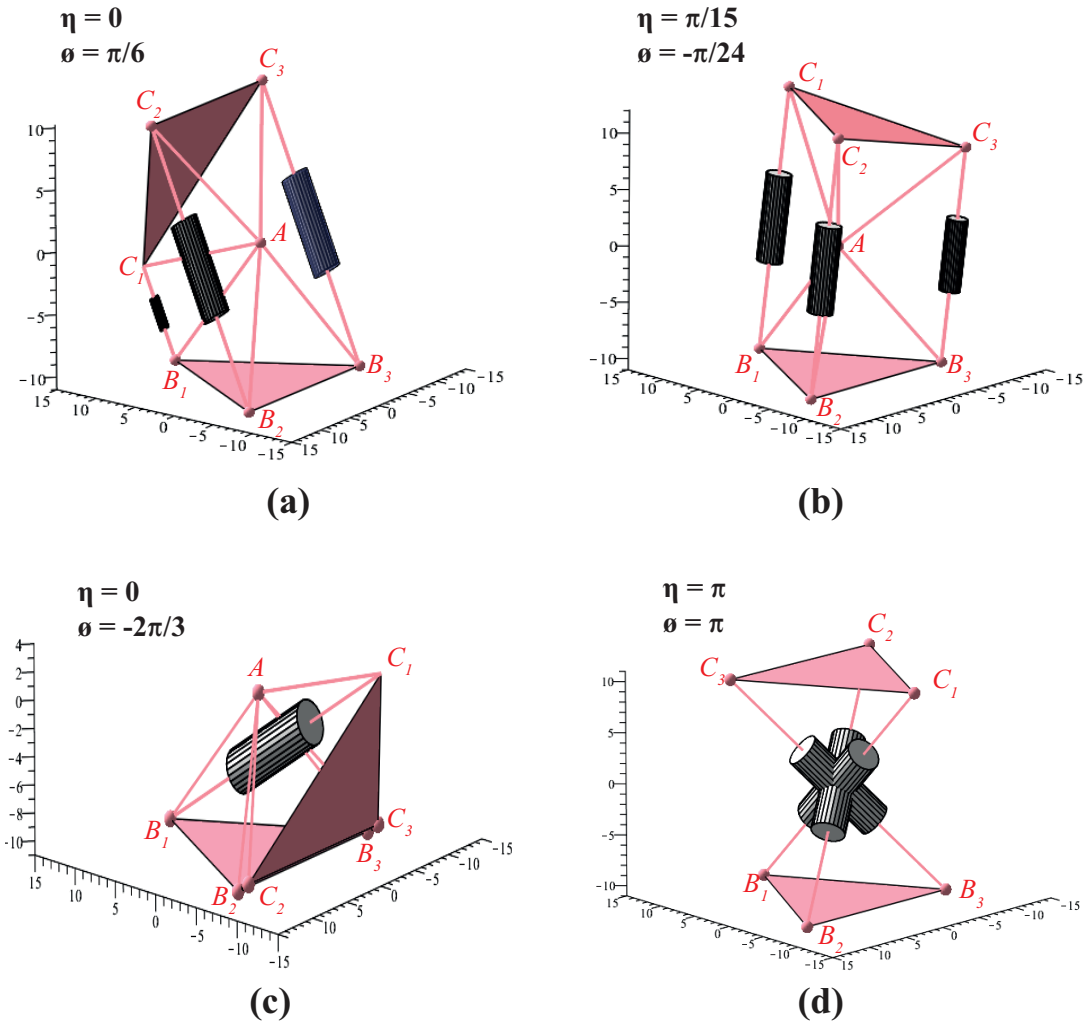


Figure 3.16 – Representation of the 3-SPS-U mechanism in SIROPA (a),(b): under working conditions within the singularity free workspace and (c),(d): in the singularity regions

with the bio-inspired robot. In such a scenario, it is highly likely that the mechanism can attain singular poses owing to the motor modules attached to the system. A singularity free workspace in the form of a square is isolated around the home-pose for tilt limits $[\eta, \phi] \in [\pm\pi/6, \pm\pi/6]$ radians by the equation:

$$\eta^N + \phi^N = \pi/6^N \ , \text{ where } \ N = \ \ 10 \tag{3.33}$$

Using Eq. (3.33), which is called the Lamé equation [Fon16], geometry in the form of a squircle that has an intermediate shape between a square and circle is generated and superimposed on the workspace around the home-pose. This singularity free workspace is depicted in green color in Figure 3.15b. Under passive modes, this green zone, which is bounded by the tilt range of $[\eta, \phi] = [\pm\pi/6, \pm\pi/6]$ is the safe limit for the mechanism to operate within the singularity boundaries. The lower limit of the spring that will be chosen for the mechanism is also an essential design parameter that determines if the mechanism can safely operate within the singularity free workspace during the event of passive compliance.

3.5 Modified architecture of the mechanism

From the singularity analysis of 3-SPS-U mechanism, it was possible to identify the maximum tilt limits within the singularity free workspace. However, when the joint limits of the mechanism were modified, the tilt limits were significantly reduced. By the addition of another tension spring, the mechanism is converted into a 4-SPS-U architecture. The singularity analysis is revisited for this modified architecture in order to verify if higher tilt limits could be obtained within the singularity free workspace for operations under passive modes.

3.5.1 Geometrical equations of the mechanism

Similar to the 3-SPS-U architecture, the tensegrity mechanism with four springs are analyzed by correlating it to a parallel manipulator of type 4-SPS-U [PCL19]. The representation of the tensegrity mechanism and the correlation to a parallel manipulator is shown below in Figure 3.17

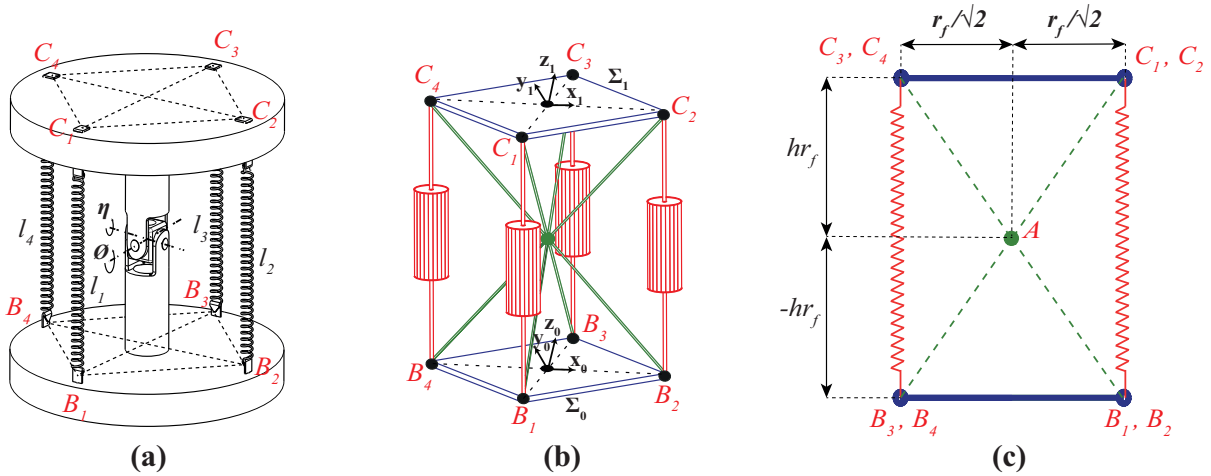


Figure 3.17 – Representation of the (a) Tensegrity mechanism at home-pose, (b) 3D view of the correlation to a 4-SPS-U manipulator and (c) 2D view of the manipulator

The home-pose of the 4-SPS-U tensegrity mechanism is represented in Figure 3.17a where the tilt angles η and ϕ are zero. The fixed coordinate frame of the base is represented by $\sum_0$

with the origin at B_0. The base mounting points for the springs, B_i $(i = 1, 2, 3, 4)$, forms an imaginary square for the 4-S$\underline{\text{P}}$S-U mechanism. The diagonal length for the square is given by $2r_f$. The vector coordinates for the base mounting points are given by:

$$\mathbf{b}_i^0 = \mathbf{R}_z(\theta)\ \mathbf{p}_i^0\ , \quad \text{with} \quad i = 1, 2, 3, 4 \tag{3.34}$$

$$\text{where} \quad \mathbf{R}_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \& \quad \mathbf{p}_i^0 = \begin{bmatrix} r_f \cos\left(\frac{\pi(i-1)}{2}\right) \\ r_f \sin\left(\frac{\pi(i-1)}{2}\right) \\ -r_f h \end{bmatrix}$$

In Eq. (3.34), $\mathbf{p}_i^0$ are the coordinates of B_i in the fixed coordinate frame $\sum_0$. The moving coordinate frame of the end-effector is represented by $\sum_1$ with the origin at C_0. The vector coordinates of C_i are calculated using the transformation matrix $\mathbf{E}$ from Eq. (3.16) of universal joint about the point A. The equations are given by:

$$\mathbf{c}_i^1 = \mathbf{E}\ \mathbf{q}_i^1, \quad \text{with} \quad i = 1, 2, 3, 4 \tag{3.35}$$

$$\text{where} \quad \mathbf{q}_i^1 = \begin{bmatrix} r_f \cos\left(\frac{\pi(i-1)}{2}\right) \\ r_f \sin\left(\frac{\pi(i-1)}{2}\right) \\ r_f h \end{bmatrix}$$

The IKP, which maps the base to the end-effector of the mechanism can be solved by computing the distance between vector coordinates of B_i and C_i. The constraint equations are given by:

$$2r_f^2(h\cos(\eta)(h\cos(\phi) - \sin(\phi)) + h^2 - h\sin(\phi) - \cos(\phi) + 1) = l_1^2 \tag{3.36a}$$

$$2r_f^2(\cos(\eta)(h^2\cos(\phi) - 1) + h\sin(\eta)(\cos(\phi) + 1) + h^2 + 1) = l_2^2 \tag{3.36b}$$

$$2r_f^2(h\cos(\eta)(h\cos(\phi) + \sin(\phi)) + h^2 + h\sin(\phi) - \cos(\phi) + 1) = l_3^2 \tag{3.36c}$$

$$2r_f^2(\cos(\eta)(h^2\cos(\phi) - 1) - h\sin(\eta)(\cos(\phi) + 1) + h^2 + 1) = l_4^2 \tag{3.36d}$$

3.5.2 Singularities and joint limits equations

For isolating the feasible workspace, the singularities and constraint equations are created for the 4-S$\underline{\text{P}}$S-U tensegrity mechanism. The parallel and serial singularities are checked for the mechanism by equating the determinants of the direct and inverse kinematic matrices to zero. The pose variables are the tilt angles η and ϕ and the articular variables are the lengths l_1, l_2, l_3 and l_4. Similar to the 3-S$\underline{\text{P}}$S-U mechanism, the matrix $\mathbf{A}$ does not correspond to a $n \times n$ square matrix. Thus, the 4-S$\underline{\text{P}}$S-U mechanism is subdivided into four sets of 2-S$\underline{\text{P}}$S-U architectures that corresponds to the length pairs $l_1 - l_2$, $l_2 - l_3$, $l_3 - l_4$ and $l_1 - l_4$. The direct kinematic

matrices are created for the four sets by differentiating Eqs. (3.36a) to (3.36d) with respect to the pose variables and the matrices are given by:

$$\mathbf{A}_{1-2} = \begin{bmatrix} \frac{\partial L_1}{\partial \eta} & \frac{\partial L_1}{\partial \phi} \\ \frac{\partial L_2}{\partial \eta} & \frac{\partial L_2}{\partial \phi} \end{bmatrix}, \mathbf{A}_{2-3} = \begin{bmatrix} \frac{\partial L_2}{\partial \eta} & \frac{\partial L_2}{\partial \phi} \\ \frac{\partial L_3}{\partial \eta} & \frac{\partial L_3}{\partial \phi} \end{bmatrix}, \mathbf{A}_{3-4} = \begin{bmatrix} \frac{\partial L_3}{\partial \eta} & \frac{\partial L_3}{\partial \phi} \\ \frac{\partial L_4}{\partial \eta} & \frac{\partial L_4}{\partial \phi} \end{bmatrix}, \mathbf{A}_{4-1} = \begin{bmatrix} \frac{\partial L_4}{\partial \eta} & \frac{\partial L_4}{\partial \phi} \\ \frac{\partial L_1}{\partial \eta} & \frac{\partial L_1}{\partial \phi} \end{bmatrix} \tag{3.37}$$

The parameters L_1, L_2, L_3 and L_4 represents the LHS of Eqs. (3.36a) to (3.36d). By equating the determinants of Eq. (3.37) to zero, the parallel singularities of the mechanism can be analyzed. The determinant values are given by:

$$\begin{aligned} \det(\mathbf{A}_{1-2}), \mathcal{J}_{p1} : & h^2\cos(\eta)^2(h\sin(\phi)+\cos(\phi)+1)+\cos(\phi)^2(h^2-h^3\sin(\eta))-\sin(\eta)\sin(\phi) \\ & -h\cos(\eta)(h\sin(\eta)(h-\sin(\phi))+\sin(\phi)-h\cos(\phi)^2+\cos(\phi)(\sin(\phi)-h-\sin(\eta))) \\ & -h^2+\cos(\phi)(h^3\sin(\phi)+\sin(\eta)(h^2\sin(\phi)+h))=0 \end{aligned} \tag{3.38}$$

$$\begin{aligned} \det(\mathbf{A}_{2-3}), \mathcal{J}_{p2} : & h^2\cos(\eta)^2(h\sin(\phi)-\cos(\phi)-1)+\cos(\phi)^2(h^3\sin(\eta)-h^2)-\sin(\eta)\sin(\phi) \\ & +h\cos(\eta)(h\sin(\eta)(h+\sin(\phi))-\sin(\phi)-h\cos(\phi)^2-\cos(\phi)(\sin(\phi)+h+\sin(\eta))) \\ & +h^2+\cos(\phi)(h^3\sin(\phi)+\sin(\eta)(h^2\sin(\phi)-h))=0 \end{aligned} \tag{3.39}$$

$$\begin{aligned} \det(\mathbf{A}_{3-4}), \mathcal{J}_{p3} : & h^2\cos(\eta)^2(h\sin(\phi)-\cos(\phi)-1)-\cos(\phi)^2(h^3\sin(\eta)+h^2)+\sin(\eta)\sin(\phi) \\ & -h\cos(\eta)(h\sin(\eta)(h+\sin(\phi))+\sin(\phi)+h\cos(\phi)^2+\cos(\phi)(\sin(\phi)+h-\sin(\eta))) \\ & +h^2+\cos(\phi)(h^3\sin(\phi)+\sin(\eta)(h-h^2\sin(\phi)))=0 \end{aligned} \tag{3.40}$$

$$\begin{aligned} \det(\mathbf{A}_{4-1}), \mathcal{J}_{p4} : & h^2\cos(\eta)^2(h\sin(\phi)+\cos(\phi)+1)+\cos(\phi)^2(h^3\sin(\eta)+h^2)+\sin(\eta)\sin(\phi) \\ & +h\cos(\eta)(h\sin(\eta)(h-\sin(\phi))-\sin(\phi)+h\cos(\phi)^2+\cos(\phi)(h-\sin(\phi)-\sin(\eta))) \\ & -h^2+\cos(\phi)(h^3\sin(\phi)-\sin(\eta)(h+h^2\sin(\phi)))=0 \end{aligned} \tag{3.41}$$

Using Eqs. (3.38) to (3.41), the parallel singularities can be studied and the workspaces with possible solutions for the pose variables can be extracted. Similarly, the inverse kinematics matrices, $\mathbf{B}$ are created by differentiating RHS of Eqs. (3.36a) to (3.36d) with respect to the articular variables. Serial singularities are verified by equating the determinant of these matrices to zero. The matrices and the determinant values are given by:

$$\mathbf{B}_{1-2} = \begin{bmatrix} \frac{\partial R_1}{\partial l_1} & \frac{\partial R_1}{\partial l_2} \\ \frac{\partial R_2}{\partial l_1} & \frac{\partial R_2}{\partial l_2} \end{bmatrix}, \mathbf{B}_{2-3} = \begin{bmatrix} \frac{\partial R_2}{\partial l_2} & \frac{\partial R_2}{\partial l_3} \\ \frac{\partial R_3}{\partial l_2} & \frac{\partial R_3}{\partial l_3} \end{bmatrix}, \mathbf{B}_{3-4} = \begin{bmatrix} \frac{\partial R_3}{\partial l_3} & \frac{\partial R_3}{\partial l_4} \\ \frac{\partial R_4}{\partial l_3} & \frac{\partial R_4}{\partial l_4} \end{bmatrix}, \mathbf{B}_{4-1} = \begin{bmatrix} \frac{\partial R_4}{\partial l_4} & \frac{\partial R_4}{\partial l_1} \\ \frac{\partial R_1}{\partial l_4} & \frac{\partial R_1}{\partial l_1} \end{bmatrix} \tag{3.42}$$

$$\det(\mathbf{B}_{1-2}), \mathcal{J}_{s1} : l_1 l_2 = 0, \quad \det(\mathbf{B}_{2-3}), \mathcal{J}_{s2} : l_2 l_3 = 0 \tag{3.43}$$

$$\det(\mathbf{B}_{3-4}), \mathcal{J}_{s3} : l_3 l_1 = 0, \quad \det(\mathbf{B}_{4-1}), \mathcal{J}_{s4} : l_4 l_1 = 0 \tag{3.44}$$

In Eq. (3.42), the parameters R_1, R_2, R_3 and R_4 represents the RHS of Eqs. (3.36a) to (3.36d). Similar to the 3-S$\underline{\text{P}}$S-U mechanism, serial singularities can occur only when the length of one or two prismatic springs is zero for the 4-S$\underline{\text{P}}$S-U mechanism. Thus, the determinant of matrix **B** does not vanish and there exists no serial singularities in the mechanism. Using the SIROPA library of Maple, Eqs. (3.38) to (3.41) are generated with *ParallelSingularities* function. The values of r_f and h are set as 11 mm and 1 for transforming Eqs. (3.36a) to (3.36b) into eight constraint equations which corresponds to the joint limits. These equations are generated using the *ConstraintEquations* function of SIROPA and they are given by:

$$\mathcal{H}_{1+4i} : 242(\cos(\phi)(\cos(\eta) - 1) - \sin(\phi)(\cos(\eta) + 1) + 2) = l_j^2 \tag{3.45}$$

$$\mathcal{H}_{2+4i} : 242(\cos(\eta)(\cos(\phi) - 1) + \sin(\eta)(\cos(\phi) + 1) + 2) = l_j^2 \tag{3.46}$$

$$\mathcal{H}_{3+4i} : 242(\cos(\phi)(\cos(\eta) - 1) + \sin(\phi)(\cos(\eta) + 1) + 2) = l_j^2 \tag{3.47}$$

$$\mathcal{H}_{4+4i} : 242(\cos(\eta)(\cos(\phi) - 1) - \sin(\eta)(\cos(\phi) + 1) + 2) = l_j^2 \tag{3.48}$$

$$\text{where for } i = \ 0 \text{ and } 1, \ l_j = \ l_{min} \text{ and } l_{max}$$

The joint limits equations and the singularity equations are solved using the CAD algorithm in Maple. The parallel singularity equations, Eqs. (3.38) to (3.41) and the joint limits equations, Eqs. (3.45) to (3.48) are transformed as inequalities in CAD algorithm for isolating the aspects around the home-pose.

3.5.3 Workspace analysis

For the CAD algorithm, the joint limits for the 4-S$\underline{\text{P}}$S-U mechanism are set as $[l_{min}, l_{min}]$ = [10 , 30] mm. The initial analysis of workspaces generated by CAD algorithm is shown in Figure 3.18a. Unlike the 3-S$\underline{\text{P}}$S-U mechanism, feasible workspaces are obtained around the home-pose with no possible solutions beyond the singularity boundaries for the 4-S$\underline{\text{P}}$S-U mechanism. The reddish-brown region depicts the feasible workspace for the 4-S$\underline{\text{P}}$S-U mechanism where there exists one solution to the DKP. The joint limits equations are represented by colored lines and they do not appear superimposed like the 3-S$\underline{\text{P}}$S-U mechanism. A uniform workspace is not obtained when the lower joint limit is set at 10 mm. Thus, the workspace with singularity free boundaries must be extracted. From the workspace obtained in Figure 3.18a, two of the four lengths of the springs reach a minimum limit when the values of η and ϕ become equal. By setting $\eta = \phi$ in Eq. (3.36a), l_1 can be rewritten as:

$$l_1 = 11\sqrt{4 - 2\cos(\phi)\sin(\phi) + 2\sin(\phi)^2 - 2\cos(\phi) - 2\sin(\phi)} \tag{3.49}$$

The minimum value of l_1 can be obtained by differentiating Eq. (3.49) with respect to ϕ

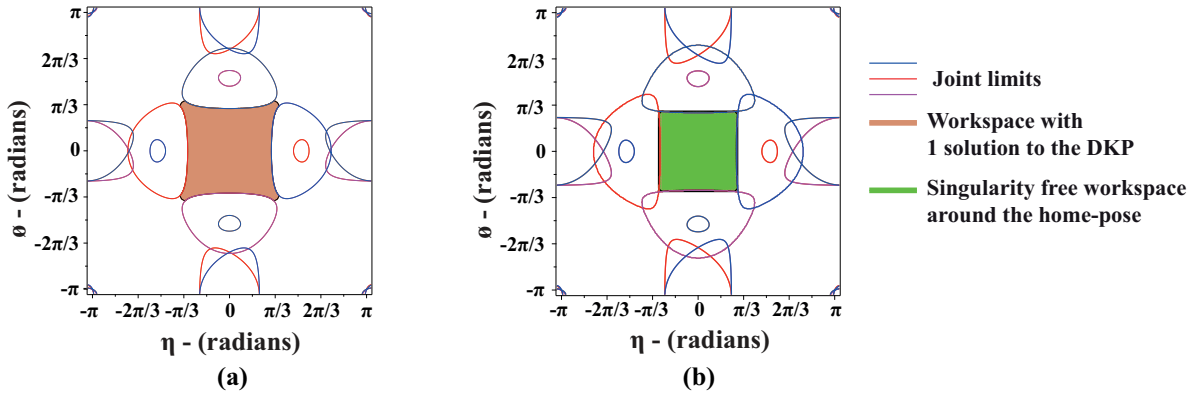


Figure 3.18 – Representation of the (a) Feasible workspace with joint limits at $l_{min} = 10$ mm and (b) Singularity free workspace with joint limits at $l_{min} = 11.1$ mm for the 4-SPS-U mechanism

and equating it to zero, for which a value of $\pm 5\pi/18$ radians is obtained. At this value of η and ϕ, the minimum possible length that could be attained by a spring is found to be 11.1 mm. This is the minimum limit to which the springs can reach to work within the workspace by avoiding parallel singularities. By modifying the joint limits as $[l_{min}, l_{min}] = [11.1\ ,\ 30]$ mm, the singularity analysis is carried out again using the CAD algorithm and the feasible workspaces are represented in Figure 3.18b. A uniform workspace in the form of a square could be observed for the 4-SPS-U architecture when the joint limits are modified. The singularity free workspace which is represented in green color appears superimposed on the workspace (reddish-brown region) generated by CAD algorithm. The maximum tilt limits for the mechanism are found to be $\pm 5\pi/18$ radians. The solution for the IKP at the tilt values of $\pm\pi/4$ (arbitrary) for each actuated prismatic link is provided below in Table 3.4.

Tilt angles (rad)		**Solutions to IKP** (mm)			
η	ϕ	l_1	l_2	l_3	l_4
0	$\pi/4$	11.9	20.32	28.74	20.32
0	-$\pi/4$	28.74	20.32	11.9	20.32
$\pi/4$	0	20.32	28.74	20.32	11.9
-$\pi/4$	0	20.32	11.9	20.32	28.74
$\pi/4$	$\pi/4$	11.9	26.9	26.9	11.9
-$\pi/4$	-$\pi/4$	26.9	11.9	11.9	26.9
$\pi/4$	-$\pi/4$	26.9	26.9	11.9	11.9
-$\pi/4$	$\pi/4$	11.9	11.9	26.9	26.9

Table 3.4 – Solutions to the IKP for varying input tilt angles

Using the *PlotRobot3D* function of SIROPA, the postures of the manipulator during home-pose and for some working conditions from Table 3.4 are demonstrated in Figures 3.19(a) to 3.19(d). The total number of cells obtained by CAD algorithm for the initial and modified joint limits are provided in Table 3.5. Compared to the 3-SPS-U mechanism, a uniform workspace

Architecture	**Joint limits** (mm)	**0 solutions**	**1 solution**	**Total cells**
4-SPS-U	$[l_{min}, l_{max}] = [10\ ,\ 30]$	3366	76	3442
	$[l_{min}, l_{max}] = [11.1\ ,\ 30]$	3506	52	3558

Table 3.5 – Number of cells obtained by CAD algorithm for the 4-SPS-U mechanism with initial and modified joint limits

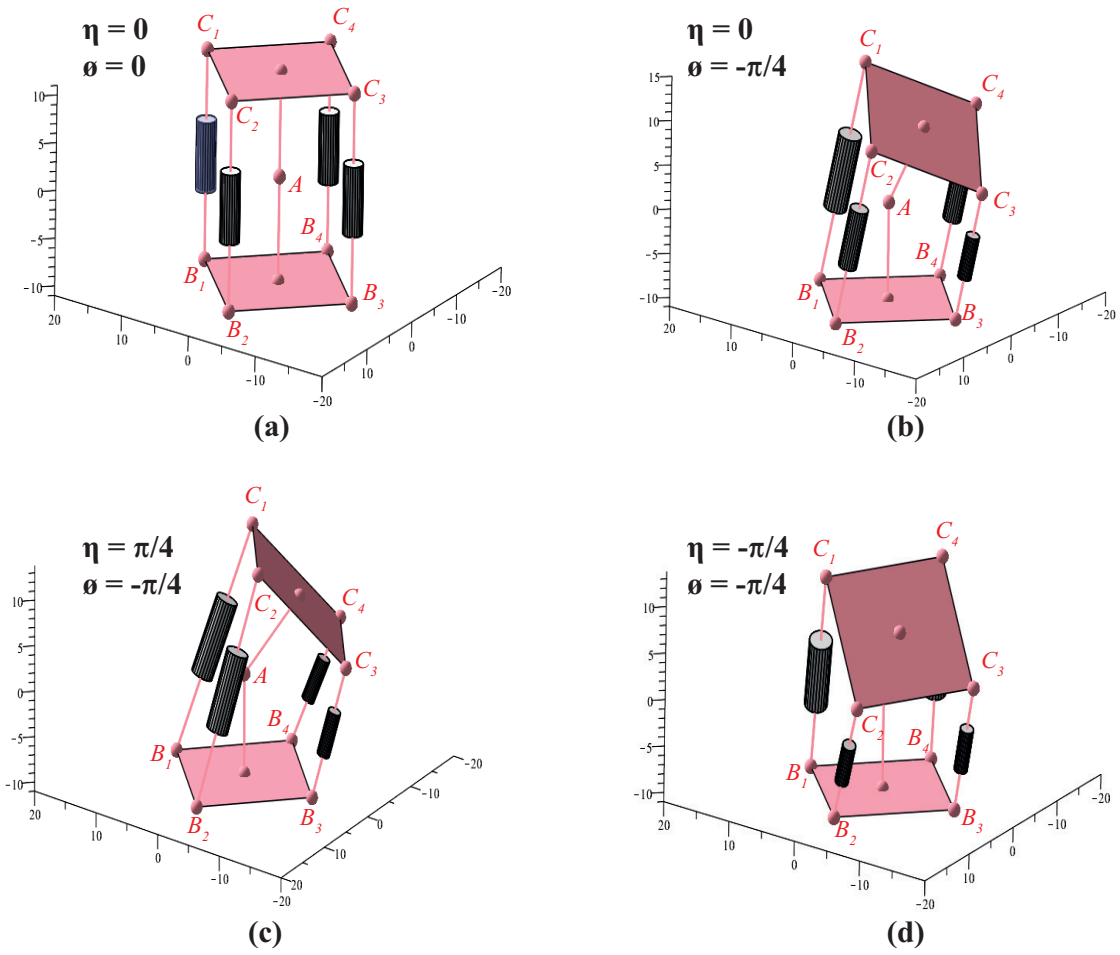


Figure 3.19 – Representation of the 4-SPS-U mechanism in SIROPA at (a)home-pose and (b),(c),(d): under working conditions

with higher tilt limits is obtained for the 4-SPS-U mechanism. Even though the redundancy of the system is increased, the mechanism appears more rigid and can operate autonomously within the singularity free workspace in the event of passive compliance. For the problem of active compliance, both architectures are suitable as cable forces can be applied to operate the mechanisms smoothly. However, for passive compliance, the 4-SPS-U mechanism is an ideal candidate over the 3-SPS-U mechanism for integrating along with the bio-inspired robot studied in Chapter-2.

3.6 Stability analysis of the mechanism

In the previous sections, a stability analysis of the 3-S$\underline{P}$S-U mechanism was carried out by using the Tilt and Azimuth angles. It was found that only the tilt angle α ensures stability while the azimuth has no influences on the mechanism, especially when there is no external loading. However, with respect to the Euler angles of the universal joint, about the central point A of the mechanism, the tilt angles η and ϕ contribute equally to the stability of the mechanism under natural conditions. Thus, the static analysis is reverified with respect to the universal joint for the 3-S$\underline{P}$S-U and 4-S$\underline{P}$S-U tensegrity mechanisms. The Lagrange equation of motion [Mei10] from Eq. (3.11) is verified at static conditions for the pose variables $\mathbf{q} = [\eta, \phi]^T$. Under static conditions, the total potential energy of the tensegrity mechanisms is given by:

$$U_{c1} = \sum_{i=1}^{3} F_i l_i \quad U_{s1} = \sum_{i=1}^{3} \frac{1}{2} k (l_i - l_0)^2 \quad \text{for 3-S}\underline{P}\text{S-U} \tag{3.50}$$

$$U_{c2} = \sum_{i=1}^{4} F_i l_i \quad U_{s2} = \sum_{i=1}^{4} \frac{1}{2} k (l_i - l_0)^2 \quad \text{for 4-S}\underline{P}\text{S-U} \tag{3.51}$$

$$U_{3-S\underline{P}S-U} = U_{c1} + U_{s1} \quad , \quad U_{4-S\underline{P}S-U} = U_{c2} + U_{s2} \tag{3.52}$$

In Eqs. (3.50) and (3.51), U_{c1}, U_{c2} are the cable forces and U_{s1}, U_{s2} are the spring forces for the 3-S$\underline{P}$S-U and 4-S$\underline{P}$S-U mechanisms respectively.

3.6.1 Stability analysis for fixed design parameters

The applied forces along the cables are considered as F_1 to F_3 and F_1 to F_4 for the 3-S$\underline{P}$S-U and 4-S$\underline{P}$S-U mechanisms respectively. The total potential energy for both mechanisms is given by $U_{3-S\underline{P}S-U}$ and $U_{4-S\underline{P}S-U}$ in Eq. (3.52). Similar to the stability analysis that was carried out earlier, the springs are considered to be massless and their free lengths l_0 are set as 0 mm for the analysis. The total potential energy for both mechanisms are calculated using Maple with the help of Eqs. (3.50) to (3.52) and they are given by:

$$\begin{aligned} U_{3-S\underline{P}S-U} = \sqrt{2} r_f (F_1 (\cos(\phi)(h^2 \cos(\eta) - 1) - h \sin(\phi)(\cos(\eta) + 1) + h^2 + 1)^{1/2} + F_2 (h^2 + 1 \\ + \frac{h \sin(\phi)}{2}(\cos(\eta) + 1) + \frac{\sqrt{3}}{2} \sin(\eta)(h \cos(\phi) + h + \frac{\sin(\phi)}{2}) - \frac{3}{4} \cos(\eta) \\ + \cos(\phi)(h^2 \cos(\eta) - \frac{1}{4}))^{1/2} + F_3 (h^2 + 1 - \frac{3}{4} \cos(\eta) + \frac{h \sin(\phi)}{2}(\cos(\eta) + 1) \\ + \cos(\phi)(h^2 \cos(\eta) - \frac{1}{4}) - \frac{\sqrt{3}}{2} \sin(\eta)(h \cos(\phi) + h + \frac{\sin(\phi)}{2}))^{1/2}) + \xi_1 \end{aligned} \tag{3.53}$$

$$\begin{aligned} U_{4-S\underline{P}S-U} = \sqrt{2} r_f (F_1 \sqrt{\cos(\phi)(h^2 \cos(\eta) - 1) - h \sin(\phi)(\cos(\eta) + 1) + h^2 + 1} \\ + F_2 \sqrt{\cos(\eta)(h^2 \cos(\phi) - 1) + h \sin(\eta)(\cos(\phi) + 1) + h^2 + 1} \end{aligned}$$

$$+ F_3\sqrt{\cos(\phi)(h^2\cos(\eta) - 1) + h\sin(\phi)(\cos(\eta) + 1) + h^2 + 1}$$
$$+ F_4\sqrt{\cos(\eta)(h^2\cos(\phi) - 1) - h\sin(\eta)(\cos(\phi) + 1) + h^2 + 1)} + \xi_2 \tag{3.54}$$

$$\text{with } \xi_i = jkr_f^2(\cos(\eta)(h^2\cos(\phi) - \frac{1}{2}) + h^2 - \frac{\cos(\phi)}{2} + 1)$$
$$\text{where for } i = [1, 2]\ ,\ j = [3, 4] \text{ for 3-SPS-U and 4-SPS-U}$$

For the initial analysis, the values of r_f and h are retained as 11 mm and 1 respectively [VC19b, VFCW] and a stiffness of k= 0.75 N/mm is chosen for each spring. At the home-pose, where the tilt angles $\eta = \phi = 0$ radians, the total potential energy of the mechanisms is calculated under zero applied forces and their plots are represented below in Figure 3.20.

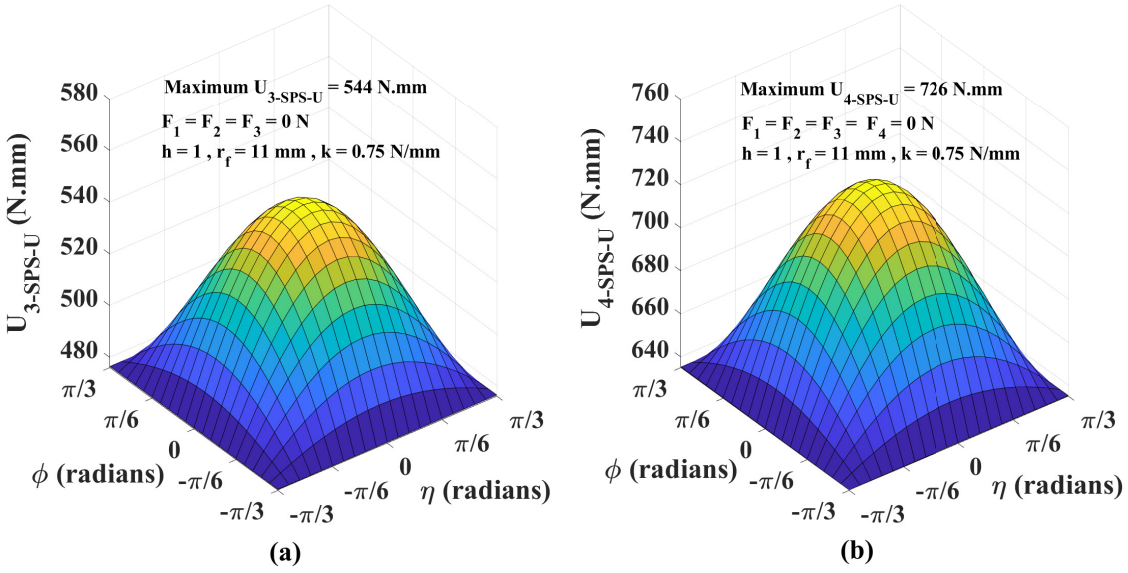


Figure 3.20 – Plot of total potential energy versus the tilt angles η and ϕ at the home-pose for the (a) 3-SPS-U and (b) 4-SPS-U tensegrity mechanisms

From Figure 3.20, it could be observed that an unstable configuration is obtained at the home-pose under the absence of external forces for both mechanisms. These configurations are not desirable, especially when the mechanism is coupled with the piping inspection robot.

3.6.2 Modification of design parameters

For fixed design parameters, it was found that the mechanism remained unstable under natural conditions. In order to determine stable configurations, it is necessary to study the derivatives of the total potential energy against the pose variables. As the mechanism has dependencies on the two pose variables, the following steps are carried out:

- Generation of the Hessian matrix or the mechanism stiffness

- Estimation of the determinant value of the Hessian matrix
- Estimation of the value of the second-order derivative of total potential energy with respect to one of the tilt angles

The Hessian matrix and its determinant are given by:

$$\mathbf{H}_i = \begin{bmatrix} \frac{\partial^2 U_j}{\partial \eta^2} & \frac{\partial^2 U_j}{\partial \eta \partial \phi} \\ \frac{\partial^2 U_j}{\partial \eta \partial \phi} & \frac{\partial^2 U_j}{\partial \phi^2} \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \tag{3.55}$$

$$\det(\mathbf{H}_i) = f_{11} f_{22} - f_{12}^2 \tag{3.56}$$

$$\text{where for } i = [1, 2],\ j = [\text{3-S}\underline{\text{P}}\text{S-U} \text{ , 4-S}\underline{\text{P}}\text{S-U}]$$

As the total potential energy is a function of the two pose variables, the following conditions are possible [Daw03]:

1. The total potential energy has a relative maximum when $\det(\mathbf{H}_i) > 0$ and $f_{11} < 0$
2. The total potential energy has a relative minimum when $\det(\mathbf{H}_i) > 0$ and $f_{11} > 0$

From Figure 3.20, the first condition could be observed for both mechanisms as there exists a relative maximum with $f_{11} < 0$. As the mechanism will be integrated along with the bio-inspired robot of Chapter-2, the value of r_f is retained as 11 mm [VCB19]. When the mechanism is coupled with the bio-inspired robot, there exists a preload on the mechanism due to the presence of motor modules and its cables. The existing prototype weighs around 6.5 N [CVB18]. However, a design modification will be done on the robot to arrive at a compact design with reduced motor sizing for overcoming pipe bends in Chapter-5. Thus, a preloading of 2 N and 1.5 N along each spring for the 3-S$\underline{\text{P}}$S-U and 4-S$\underline{\text{P}}$S-U mechanisms are considered for the analysis. For determining optimal values of h and spring stiffness k under the presence of a preload, a simplified optimization problem using Genetic algorithm is carried out in MATLAB [CF95]. A stochastic approach is being followed for extracting multiple solutions that could be distributed over a domain. The optimization problem is stated as:

$$\begin{aligned} \text{Maximize: } & f_{11}(\mathbf{x}) \\ \text{subject to constraints: } & g_1\text{: } \det(\mathbf{H}) \geq 0 \\ & g_2\text{: } f_{11}(\mathbf{x}) \geq 0 \end{aligned}$$

$$\text{where } \mathbf{x} = [h, k]^T$$

The objective function aims to maximize the derivative f_{11} such that it remains positive along with the determinant of the Hessian matrix throughout the optimization process. The equations for the objective function and the determinant value of the Hessian matrix for both mechanisms are given by:

$$f_{11}(i) = \frac{11j(11k - 22h^2k - F_l h)}{2} \tag{3.57}$$

$$\det(\mathbf{H}_i) = \frac{12j^2(F_l h + 22h^2 - 11k)^2}{4} \tag{3.58}$$

where for $i = [1, 2]$, $j = [3, 4]$ for 3-SPS-U & 4-SPS-U

also for $i = 1$, $F_1 = F_2 = F_3 = F_l = 2$ N for 3-SPS-U

and for $i = 2$, $F_1 = F_2 = F_3 = F_4 = F_l = 1.5$ N for 4-SPS-U

The lower and upper bounds for h are set between 0.2 to 0.8 while for the spring these values are set between 0.1 to 1. By using Eqs. (3.57) to (3.58) and the bounds, the optimization problem is solved using the function *ga* in MATLAB [CF95]. The results of solutions obtained from the optimization problem is represented below in Figure 3.21.

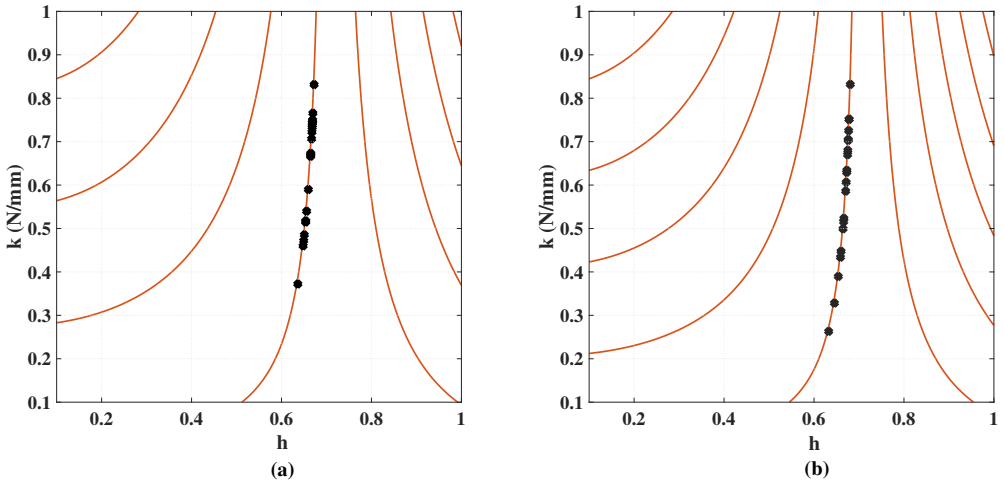


Figure 3.21 – Solutions for optimization problem obtained by Genetic Algorithm for the (a) 3-SPS-U and (b) 4-SPS-U mechanisms

In Figures 3.21a and 3.21b, the solutions obtained by Genetic algorithm in MATLAB are depicted as black colored scatter points over the contour of the objective function. The values of h and k for Figures 3.21a and 3.21b are provided in Tables 3.6 and 3.7. It could be observed that optimum values of h are distributed between 0.6 to 0.7. For each value of h, an optimum value for spring stiffness k is obtained. From the results of Figure 3.21 and Tables 3.6 and 3.7, the value of k that matches with springs available at LS2N is 0.75 N/mm. The rows having values of k closer to 0.75 N/mm are highlighted in green in Tables 3.6 and 3.7. For this value of k, Eq. (3.57) is differentiated with respect to h and equated to zero under external forces of 2 N and 1.5 N for the 3-SPS-U and 4-SPS-U mechanisms respectively. The plot of parameter h versus f_{11} for both mechanisms at k=0.75 N/mm is represented in Figure 3.22.

Parameter h	**Spring stiffness** k (N/mm)	f_{11} (N.mm/rad)	det(**H**)
0.628	0.531	0.366	0.134
0.612	0.433	0.477	0.227
0.549	0.245	0.455	0.207
0.655	0.820	0.489	0.239
0.613	0.438	0.477	0.227
0.636	0.589	0.569	0.323
0.632	0.559	0.447	0.200
0.650	0.742	0.576	0.331
0.546	0.239	0.503	0.253
0.643	0.656	0.609	0.371
0.627	0.520	0.518	0.268
0.636	0.589	0.569	0.323
0.657	0.862	0.294	0.086
0.654	0.804	0.486	0.236
0.646	0.698	0.368	0.135
0.567	0.282	0.437	0.191
0.654	0.806	0.433	0.188
0.641	0.635	0.611	0.373
0.645	0.685	0.404	0.163
0.603	0.393	0.442	0.195

Table 3.6 – Optimum values of k and h for the 3-SPS-U mechanism from Figure 3.21a

Parameter h	**Spring stiffness** k (N/mm)	f_{11} (N.mm/rad)	det(**H**)
0.664	0.749	0.486	0.236
0.663	0.737	0.323	0.104
0.621	0.360	0.567	0.322
0.663	0.735	0.381	0.145
0.669	0.859	0.275	0.076
0.618	0.348	0.506	0.256
0.646	0.524	0.348	0.121
0.663	0.724	0.703	0.494
0.649	0.550	0.441	0.194
0.656	0.629	0.440	0.193
0.657	0.639	0.542	0.293
0.660	0.679	0.665	0.379
0.665	0.761	0.293	0.442
0.663	0.738	0.323	0.086
0.665	0.773	0.363	0.109
0.641	0.482	0.245	0.131
0.667	0.816	0.560	0.060
0.658	0.652	0.556	0.313
0.654	0.601	0.533	0.309
0.635	0.436	0.496	0.284

Table 3.7 – Optimum values of k and h for the 4-SPS-U mechanism from Figure 3.21b

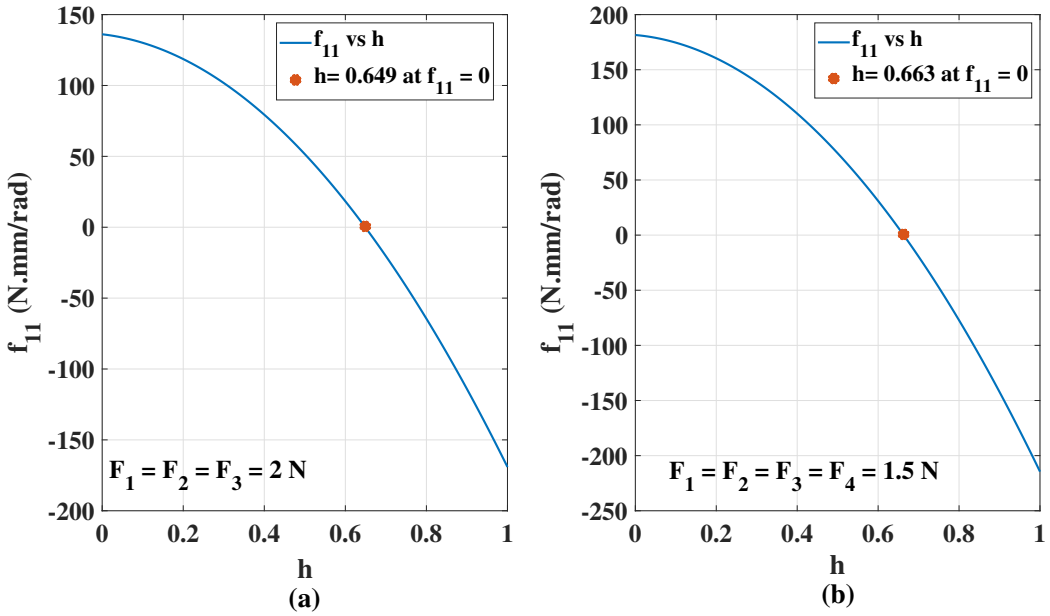


Figure 3.22 – Optimum value of h at spring stiffness of 0.75 N/mm for the (a) 3-SPS-U and (b) 4-SPS-U mechanisms under the presence of external forces

From Figures 3.22a and 3.22b, a value of 0.649 and 0.663 for h is obtained for the 3-SPS-U and 4-SPS-U mechanisms where f_{11} is zero. For ease of calculations and prototyping, a common value of $h = 0.6$ is taken for both the mechanisms. The stability plot for both the mechanisms with the modified design parameters under the presence of a preload is represented below in Figure 3.23.

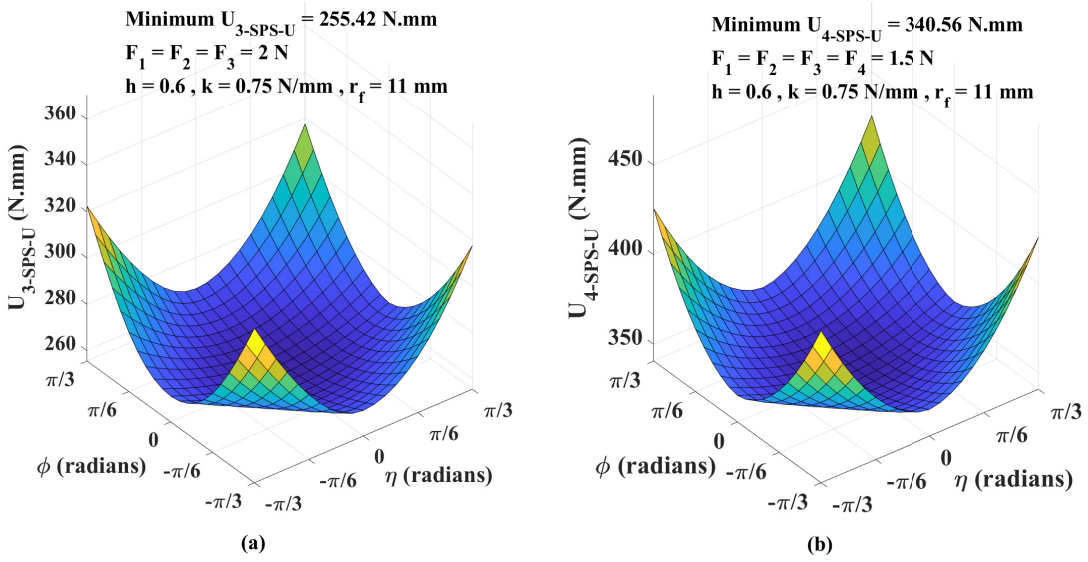
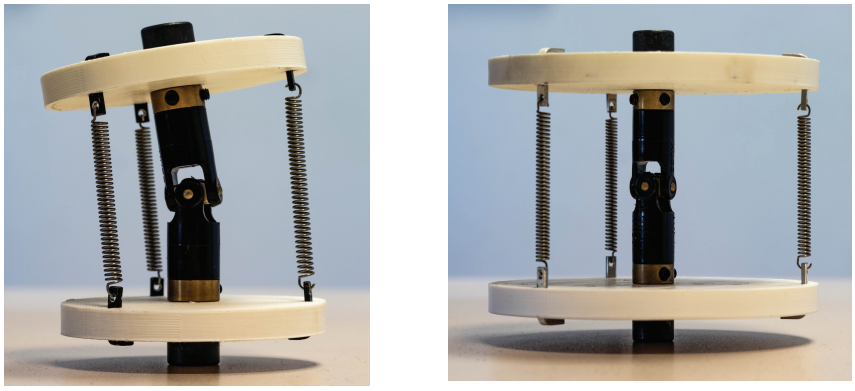


Figure 3.23 – Plot of total potential energy versus the tilt angles η and ϕ for $h = 0.6$ at the home-pose condition for the (a) 3-SPS-U and (b) 4-SPS-U mechanisms

It could be observed from Figures 3.23a and 3.23b that a stable configuration has been obtained for both mechanisms under the presence of a preload. Using standard parts available at LS2N, the 3-S$\underline{P}$S-U tensegrity mechanism is realized. A standard universal joint of 68 mm length and springs of 0.75 N/mm stiffness are employed to assemble the tensegrity mechanism. The base and end-effector are realized by 3D printing. Two prototypes with h= 1 and h= 0.6 are realized where the corresponding values of r_f are 34 mm and 56.7 mm. The prototypes are represented below in Figure 3.24.



(a) **(b)**

Figure 3.24 – Prototypes of the 3-S$\underline{P}$S-U tensegrity mechanism realized at LS2N with (a) $h = 1$ and (b) h=0.6

From Figure 3.24a, it could be observed that under natural conditions, the mechanism remains unstable when h= 1. Upon the design parameter modification of h from 1 to 0.6, a stable configuration could be observed in Figure 3.24b. The stability analysis was carried out by fixing the free lengths as 0 mm. However, upon considering a value of 35 mm as free length for the spring employed in the prototypes depicted in Figure 3.24, the mechanism still remained unstable at h=1 upon numerical analysis, which is not presented here. At the home-pose and under the absence of any external forces, the stability of the mechanism purely depends on the parameter h and the stiffness of the tension spring. The influence of parameter h plays an essential role to determine the overall stability of the mechanism. Since the design parameter h has been modified, the singularity analysis is carried out again to identify the maximum tilt limits for the mechanism. Experimental validation of the mechanism represented in Figure 3.24b will be carried out in Chapter-4 to demonstrate the actuation strategies.

3.6.3 Workspace analysis of the 3-S$\underline{P}$S-U mechanism

For the modified design parameter h, the CAD algorithm is executed using the parallel singularities equations, Eqs. (3.21) to (3.23) and joint limits equations, Eqs. (3.26) to (3.28). Since the height of the mechanism has been reduced by almost one-half at the home-pose, the joint limits are set as $[l_{min}, l_{max}] = [5\ ,\ 30]$ mm. Using $CellDecompositionPlus$ function of SIROPA

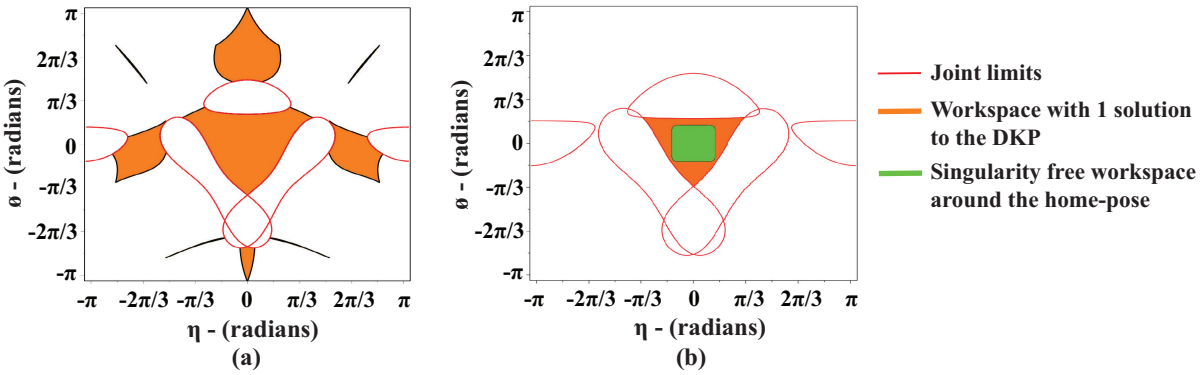


Figure 3.25 – Representation of the (a) Feasible workspace with joint limits at $l_{min} = 5$ mm and (b) Singularity free workspace with joint limits at $l_{min} = 6.5$ mm for the 3-SPS-U mechanism

in Maple, the workspace for the mechanism is generated and represented in Figure 3.25a. The orange regions indicate the feasible workspaces for the 3-SPS-U mechanism where there exists one solution to the DKP. Singularity regions are represented by white zones. Some possible solutions are also obtained beyond the workspace centered around the home-pose. These postures are difficult to attain by the mechanism especially under passive modes as the mechanism has to cross the singularity boundaries. The joint limits equations for the three springs appear superimposed and they are shown in red lines for the mechanism in Figure 3.25a. The regular triangular workspace for the mechanism has to be extracted. Similar to the previous analysis, one edge of the triangle is extracted at $[\eta, \phi] = [0\ ,\ -\pi/3]$ radians. At this position, the values of l_1, l_2 and l_3 are found to be 22 mm, 6.5 mm and 6.5 mm. The length of 6.5 mm at $[\eta, \phi] = [0\ ,\ -\pi/3]$ radians is the minimum value to which the prismatic springs can reach to avoid parallel singularities. For estimating the remaining edges of the triangle, one of the two lengths (l_2 or l_3) are set at 22 mm while the other length pairs ($l_1 - l_3$ or $l_2 - l_1$) are fixed at 6.5 mm. By solving Eqs. (3.29) to (3.32) for $[l_{min}, l_{max}] = [6.5\ ,\ 22]$ mm, eight possible values for η and ϕ are estimated. After eliminating complex solutions and solutions obtained in singularity regions, the remaining edges of the triangle are extracted and are given by $[\eta, \phi] = [\pm\pi/3\ ,\ 0]$ radians. The joint limits for the mechanism are modified as $[l_{min}, l_{max}] = [6.5\ ,\ 30]$ mm and the workspace is recalculated using the CAD algorithm. The results of workspace for the modified joint limits are represented in Figure 3.25b where a workspace in the form of a Reuleaux triangle is obtained. The intermediate solutions obtained beyond the home-pose regions are eliminated. The singularity free workspace that defines a square is isolated around the home-pose where the maximum tilt limits are bounded by $[\eta, \phi] \in [\pm 2\pi/15, \pm 2\pi/15]$ radians. The singularity free workspace is represented in green region in Figure 3.25b with the help of Eq. (3.33) at $N = 2\pi/15$ radians. The total number of cells obtained by CAD algorithm is provided in Table 3.8. With the modified value of the design parameter h, the tilt limits for the 3-SPS-U mechanism are reduced. When this mechanism is coupled with the bio-inspired robot, it may not be possible for the mechanism to overcome pipe bends at $\pi/2$ radians. Also, under

Architecture	**Joint limits** (mm)	**0 solutions**	**1 solution**	**Total cells**
3-SPS-U	$[l_{min}, l_{max}] = [5\ ,\ 30]$	1054	172	1226
	$[l_{min}, l_{max}] = [6.5\ ,\ 30]$	1198	144	1342

Table 3.8 – Number of cells obtained by CAD algorithm for the 3-SPS-U mechanism under initial and modified joint limits at $h = 0.6$

passive modes, there are chances that the mechanism can reach singular configurations while trying to overcome bends. Thus, this architecture will not be a suitable solution for addressing the issue of passive compliance. One solution for this problem can be addressed by using a stacked tensegrity module to attain higher tilt limits. However, the singularity analysis must be reverified for the stacked model for understanding the feasible workspace of the mechanism.

3.6.4 Workspace analysis of the 4-SPS-U mechanism

Similar to the 3-SPS-U mechanism, the singularity analysis is carried out for the 4-SPS-U mechanism with the initial joint limits at $[l_{min}, l_{max}] = [5\ ,\ 30]$ mm for h=0.6. Using the parallel singularities equations, Eqs. (3.38) to (3.41) and joint limits equations, Eqs. (3.45) to (3.48), the workspaces are generated using the CAD algorithm in Maple. The results of workspace generated using *CellDecompositionPlus* function of SIROPA library are represented below in Figure 3.26a.

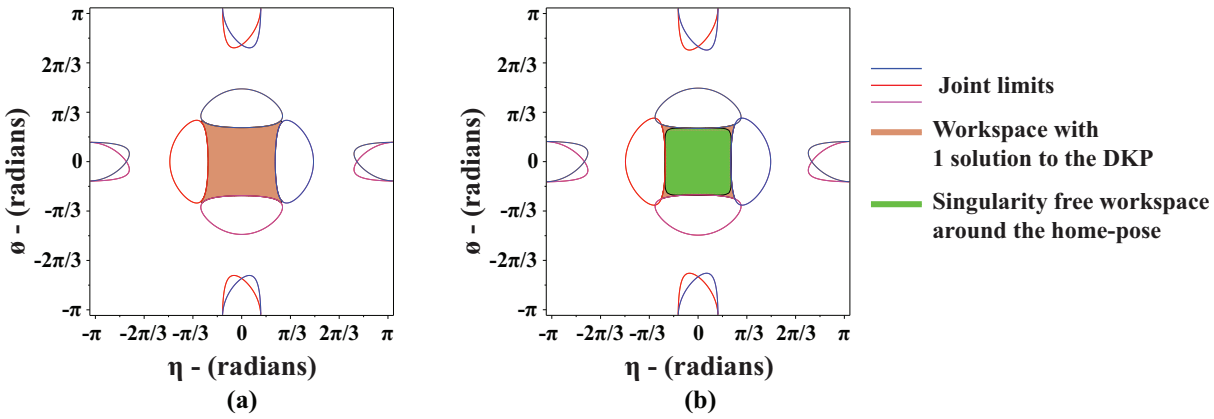


Figure 3.26 – Representation of (a) Feasible workspace and joint limits at $l_{min} = 5$ mm and (b) Singularity free workspace and joint limits at $l_{min} = 5.3$ mm for the 4-SPS-U mechanism

The feasible workspace for the mechanism is centered around the home-pose and they are represented by the reddish-brown regions in Figure 3.26a, where there exists one solution to the DKP. The joint limits are represented by the colored lines. The singularity free workspace in the form of a square must be extracted to identify the maximum tilt limits. It was analyzed earlier that for the 4-SPS-U mechanism, two of the four spring lengths reach a minimum value

when both tilt angles become equal. By setting $\eta = \phi$ and $h = 0.6$ in Eq. (3.36a), l_1 can be written as:

$$l_1 = \frac{11\sqrt{68 - 30\cos(\phi)\sin(\phi) + 18\sin(\phi)^2 - 50\cos(\phi) - 30\sin(\phi)}}{5} \tag{3.59}$$

By differentiating Eq. (3.59) with respect to ϕ and equating it to zero, the minimum value of l_1 could be found. At this length, the tilt angles are found to be $\pm\pi/4$ radians where one or two of the prismatic spring reaches their minimum length of 5.3 mm. By modifying the joint limits to $[l_{min}, l_{max}] = [5.3\ ,\ 30]$ mm, the singularity analysis is carried out again using the CAD algorithm and the results are represented in Figure 3.26b. The singularity free workspace that represents a square is then extracted using Eq. (3.33) at $N = \pi/4$ radians and it is depicted by green region in Figure 3.26b. The maximum tilt limits within the singularity free workspace for the 4-SPS-U mechanism at h= 0.6 is given by $\pm\pi/4$ radians. The total number of cells obtained by CAD algorithm is provided below in Table 3.9.

Architecture	**Joint limits** (mm)	**0 solutions**	**1 solution**	**Total cells**
4-SPS-U	$[l_{min}, l_{max}] = [5\ ,\ 30]$	1388	46	1434
	$[l_{min}, l_{max}] = [5.3\ ,\ 30]$	1496	38	1534

Table 3.9 – Number of cells obtained by CAD algorithm for the 4-SPS-U mechanism under initial and modified joint limits at $h = 0.6$

With the modification of design parameter h, the tilt limits for the 4-SPS-U mechanism has been reduced to $\pm\pi/4$ radians. However, with a uniform workspace and better rigidity, the 4-SPS-U mechanism can operate autonomously within the singularity free workspace. Thus, the 4-SPS-U tensegrity mechanism is an ideal candidate for addressing passive and active compliance issues. However, the reduced tilt limits might pose a problem to overcome pipe bends at 90°. By simulation trails, it has to be determined if a single module or a stacked module has to be employed for the bio-inspired robot to overcome pipe bends, which will be investigated in Chapter-5.

3.7 Conclusions

This chapter addressed the key design issues for a piping inspection robot to overcome pipe bends and junctions. Based on the classification of the issues, a tensegrity mechanism that employs three tension springs and a passive universal joint was proposed as an initial solution for the bio-inspired robot studied in Chapter-2. This mechanism was analyzed in two methods. The initial method employed the theory of Tilt & Torsion. By classifying three case-study postures, the inverse pendulum configuration was chosen for further analysis. This was done using the CAD algorithm in Maple based on tilt limits criteria for the joint limits of the springs. Followed

by that, the stability analysis was carried out on the mechanism under a static mode. Since the exact architecture of the mechanism was unknown, the gravity forces and the free lengths of the spring were not considered for the analysis. It was found that the mechanism remained unstable under no-load conditions. By studying the mechanism stiffness, the design parameters of the mechanism were exploited which led to the modification of the value of parameter h from 1 to 0.5. This modification led to a stable configuration for the mechanism under no-load condition as well as under the presence of a preload of 2 N along each spring. However, the tilt limits for the mechanism were reduced from $\pm\pi/3$ to $\pm\pi/6$ radians.

The second method dealt with the singularity analysis of the mechanism with respect to the Euler angles of the universal joint. By using the SIROPA library in Maple, the parallel singularities were identified by setting initial joint limits for the 3-S$\underline{\text{P}}$S-U tensegrity mechanism. For extracting the feasible workspace and singularity free workspace of the mechanism, the triangular region was extracted by identifying tilt limits of the mechanism at one extremity of the workspace. By modification of joint limits, a singularity free workspace with a tilt range of $\pm\pi/6$ radians was obtained. By the addition of another tension spring to the mechanism, the 4-S$\underline{\text{P}}$S-U mechanism was exploited. Upon singularity analysis and modification of joint limit parameters, the 4-S$\underline{\text{P}}$S-U architecture generated higher tilt limits of $\pm 5\pi/18$ radians. The stability analysis was the reverified for both architectures for the Euler angles of the universal joint. Under fixed design parameters, both mechanisms remained unstable. The Hessian matrices and second-order derivatives of the tilt angles were studied to identify stable configurations through an optimization approach. By fixing r_f at 11 mm, optimum values of h and spring stiffness k were identified from the results of Genetic algorithm where the mechanism remains stable under the presence of a preload. A simple prototype of the 3-S$\underline{\text{P}}$S-U tensegrity mechanism was realized at LS2N with $h = 1$ and h= 0.6 to compare their stabilities. For the modified design parameter h, the singularity analysis was carried out again using the CAD algorithm for the 3-S$\underline{\text{P}}$S-U and 4-S$\underline{\text{P}}$S-U mechanisms. From the results of the algorithm, it was proved that the 4-S$\underline{\text{P}}$S-U architecture was suitable for addressing passive and active compliance issues. The narrow tilt range of 3-S$\underline{\text{P}}$S-U architecture restricts its application to address only the issue of active compliance.

CHAPTER 4

Experimental validation of the 3-SPS-U tensegrity mechanism

Problem statement:

1. *What are the possible ways for actuating a 3-SPS-U tensegrity mechanism?*
2. *What are the governing equations for achieving the the desired trajectory and the desired positions by the mechanism?*

4.1 Introduction

In this chapter, experimental validation is done on the 3-SPS-U tensegrity mechanism, which was analyzed in Chapter-3. This validation corresponds to how the active compliance issue could be addressed when the mechanism is integrated with the bio-inspired robot studied in Chapter-2. The actuation strategy of the mechanism is similar to that of a parallel manipulator. Parallel kinematics machines (PKM) have interesting applications in the industries over serial machines. PKM's offer better accuracy, lower mass/inertia properties and high structural stiffness [Mer06]. The performance of parallel robots varies within their workspace, which is considerably smaller when compared to serial robots. The inverse kinematics, in general, is easier to solve for parallel robots, especially when prismatic joints are employed. The 3-SPS-U architecture employs three actuated prismatic springs and the IKP appears simpler. However, it is complicated to determine the inverse of the Jacobians for parallel robots and the computation times appear to be higher.

For the tensegrity mechanism, the Jacobian matrix can be obtained from the direct kinematics matrix. The simpler architecture of the mechanism makes it easier to identify the cartesian velocities and accelerations. The most important step will be to plan a trajectory in such a way that the mechanism operates within the singularity free workspace. In the initial step, the workspaces are verified numerically for the prototype of a 3-S$\underline{P}$S-U tensegrity mechanism developed at LS2N using the CAD algorithm. This analysis provides the tilt and the joint limits for the mechanism within the singularity free workspace. By using these limits, experiments are conducted on the prototype using DC-motors. The DC-motors are equipped with position encoders which gives information about each angular displacement of the output shaft. Two algorithms were studied in Chapter-2 viz: Force control and Position control algorithms. The former is employed in the experiments to control the tilt limits of the mechanism. By providing the tilt angles as inputs, the position, velocity and acceleration of each prismatic joint are controlled. For attaining the desired position for a given tilt angle, a simplified Proportional-Integral-Derivative (PID) algorithm is employed. A BeagleBone(BB) black microcomputer is used to accomplish this closed-loop system. Two sets of experiments; a linear trajectory and a circular trajectory are carried out on the tensegrity mechanism for vertical and horizontal orientations. The theoretical curves are presented initially to identify the simulation time based on the maximum angular velocity of the motor. By performing experiments on the prototype, the actual position, the motor torques and the error between the measured and desired data are represented and studied.

4.2 Prototype and workspace of the tensegrity mechanism

A prototype of the 3-S$\underline{P}$S-U tensegrity mechanism is designed and developed at LS2N. A standard universal joint of 68 mm length, which was available at LS2N is employed for the tilting of the mechanism. In order to respect the value of $h = 0.6$ based on the results of stability analysis obtained in Chapter-3, a value of $r_f = 56.7$ mm is chosen. The base and end-effector platform are realized by rapid prototyping and they are made of Acrylonitrile Butadiene Styrene (ABS). Three slots, each on the end-effector and the base at a distance of r_f from the centre is provided to assemble L-shaped supports that hold the tension springs whose stiffness is 0.75 N/mm. The representation of the prototype and an exploded view of the digital model in CATIA is shown in Figure 4.1. The base of the prototype has threaded holes (Figure 4.1b), which are bolted to a rigid test platform for performing experiments. The prototype shown in Figure 4.1a is a scaled-up model from the actual design where the value of r_f is 11 mm. With the modification of r_f, the workspace for the 3-S$\underline{P}$S-U mechanism must be recalculated. At the home-pose where the tilt angles are zero, the distance between the base and the end-effector platform will be equal to the length of the universal joint (68 mm). Under working conditions, the solutions to the IKP can be solved using Eqs. (3.18a) to (3.18c) for a given tilt angle. The workspace of the mechanism is affected purely by the value of parameter h. The

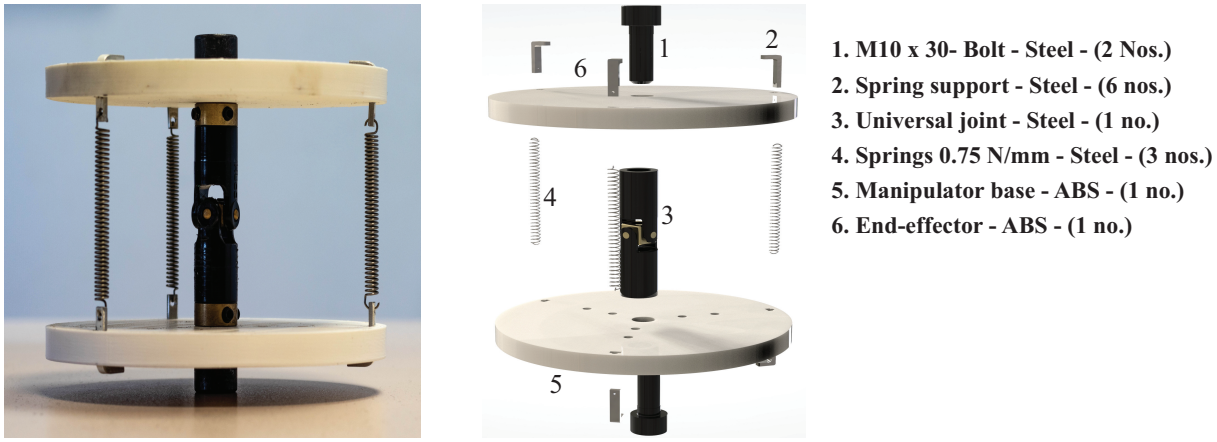


Figure 4.1 – Representation of the (a) prototype of 3-SPS-U tensegrity mechanism developed at LS2N and (b) exploded view of the digital model of the mechanism in CATIA

tilt limits for operation within the singularity free workspace for this configuration remains at $\pm 2\pi/15$ radians, based on the results obtained in Chapter-3. However, the joint limits for the mechanism are modified as the value of r_f is increased. From the analysis of CAD algorithm, two of the prismatic springs reach a minimum length of 40 mm at $[\eta, \phi] = [0\ ,\ -\pi/3]$ radians, while the third spring reaches a length of 106 mm. The remaining edges of the triangle are estimated for these joint limits using Eqs. (3.29) to (3.32). The singularity free workspace in the form of a square bounded by $\pm 2\pi/15$ radians is also generated and superimposed on the workspace of the mechanism obtained by CAD algorithm, which is represented below in Figure 4.2.

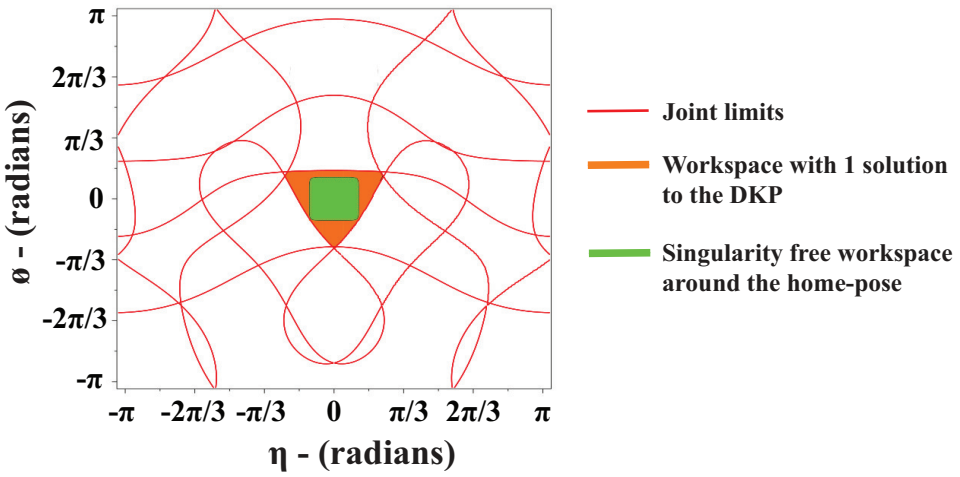


Figure 4.2 – Workspace of the 3-SPS-U tensegrity mechanism obtained by CAD algorithm for $r_f = 56.7$ mm and $h = 0.6$ at $[l_{min}, l_{max}] = [40\ ,\ 106]$ mm

The orange regions correspond to the workspace where there exists one solution to the DKP.

A total of 104 cells represent this region whereas the white zones represent the singularity regions with a total of 3460 cells. Linear and circular trajectories within this singularity free workspace are chosen for performing experiments on the prototype shown in Figure 4.1a.

4.3 Experimental setup

The control of the 3-SPS-U tensegrity mechanism is carried out using a BB black [**?**] micro-computer, which was employed for the bio-inspired robot studied in Chapter-2. In order to have a higher torque, a static model is validated by the actuation of the springs through a DC-motor coupled with a planetary gearhead. In general, a DC motor speed control can be achieved through variation of voltage but the position control of the motor shaft is difficult to achieve [MLN18]. The DC motors are integrated with quadrature encoders which provides information about the motor pulse counts for every revolution of the output shaft of the gearbox. The direction control of DC motors is achieved using microcontrollers that use an H-bridge module, which is controlled through PWM duty cycles from BB black. In order to attain the desired prismatic length and tilt the tensegrity mechanism, a simplified Proportional-Integral-Derivative (PID) algorithm is employed to have a closed-loop system. Through a PID, it is possible to attain the desired spring length by calculating the error between the measured and desired set points of the algorithm and minimizing it.

4.3.1 Interfacing between motors and controllers

The actuation of springs is carried out using three Maxon DC-motors (Series-352994) [Max]. Each DC-motor has a Maxon 36/2 servo controller for position control. The DC-motors are coupled with a planetary gearhead (Series-144041) which has a reduction ratio of 1621:1. Thus, higher output torque is obtained at the gearbox output shaft. The key specifications of the motor and the gearbox are provided below in Tables 4.1 and 4.2.

Parameters	**Value**
Nominal voltage	24 V
Nominal current (I_c)	0.177 A
Nominal speed	3720 rpm
Nominal torque	6.05 mNm
Stall torque	14.2 mNm
Torque constant (k_t)	34.8 mNm/A
Speed constant (k_e)	274 rpm/V
Armature resistance (R)	59 Ω

Table 4.1 – Specifications of the Maxon DC-motor (Part No. 352994 ϕ22) [Max]

Each DC motor is integrated with a 3-channel optical quadrature encoder (HEDL 5540)

Parameters	Value
No. of stage in reduction	5
Reduction ratio (G)	1621:1
Max. dynamic axial load	100 N
Output gearshaft diameter	3.2 mm
Efficiency (η_s)	42 %

Table 4.2 – Specifications of the Maxon Planetary gearhead (Part No. 144041) [Max]

offered by Maxon [Max]. The encoder pins consist of an input supply of 5 V, channels A–B and an index channel. The channels A and B are employed in the experiments to sense the direction of rotation of motor for every angular displacement. Both these channels have a phase difference of 90°. When the motor rotates in a clockwise direction, channel A leads channel B and vice-versa when the direction of rotation is counterclockwise [Kim13]. The encoder can read 500 counts for every change in the angular position of the DC-motor. For controlling the DC motor direction and input power supply, the PWM and GPIO pins of BB black are employed. The maximum voltage supported by these pins is 3.3 V. A 5 V output pin from the BB black (Pin P9.8) is used as a power source for the quadrature encoder of each motor through a parallel circuit connection. The channel A and channel B information from each encoder are passed as inputs for three eQEP pairs of BB black. As the output voltage of the quadrature encoder is 5 V and the BB black eQEP pins support a maximum tension of 3.3 V, a 1 kΩ potential divider bridge is employed to step down the voltage. The interfacing between BB Black to one of the DC-motors with its associated servo controller and quadrature encoder is represented in Figure 4.3. As seen in Figure 4.3, each DC-motor is connected to its servo controller through a **J2A** connector. The **J5** connector is used for the PWM and GPIO ports. Using the **J7** cable, the BB black is connected to the computer. The list of pins from BB black that are connected with the ESCON servo-controllers and the encoders of each motor is provided below in Table 4.3.

Motor	**PWM (Duty cycle)**	**GPIO (Enable)**	**eQEP-A (Ch-A)**	**eQEP-B (Ch-B)**
Motor-1	P9.14 (DigIN1)	P9.13 (DigIN2)	P9.92	P9.27
Motor-2	P8.13 (DigIN1)	P8.15 (DigIN2)	P8.35	P8.33
Motor-3	P9.31 (DigIN1)	P9.25 (DigIN2)	P8.12	P8.11

Table 4.3 – BB black pin address for each motor with its associated servo controller and encoder

The three DC-motors are assembled on a platform, which is made by rapid prototyping at LS2N. As the distance between the gearbox output shaft and the tensegrity mechanism is higher, an extended shaft with a flange coupling is employed to connect to the gearbox shaft. A 40 mm diameter pulley is assembled on the extended shaft and a cable of 1.5 mm diameter is wound over this pulley.

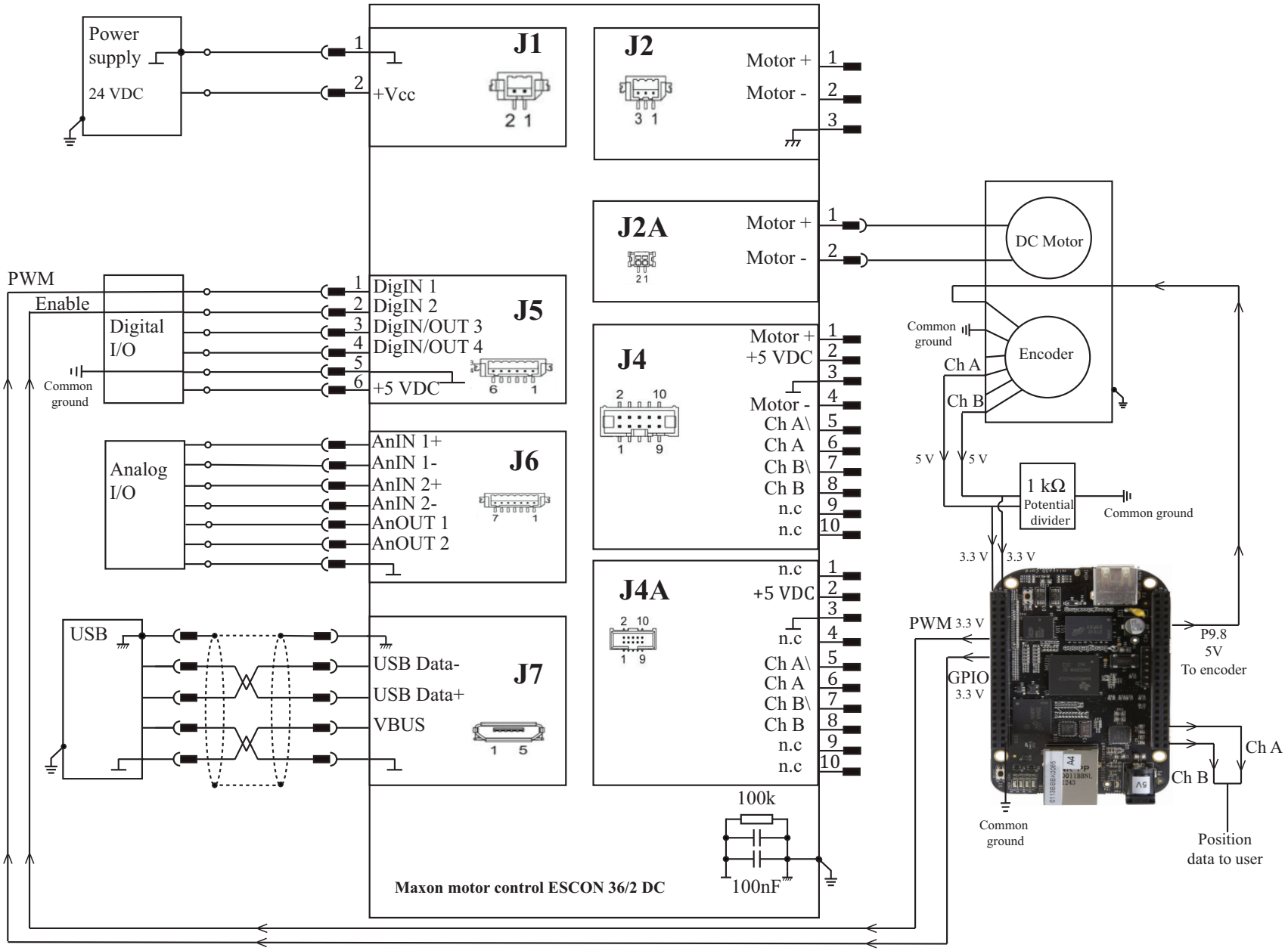


Figure 4.3 – Layout of the interface between BB black to one DC-motor with its associated servo controller and encoder

The other end of the cable connects to the end-effector mounting support by passing through each spring. For eliminating the effects of radial play in the pulley shaft, flange couplings are employed to minimize these effects. The cables are assembled in such a way that at the home-pose, the tension along the cables is high. The experiments are carried out for two orientations; vertical and horizontal under no-load conditions. The entire experimental setup and it's digital model are represented below in Figures 4.4a to 4.4d.

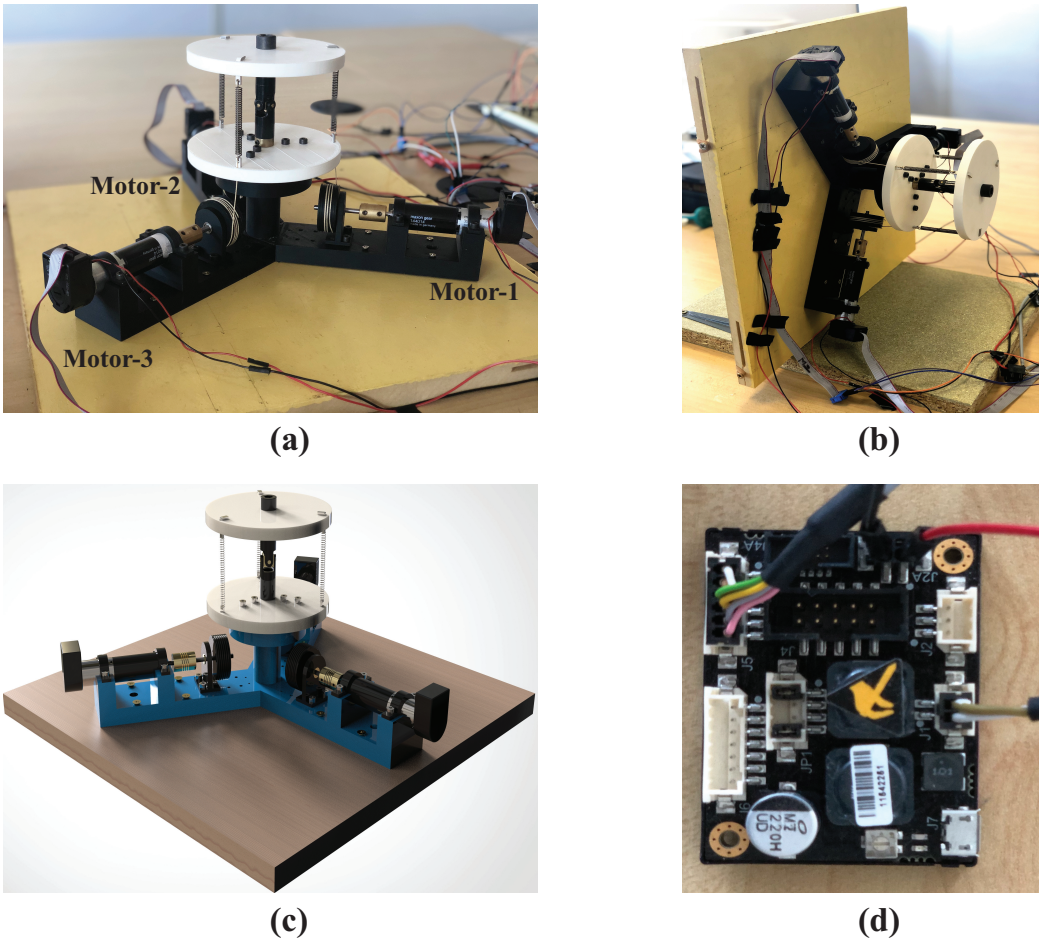


Figure 4.4 – Representation of the (a) Experimental setup of the tensegrity mechanism in vertical orientation , (b) Experimental setup of the tensegrity mechanism in horizontal orientation , (c) Digital model of the experimental setup and (d) one of the three ESCON 36/2 servo-controllers

4.4 Trajectory planning and control methodology

In the experimental validation, the IKP is solved. This means that a set of tilt angles η and ϕ are passed as inputs to the control loop, for which the IKP is solved through actuation of the motors connected to the springs. The pose variables and the articular variables for the mechanism are given by $\mathbf{q} = [\eta, \phi]$ and $\rho = [l_1, l_2, l_3]$ [VC20]. The closed-loop equations which

provide the solutions to the IKP were studied in Chapter-3 as well as in [VC20] with respect to the Euler angles of the universal joint. The simpler form of the IKP from Figure 3.4a is given by:

$$(b_{ix} - c_{ix})^2 + (b_{iy} - c_{iy})^2 + (b_{iz} - c_{iz})^2 = l_i^2 \text{ , with } i = 1, 2, 3 \tag{4.1}$$

The length of each spring for a given tilt angle can be estimated using Eq. (4.1), which is a simpler form of Eqs. (3.18a) to (3.18c). The IKP for each spring is converted into angular displacements of the pulley, which will be the target position for each DC-motor. The equation for the desired angular position of the pulley with respect to the IKP is given by:

$$\theta_{di} = \frac{(l_i - l_{home})}{r} \text{ , with } i = 1, 2, 3 \tag{4.2}$$

In Eq. (4.2), θ_{di} is the desired angular position calculated from IKP for given input tilt angles η and ϕ. The parameter r indicates the pulley radius, which is 20 mm. At the home position, the value of l_{home} is 68 mm for r_f = 56.7 mm and h= 0.6. For every angular displacement of the DC-motor, encoder data is passed to the BB black ports. The angular displacement of the pulley from the encoder channel information can be calculated by:

$$\theta_{mi} = \frac{E_i \pi}{2CG} \text{ , with } i = 1, 2, 3 \tag{4.3}$$

In Eq. (4.3), E_i indicates the output data of the encoder channels of Motor-i. The angle θ_{mi} is the measured angular displacement at the output shaft of the gearbox. C indicates the counts per revolution of the encoder, which is 500 and G is the reduction ratio of the gearbox, which is taken from Table 4.2. At a given tilt angle, the mechanism tries to attain the position θ_{di}. In order to minimize the errors between the desired and measured angular positions, the PID control algorithm is employed.

4.4.1 Trajectory generation

The trajectory planning approach proposed in [KD04], which was studied in Chapter-2 for the dynamic force model is employed for the control of the tensegrity mechanism. For every angular displacement of the DC-Motor, it is necessary to determine the joint velocities and accelerations for the mechanism. During the change of tilt angles from home-pose to the desired position, the cartesian velocities of the prismatic joints can be calculated using the equation [CCLW15]:

$$\begin{bmatrix} \dot{l_1} \\ \dot{l_2} \\ \dot{l_3} \end{bmatrix} = \mathbf{J}_c \begin{bmatrix} \dot{\eta} \\ \dot{\phi} \end{bmatrix} \text{ , with } \mathbf{J}_c = \begin{bmatrix} \frac{\partial l_1}{\partial \eta} & \frac{\partial l_1}{\partial \phi} \\ \frac{\partial l_2}{\partial \eta} & \frac{\partial l_2}{\partial \phi} \\ \frac{\partial l_3}{\partial \eta} & \frac{\partial l_3}{\partial \phi} \end{bmatrix} \tag{4.4}$$

In Eq. (4.4), $\mathbf{J}_c$ represents the direct kinematics matrix ($\mathbf{A}$) of the 3-SPS-U tensegrity mechanism. Using Eq. (4.4), the joint velocities are estimated, followed by which the desired angular velocities of the output shaft can be estimated. For given input tilt angles, the trajectory generation is carried out using a fifth-degree polynomial equation [KD04]. This equation is used to define the position $s(t)$, velocity $\dot{s}(t)$ and acceleration $\ddot{s}(t)$ for the mechanism and they are given by:

$$s(t) = 10\left(\frac{t}{t_f}\right)^3 - 15\left(\frac{t}{t_f}\right)^4 + 6\left(\frac{t}{t_f}\right)^5 \tag{4.5}$$

$$\dot{s}(t) = 30\left(\frac{t^2}{t_f^3}\right) - 60\left(\frac{t^3}{t_f^4}\right) + 30\left(\frac{t^4}{t_f^5}\right) \tag{4.6}$$

$$\ddot{s}(t) = 60\left(\frac{t}{t_f^3}\right) - 180\left(\frac{t^2}{t_f^4}\right) + 120\left(\frac{t^3}{t_f^5}\right) \tag{4.7}$$

For the experiments, a linear trajectory and a circular trajectory are performed on the tensegrity mechanism for both orientations. The position, velocity and acceleration equations, as well as the maximum simulation time, varies according to the type of trajectory.

Linear trajectory

In the case of linear trajectory, the minimum travelling time t_f is determined when the velocities and accelerations reach the saturation point in the trajectory. For the fifth degree polynomial, the value of t_f for a linear trajectory is given by [KD04]:

$$t_f = \max\left(\frac{15|D|}{8k_v}, \sqrt{\frac{10|D|}{\sqrt{3}k_a}}\right) \tag{4.8}$$

In Eq. (4.8), $|D|$ represents the norm of the position, k_v represents the maximum velocity and k_a represents the maximum acceleration. At home-position, let $\mathbf{D}_1$ be the vector that contains the initial tilt angles η_i and ϕ_i. The final tilt positions are given by the vector $\mathbf{D}_2$. The intermediate pose $\mathbf{D}$ is given by:

$$\mathbf{D}(t) = \mathbf{D}_1 + (\mathbf{D}_2 - \mathbf{D}_1)s(t) \tag{4.9}$$

The corresponding velocity and acceleration vectors can be found using the first and second order derivative of Eq. (4.9) with respect to time. The equations are given by [CCLW15]:

$$\mathbf{V}(t) = (\mathbf{D}_2 - \mathbf{D}_1)\dot{s}(t) \tag{4.10}$$

$$\mathbf{A}(t) = (\mathbf{D}_2 - \mathbf{D}_1)\ddot{s}(t) \tag{4.11}$$

The joint coordinate vector can be obtained by the equation:

$$\mathbf{q}(t) = \mathbf{f}\left(\mathbf{D}(t)\right) \tag{4.12}$$

In Eq. (4.12), $\mathbf{f} = [l_1, l_2, l_3]^T$ is the vector that contains the solution of the IKP for given tilt angles. The joint velocities can be computed with the help of the Jacobian matrix $\mathbf{J}_c$ using the equation:

$$\dot{\mathbf{q}}(t) = \mathbf{J}_c\mathbf{V}(t) \tag{4.13}$$

The accelerations in the joints can be computed using the Jacobian matrix and it's time derivative. The equation is given by:

$$\ddot{\mathbf{q}}(t) = \mathbf{J}_c\mathbf{A}(t) + \dot{\mathbf{J}}_c\mathbf{V}(t) \tag{4.14}$$

The desired position, velocity and acceleration vectors of the joints from Eqs. (4.12) to (4.14) will be used as inputs for the PID algorithm to perform a linear trajectory.

Circular trajectory

Compared to the linear trajectory, the travelling time required by the mechanism is higher as the circular path goes from 0 to 2π radians. The value of t_f is given by:

$$t_f = \max\left(\gamma_V, \sqrt{\gamma_A}\right) \tag{4.15}$$

In Eq. (4.15), γ_V is the ratio of maximum joint velocity to speed of motor and γ_A is the ratio of maximum joint acceleration to the acceleration of motor [CCLW15]. Before the creation of circular trajectory, an initial linear trajectory is performed where one of the three springs is tilted to a certain angle. The values of the input tilt angles to perform a circular trajectory is given by:

$$\mathbf{D}(t) = \begin{bmatrix} \eta \\ \phi \end{bmatrix} = \begin{bmatrix} R\sin(r_0 + (r_1 - r_0)s(t)) \\ R\cos(r_0 + (r_1 - r_0)s(t)) \end{bmatrix} \tag{4.16}$$

In Eq. (4.16), R corresponds to the radius of the circular trajectory which is equal to the tilt angle chosen within the singularity free workspace. The values r_0 and r_1 maps the initial and final points of circle and it goes from 0 to -2π to perform a counterclockwise circular trajectory. The corresponding angular velocities and accelerations are given by:

$$\mathbf{V}(t) = \begin{bmatrix} R(r_1 - r_0)\dot{s}(t)\cos(r_0 + (r_1 - r_0)s(t)) \\ -R(r_1 - r_0)\dot{s}(t)\sin(r_0 + (r_1 - r_0)s(t)) \end{bmatrix} \tag{4.17}$$

$$\mathbf{A}(t) = \begin{bmatrix} R(r_1 - r_0)\ddot{s}(t)\cos(r_0 + (r_1 - r_0)s(t)) - R(r_1 - r_0)^2\dot{s}(t)^2\sin(r_0 + (r_1 - r_0)s(t)) \\ -R(r_1 - r_0)\ddot{s}(t)\sin(r_0 + (r_1 - r_0)s(t)) - R(r_1 - r_0)^2\dot{s}(t)^2\cos(r_0 + (r_1 - r_0)s(t)) \end{bmatrix} \tag{4.18}$$

The IKP, the joint velocities and accelerations can then be computed by substituting Eqs. (4.16) to (4.18) in Eqs. (4.12) to (4.14). At the end of the circular trajectory, once again a linear trajectory is performed to bring the mechanism back to the home position.

4.4.2 Force control algorithm

In Chapter-2, two types of algorithm were classified namely: Force control and position control. The former was employed for the bio-inspired robot in order to accomplish static phases through motor torque control by setting threshold limits for moving inside unknown environments. Similarly, for the control of the tensegrity mechanism, the force control algorithm is employed to reach the desired positions for input tilt angles. By applying motor torques and current, the mechanism tilts to attain the desired position. Once the desired position is reached, the current and torque are cut off for attaining a static phase. For each angular position of DC-Motor, encoder position data is transmitted to the user by the BB black. However, the output data from encoders are not directly used to calculate the solutions to the IKP as there exists a significant difference between the desired and measured positions. These differences are caused by factors such as motor inertia, frictional effects and inertial forces. In order to compensate these factors, a closed loop feedback system is employed which tries to minimize the errors between the desired and measured positions. The classical relation for the PID control scheme which provides the motor torque is given by the equation [CCLW15, JCRM16, Lem19]:

$$\Gamma = J(\ddot{\theta}_i + K_P(\theta_{di} - \theta_{mi}) + K_D(\dot{\theta}_{di} - \dot{\theta}_{mi}) + K_I \int_0^t (\theta_{di} - \theta_{mi})) \text{ , with } i = 1, 2, 3 \tag{4.19}$$

In Eq. (4.19), Γ is the regulated torque obtained at the gearbox output shaft after PID correction. For computing the torques induced on each motor, Eq. (4.19) is multiplied by the gear reduction ratio G from Table 4.2. The inertia of motor-gearbox assembly J is taken as $4.1e-7$ kg.m^2 [Max]. In Eq. (4.19), K_P is the proportionality term which examines the error term and responds proportionally. The integral term K_I tries to reduce the error at each time instant. The overshoot of value from the desired position is compensated by the difference term K_D. Thus, the PID in general takes into account the past step, the present step and the future step such that the error is reduced [MLN18, O'D09]. The values for the PID terms of Eq. (4.19) are given by:

$$\begin{aligned} &\omega = \frac{k_t k_e}{RJ} = 14 \ rad/s \\ &K_P = 3\omega^2 = 588 \ , \ K_D = 3\omega = 42 \ , \ K_I = \omega^3 = 2744 \end{aligned} \tag{4.20}$$

In Eq. (4.20), ω is a parameter which is determined from the torque constant (k_t), speed constant (k_e), resistance (R) and inertial parameters (J) of the DC-Motor [CCLW15, Lem19]. The closed loop system which is incorporated for the control of the 3-SPS-U tensegrity mechanism is represented in Figure 4.5. In Figure 4.5, e and $\dot{e}$ indicates the error and its derivative which

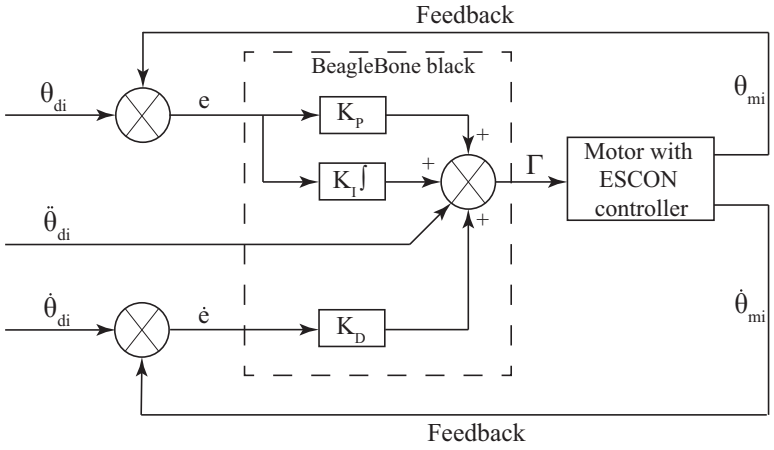


Figure 4.5 – Closed loop feedback control scheme employed for the tensegrity mechanism [CCLW15, Lem19]

are computed based on the difference between desired and measured data. Before the start of experiments, the ESCON servo controllers are calibrated in a computer. The saturation current is used as reference for controlling the direction of rotation of the DC-Motors. For safer operations, 90% and 10% duty cycles are set as limits for the rotation of motors on either direction. The PWM calibration of each ESCON servo-controller is done as per Table 4.4.

PWM Duty	**Nominal current** (A)	**Motor torque** (mN.m)	**Motor speed** (rpm)
10%	-0.177	-6.05	3720 (Counter-clockwise)
50% (idle)	0	0	0
90%	0.177	6.05	3720 (Clockwise)

Table 4.4 – ESCON controller calibration for the force control algorithm

The force control algorithm is executed in C language using the BB black. A simple form of the control loop that is employed to perform experiments on the 3-SPS-U tensegrity mechanism for linear and circular input trajectories is provided in Algorithm 2.

4.5 Results of experiments

As mentioned in the previous sections, the experiments are carried out for a linear trajectory as well as a circular trajectory. Both these trajectories will be executed for two orientations of the mechanism namely: Vertical and Horizontal. The vertical orientation is the natural pose of the mechanism as depicted in Figure 4.1a. This orientation is also inline with the vertical orientation of the bio-inspired robot studied in Chapter-2. The horizontal orientation is the posture when the tensegrity mechanism is coupled with the bio-inspired robot and the assembly is moving inside a horizontal pipeline. The experiments are carried out in BB black which

Algorithm 2 Force control algorithm using PID for a given input trajectory

Initialize: Measured & desired positions (θ_{mi},θ_{di}) , velocities ($\dot{\theta}_{mi}$,$\dot{\theta}_{di}$) and accelerations ($\ddot{\theta}_{mi}$,$\ddot{\theta}_{di}$)
Set: Home and target tilt limits: $[\eta_0 \rightarrow \eta_f]$, $[\phi_0 \rightarrow \phi_f]$
Initialize: Error (e_i), Sum (a_i) and Difference terms ($\dot{e}_i$)
Initialize and set PWM, GPIO and eQEP to idle states
Initialize time t and set finish time t_f
Set status = 1 ▷ 1 indicates motor status "RUN"
while (status == 1) **do**
    for t = 1 – t_f **do**
        for i = 1 – 3 **do** ▷ i indicates the Motor number
            Compute $s(t)$, $\dot{s}(t)$ and $\ddot{s}(t)$ for polynomial interpolation
            Compute $[\eta(t), \dot{\eta}(t), \ddot{\eta}(t)]$ and $[\phi(t), \dot{\phi}(t), \ddot{\phi}(t)]$
            Calculate $\mathbf{J}_c$ and $\dot{\mathbf{J}}_c$, Compute $\sum_{k=1}^{3} [l_k, \dot{l}_k, \ddot{l}_k]$ and $\sum_{k=1}^{3} [\theta_{kdi}, \dot{\theta}_{kdi}, \ddot{\theta}_{kdi}]$
            Obtain measured positions from encoder: $\theta_{mi}(t)$
            Calculate error, $e[i](t) = \theta_{di}(t)$ - $\theta_{mi}(t)$
            Calculate Sum, $a[i](t) = a[i](t-1) + e[i](t)$ ▷ $a[i](t-1)$ for first iteration is 0
            Calculate Diff, $\dot{e}[i](t) = e[i](t)/t$
            Compute torque ($\Gamma[i](t)$) and then current ($I[i](t)$) using torque constant
            if $I[i](t) > 0.177$ **then** ▷ 0.177 A is the nominal current
                Rotate motors clockwise by applying 90% duty cycle
            else if $I[i](t) <$ -0.177 **then**
                Rotate motors anti-clockwise by applying 10% duty cycle
            end if
        end for
    end for
    set status = 0 ▷ 0 indicates motor status "STOP"
end while
Set PWM, GPIO and eQEP to idle state
return Desired and measured data values to the user

operates on Debian 9.5. The link for the complete algorithm written in C for the control of the tensegrity mechanism as well as pin activation sequence for BB black is provided at the bottom of this page [1].

4.5.1 Linear trajectory

For the linear trajectory, two kinds of experiments are performed. In the first experiment, which is called the One spring pull, one of the three springs pulls the end-effector platform. In the second experiment, which is called the Two springs pull, two of the three springs pulls the end-effector platform. The experiments are carried out at a frequency of 1000 Hz. Due to the memory issues associated with the BB black, both experiments could not be carried out on a

1. BB black pin activation and control of the tensegrity mechanism written in C : GitHub link.

single stretch. Firstly, the theoretical trajectories are created in MATLAB for both experiments which provide the simulation time with respect to the maximum angular velocity of the motor after gear reduction. Followed by that the results of both experiments in vertical and horizontal orientations of the mechanism are presented.

Theoretical curves

Before performing the experiments, it is necessary to identify the simulation time required to attain the desired tilt position. The maximum angular velocity of the DC-Motor from Table 4.1 is around 0.25 rad/s after gear reduction. From the singularity analysis curves, at $2\pi/15$ radians, the spring attached to Motor-1 pulls the end-effector platform while at $-2\pi/15$ radians, Motor-2 and Motor-3 pulls the end-effector platform. Based on the results of IKP at these tilt angles, Eqs. (3.29) to (3.32) are employed to determine the other sets of input tilt angles which are found to be $[\eta, \phi] = [\pm 7\pi/60\ ,\ \pm\pi/15]$ radians. The two experiments are then carried out in the following sequence:

- **One spring pull** : Home → Motor-3 Tilt → Home → Motor-2 Tilt → Home → Motor-1 Tilt → Home
- **Two springs pull** : Home → Motors-1,3 Tilt → Home → Motors-2,3 Tilt → Home → Motors-2,1 Tilt → Home

The input tilt angles for performing both experiments are chosen within the singularity free workspace square of the 3-SPS-U tensegrity mechanism. The trace of trajectory of input tilt angles chosen within the workspace of the mechanism is represented below in Figure 4.6.

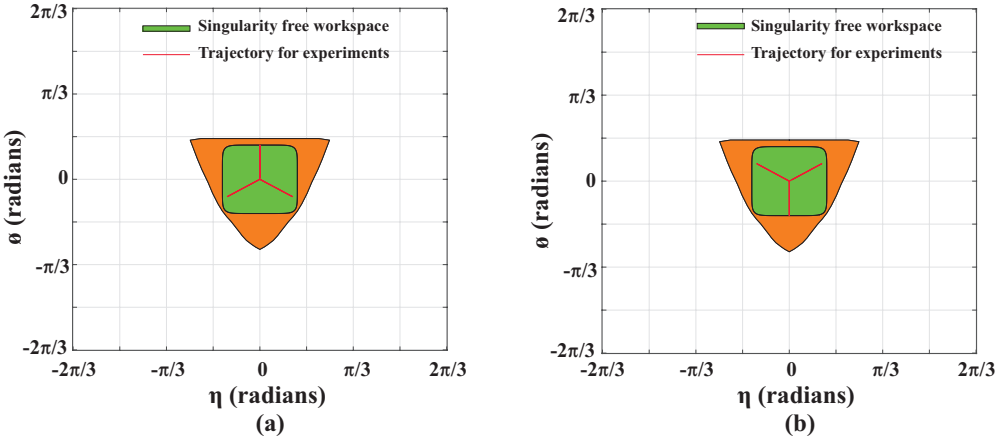


Figure 4.6 – Trace of the desired input trajectories chosen for the (a) one spring pull and (b) two springs pull experiments within the workspace of the tensegrity mechanism

The desired anglular positions of the pulley for each steps of both experiments are calculated with the help of IKP and the pulley radius. The corresponding values are provided in Table 4.5. From Table 4.5, the maximum angular position for the one spring pull is around ± 1.25 radians and for the two springs pull it is around ± 1.1 radians. Using Eq. (4.8), for a maximum angular

Type	**Input tilt angles** (rad)		l_1 (mm)	θ_{d1} (rad)	l_2 (mm)	θ_{d2} (rad)	l_3 (mm)	θ_{d3} (rad)
	η	ϕ						
One spring pull	0	0	68	0	68	0	68	0
	$7\pi/60$	$-\pi/15$	78	0.5	78	0.5	43	-1.25
	0	0	68	0	68	0	68	0
	$-7\pi/60$	$-\pi/15$	78	0.5	43	-1.25	78	0.5
	0	0	68	0	68	0	68	0
	0	$2\pi/15$	43	-1.25	78	0.5	78	0.5
	0	0	68	0	68	0	68	0
Two springs pull	0	0	68	0	68	0	68	0
	$7\pi/60$	$\pi/15$	54.8	-0.66	90	1.1	54.8	-0.66
	0	0	68	0	68	0	68	0
	0	$-2\pi/15$	90	1.1	54.8	-0.66	54.8	-0.66
	0	0	68	0	68	0	68	0
	$-7\pi/60$	$\pi/15$	54.8	-0.66	54.8	-0.66	90	-1.1
	0	0	68	0	68	0	68	0

Table 4.5 – Desired angular positions of the pulley for one spring and two springs pull experiments

velocity of 0.25 rad/s motor after gear reduction and the norm being the maximum angular position of the experiment, the simulation time to perform one cycle of the experiment is found to be 9 s for both experiments. Each experiment comprises of six steps and at the end, the mechanism is returned to the home position. Thus, the total theoretical simulation time to perform the six steps is around 54 s. Using the fifth-degree polynomial equations and the Jacobian matrix, the theoretical traces of the angular positions (θ_{di}), velocities (μ_{di}) and accelerations (ν_{di}) of each pulley are generated for both experiments and are represented in Figures 4.7 and 4.8. The maximum angular acceleration for the experiments is around 0.0832 rad/s^2. From the traces it is found that the maximum angular velocity of 0.25 rad/s is reached when the pulley travels between home position to $\pm$1.25 radians for one spring pull and $\pm$1.1 radians for two springs pull experiments. Similarly, the position, velocity and acceleration profiles are generated for the input tilt angles η and ϕ. For the tilt angles, the maximum velocities and accelerations are given by $[\dot{\eta}, \dot{\phi}]$ = [0.0758 , 0.0873] rad/s and $[\ddot{\eta}, \ddot{\phi}]$ = [0.0259 , 0.03] rad/s^2. The plots of the curves corresponding to the input tilt angles for the one spring pull experiment are demonstrated in Figure 4.9. The plots of the tilt angle positions, velocities and accelerations for the two springs pull experiment are provided in Appendix-B.

Results of one spring pull experiment

The one spring pull experiment is performed on the 3-SPS-U tensegrity mechanism for vertical and horizontal orientations. The simulation time for each cycle of the experiment is set at 9 s so that the maximum velocity is reached. The control algorithm consists of sleep routines between each cycle for smoother operations. Thus, the total time taken to perform the one spring pull experiments is around 128 s. At each instance of the experiment, the IKP is calculated for the input tilt angles. The minimum length reached by one of the actuated prismatic spring is 43 mm while the other two springs reaches a maximum length of 78 mm.

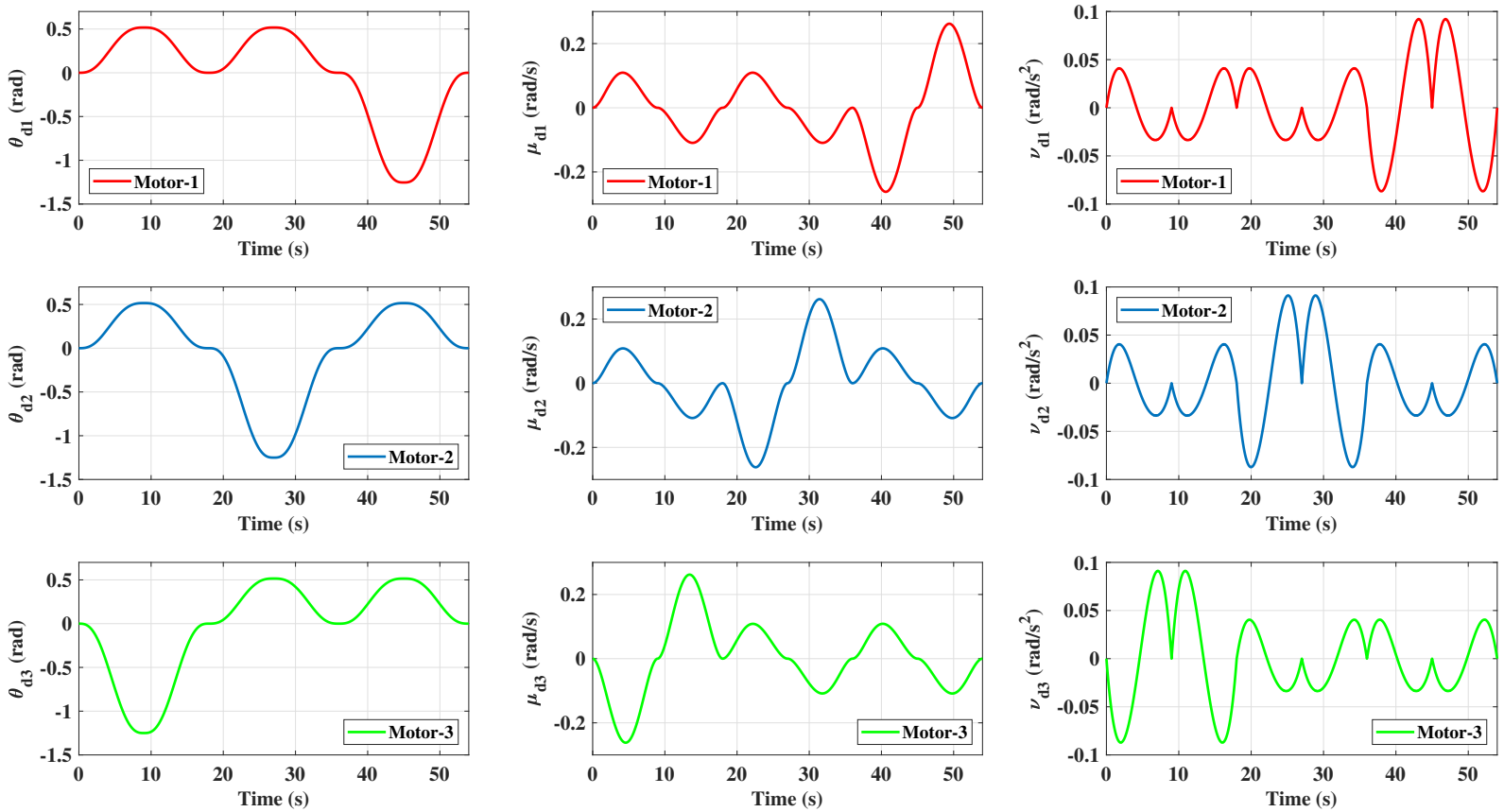


Figure 4.7 – Theoretical angular positons (θ_d), velocities (μ_d) and accelerations (ν_d) of the pulleys of Motor-1 (red) , Motor-2 (blue) and Motor-3 (green) for one spring pull experiment

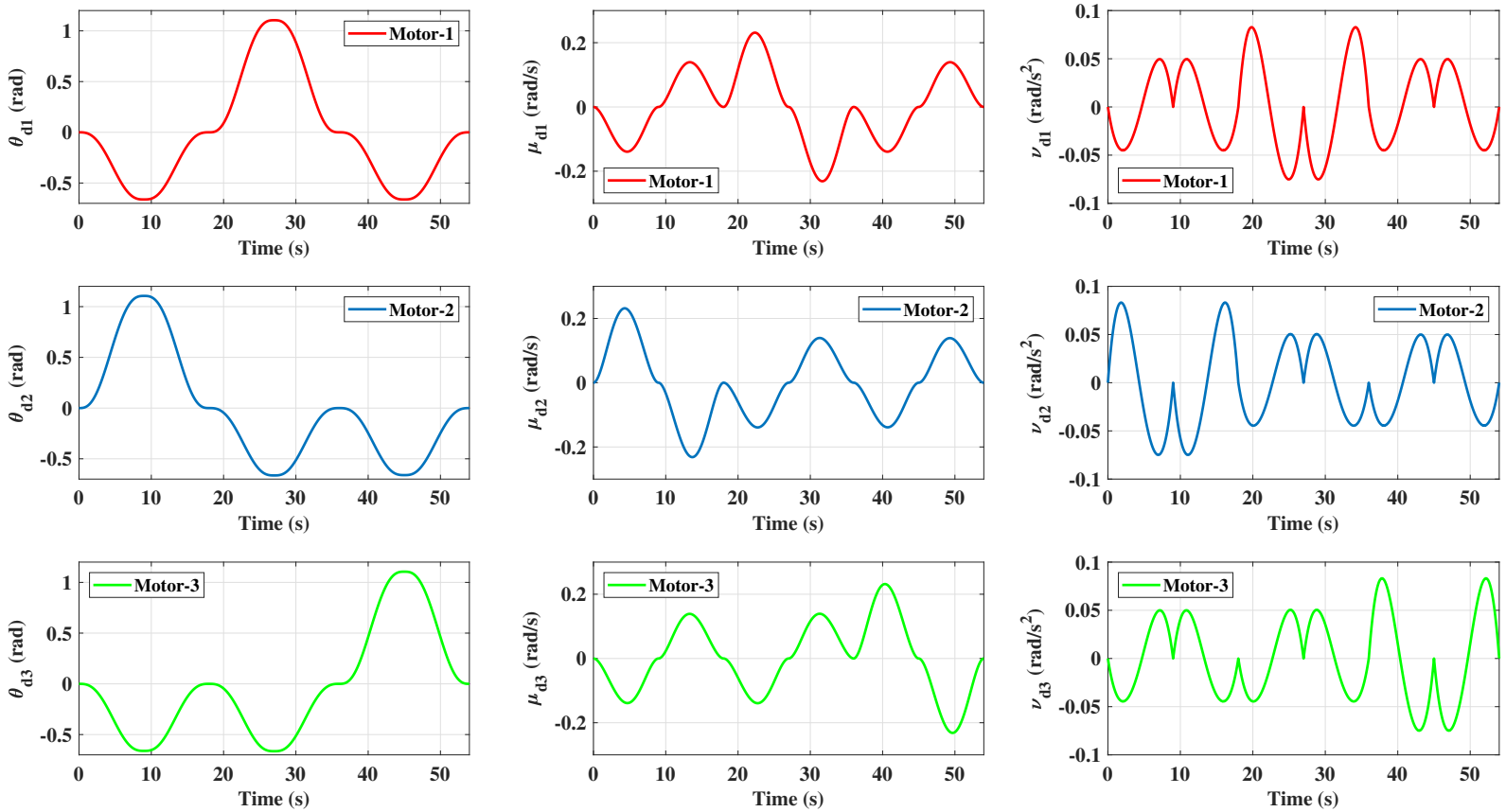


Figure 4.8 – Theoretical angular positons (θ_d), velocities (μ_d) and accelerations (ν_d) of the pulleys of Motor-1 (red) , Motor-2 (blue) and Motor-3 (green) for two springs pull experiment

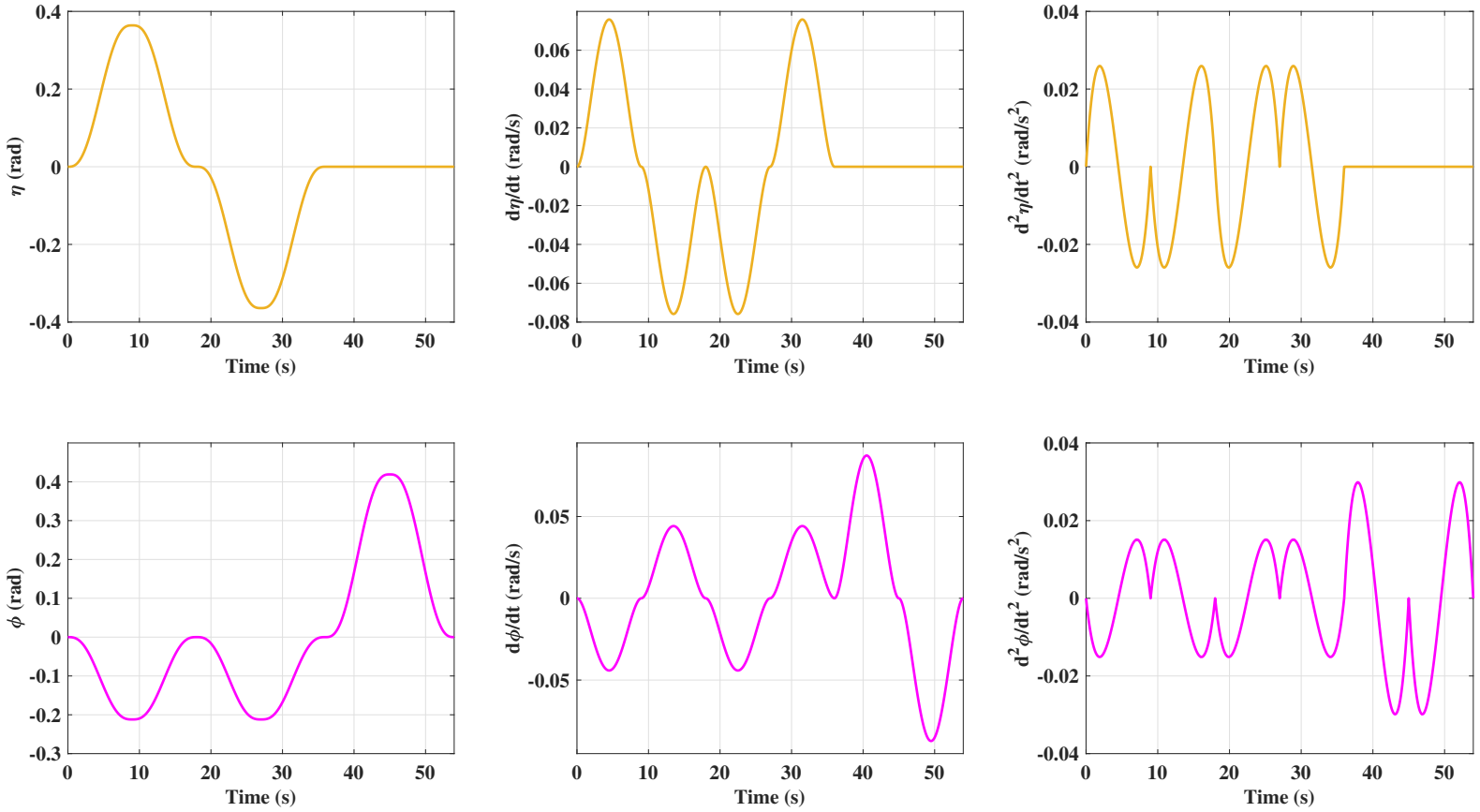


Figure 4.9 – Theoretical positions, velocities and accelerations of the input tilt angles η (orange) and ϕ (magenta) for the one spring pull experiment

The plot of the joint positions (IKP) of each actuated prismatic joint from the experiment is represented below in Figure 4.10 for both orientations of the mechanism.

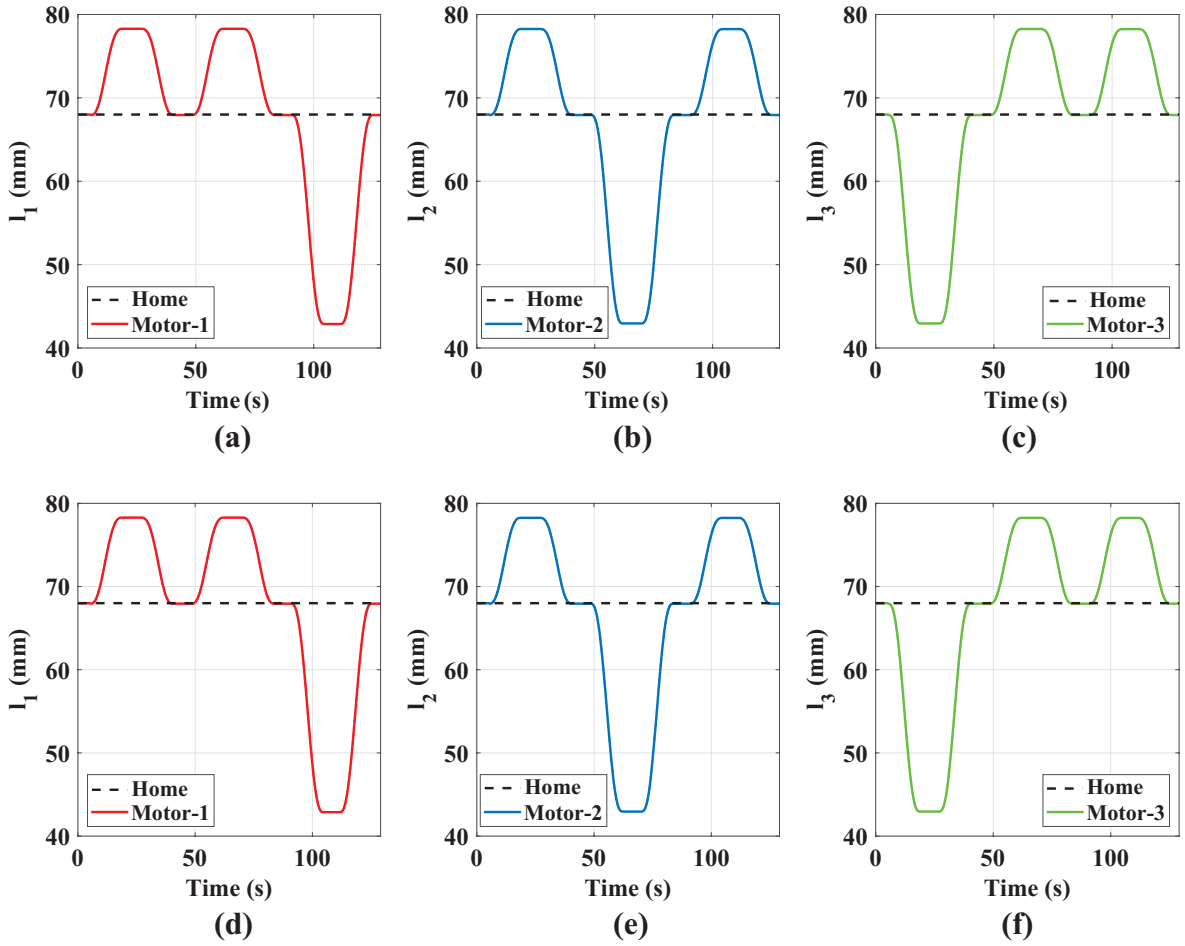


Figure 4.10 – Position of prismatic springs along the linear trajectory for one spring pull experiment in the (a),(b),(c): vertical and (d),(e),(f): horizontal orientations of the mechanism

In Figure 4.10, the dotted black line in each plot represents the home position of the mechanism at 68 mm. The values of the actual position of the three prismatic springs for the entire experimental cycle are provided in Table 4.6. From Figures 4.10a to 4.10f and from Table 4.6, a good response to the PID control could be observed for the joint positions. The position data for both orientations almost remain the same and while returning to the home-pose, a uniform distance of 67.94 mm on each prismatic length could be observed at some instances for both orientations of the mechanism. From the obtained results, the error can be calculated between the measured and desired angular positions of the pulley using the equation:

$$Error = \frac{(\theta_{mi} - \theta_{di})180}{\pi} \tag{4.21}$$

Experiment type	**Orientation**	**Operation type**	**Input tilt angles** (rad)		**IKP**, l_1 (mm)		**IKP**, l_2 (mm)		**IKP**, l_3 (mm)		**Global error** (degrees)	
			η	ϕ	**Desired**	**Actual**	**Desired**	**Actual**	**Desired**	**Actual**	**Pull**	**Push**
One spring pull	Vertical	Initial	0	0	68	68	68	68	68	68	-0.18 to -0.46	0.19 to 0.42
	Horizontal				68	68	68	68	68	68		
	Vertical	Pull:Motor-3	$7\pi/60$	$-\pi/15$	78	78.33	78	78.26	43	42.78		
	Horizontal				78	78.28	78	78.26	43	42.94		
	Vertical	Home-pose	0	0	68	67.94	68	67.94	68	67.94		
	Horizontal				68	67.94	68	67.94	68	67.94		
	Vertical	Pull:Motor-2	$-7\pi/60$	$-\pi/15$	78	78.29	43	42.96	78	78.26		
	Horizontal				78	78.28	43	43.1	78	78.26		
	Vertical	Home-pose	0	0	68	67.85	68	68.4	68	68.3		
	Horizontal				68	67.94	68	67.94	68	67.94		
	Vertical	Pull:Motor-1	0	$2\pi/15$	43	43.2	78	77.9	78	77.67		
	Horizontal				43	42.87	78	78.24	78	78.24		
	Vertical	Home-pose	0	0	68	67.87	68	68.3	68	68.24		
	Horizontal				68	67.92	68	68.1	68	68.1		
Two springs pull	Vertical	Initial	0	0	68	68	68	68	68	68	-0.17 to -0.45	0.23 to 0.39
	Horizontal				68	68	68	68	68	68		
	Vertical	Pull Motor-1&3	$7\pi/60$	$\pi/15$	54.8	54.7	90	90.06	54.8	54.72		
	Horizontal				54.8	54.7	90	90.05	54.8	54.7		
	Vertical	Home-pose	0	0	68	67.94	68	67.94	68	67.94		
	Horizontal				68	67.94	68	67.94	68	67.94		
	Vertical	Pull Motor-3&2	0	$-2\pi/15$	90	90.04	54.8	54.66	54.8	54.66		
	Horizontal				90	90.2	54.8	54.6	54.8	54.6		
	Vertical	Home-pose	0	0	68	68.1	68	67.87	68	67.87		
	Horizontal				68	67.94	68	67.94	68	67.94		
	Vertical	Pull Motor-2&1	$-7\pi/60$	$\pi/15$	54.8	54.7	54.8	54.7	90	90.06		
	Horizontal				54.8	54.7	54.8	54.75	90	90.05		
	Vertical	Home-pose	0	0	68	67.94	68	67.94	68	67.94		
	Horizontal				68	67.94	68	67.94	68	67.94		

Table 4.6 – Desired and measured solutions to the IKP of each prismatic joint for the 3-SPS-U tensegrity mechanism in the vertical and horizontal orientations for input tilt angles during one spring pull and two springs pull experiments

In Eq. (4.21), θ_{mi} and θ_{di} are the actual and desired angular positions of the pulley which are computed using the encoder data and IKP. The plot of error data for the one spring pull experiment is represented below in Figure 4.11.

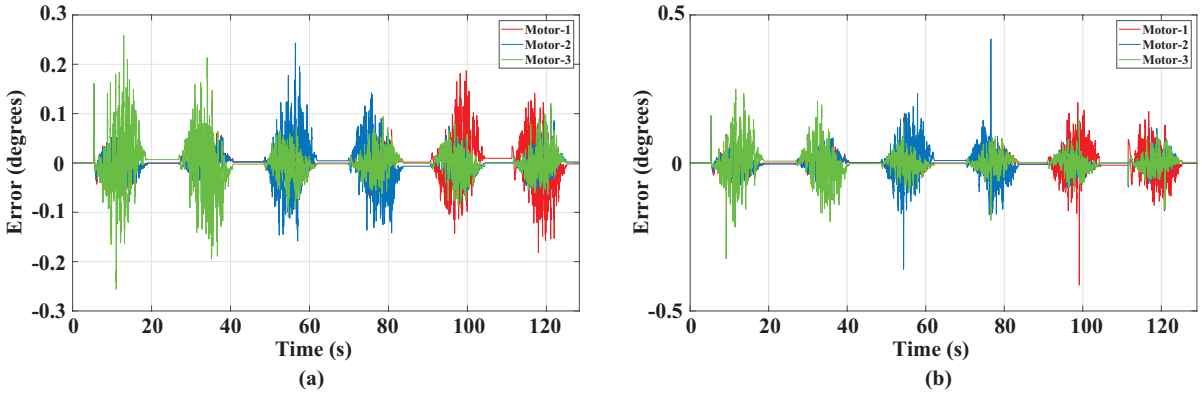


Figure 4.11 – Joint position errors along the linear trajectory in the (a) vertical and (b) horizontal orientations of the mechanism for the one spring pull experiment

From Figure 4.11, the global error between the desired and measured data lies between -0.18° to -0.46° for the cable pull and 0.19° to 0.42° for the cable push operations in both orientations of the mechanism. This also suggests that a simple PID control loop is sufficient for the tilting of the mechanism, provided the same motor units are employed. However, higher noises could be observed in Figure 4.11, which is caused by the frequency issues associated with the BB black microcomputer. At each instance of the experiment, the torque induced on each motor to perform cable pull or push is also computed. The PID control algorithm returns the measured current and by using the torque constant, the torque induced on each motor is calculated. The representation of motor torques for both orientations is provided below in Figure 4.12.

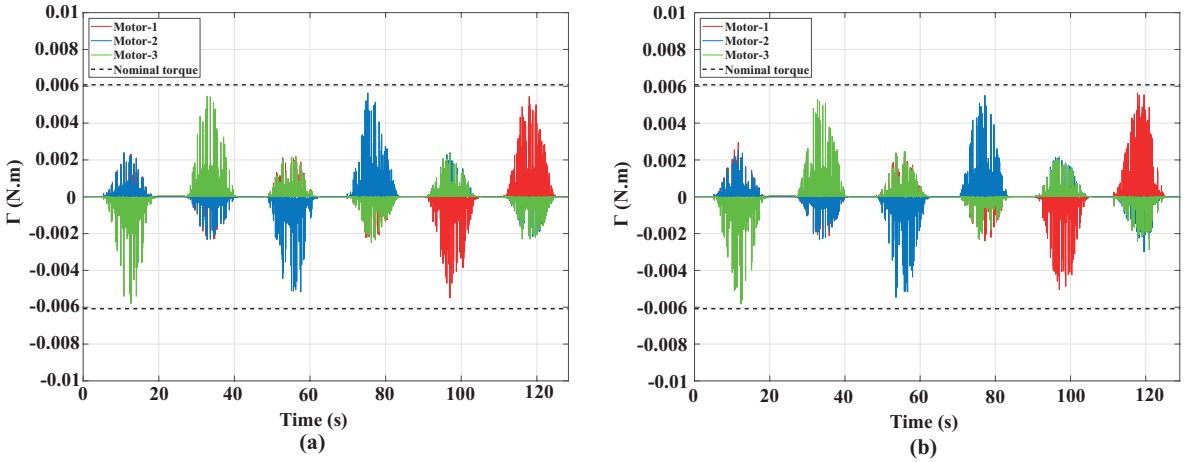


Figure 4.12 – Motor torques during operation along the linear trajectory in the (a) vertical and (b) horizontal orientations of the mechanism for the one spring pull experiment

In Figures 4.12a and 4.12b, higher noises could also be observed on the torque values. The operating torque and the peak torques measured from experiments are found to be well within the nominal motor torque which is represented by dotted black lines. The maximum measured torques on each motor during each cycle of the experiment from Figure 4.12 are provided in below in Table 4.7.

Experiment type	**Orientation**	**Input tilt ranges** (rad)		**Max. Torque** (N.m)		
		$\eta_o \rightarrow \eta_f$	$\phi_o \rightarrow \phi_f$	**Motor-1**	**Motor-2**	**Motor-3**
One spring pull	Vertical	$0 \rightarrow 7\pi/60$	$0 \rightarrow -\pi/15$	0.0021	0.0023	-0.0059
	Horizontal			0.0029	0.0025	-0.0059
	Vertical	$7\pi/60 \rightarrow 0$	$-\pi/15 \rightarrow 0$	-0.0023	-0.0023	0.0055
	Horizontal			-0.0022	-0.0023	0.0053
	Vertical	$0 \rightarrow -7\pi/60$	$0 \rightarrow -\pi/15$	0.0021	-0.0052	0.0022
	Horizontal			0.0023	-0.0055	0.0025
	Vertical	$-7\pi/60 \rightarrow 0$	$-\pi/15 \rightarrow 0$	-0.0022	0.0056	-0.0025
	Horizontal			-0.0024	0.0055	-0.0023
	Vertical	$0 \rightarrow 0$	$0 \rightarrow 2\pi/15$	-0.0055	0.0024	0.0023
	Horizontal			-0.0050	0.0022	0.0022
	Vertical	$0 \rightarrow 0$	$2\pi/15 \rightarrow 0$	0.0054	-0.0021	-0.0023
	Horizontal			0.0056	-0.003	-0.0029

Table 4.7 – Maximum measured torque on each motor for one spring pull experiment during operation in the vertical and horizontal orientations of the mechanism

From Table 4.7, it could be observed that the torque values almost remains similar for both orientations of the mechanism. The maximum torque attained by one of the motor during cable pull is around -0.0059 N.m in both vertical and horizontal orientations. The effects of gravity and self-weight of the mechanism has negligible effects on the motors torques. However, with the addition of an external load on the mechanism, these values might not remain the same, especially when the orientation is horizontal.

Results of two springs pull experiment

Followed by the one spring pull, the two springs pull experiments are performed. The input tilt angles are similar to the one spring pull and the corresponding values are provided in Table 4.6. Unlike the one spring pull, the length reached by two of the springs is around 54.8 mm over 43 mm while the third spring reaches a maximum length of 90 mm. As two springs operate to reach the desired tilt angle, the efforts are distributed among both motors to pull or push the end-effector platform. The total time taken to perform the two springs pull experiment is also 128 s, taking into account the sleep routines included in the control loop. The plot of the joint positions which provides the solutions to the IKP is represented in Figure 4.13 for both orientations of the mechanism. The desired and measured joint positions

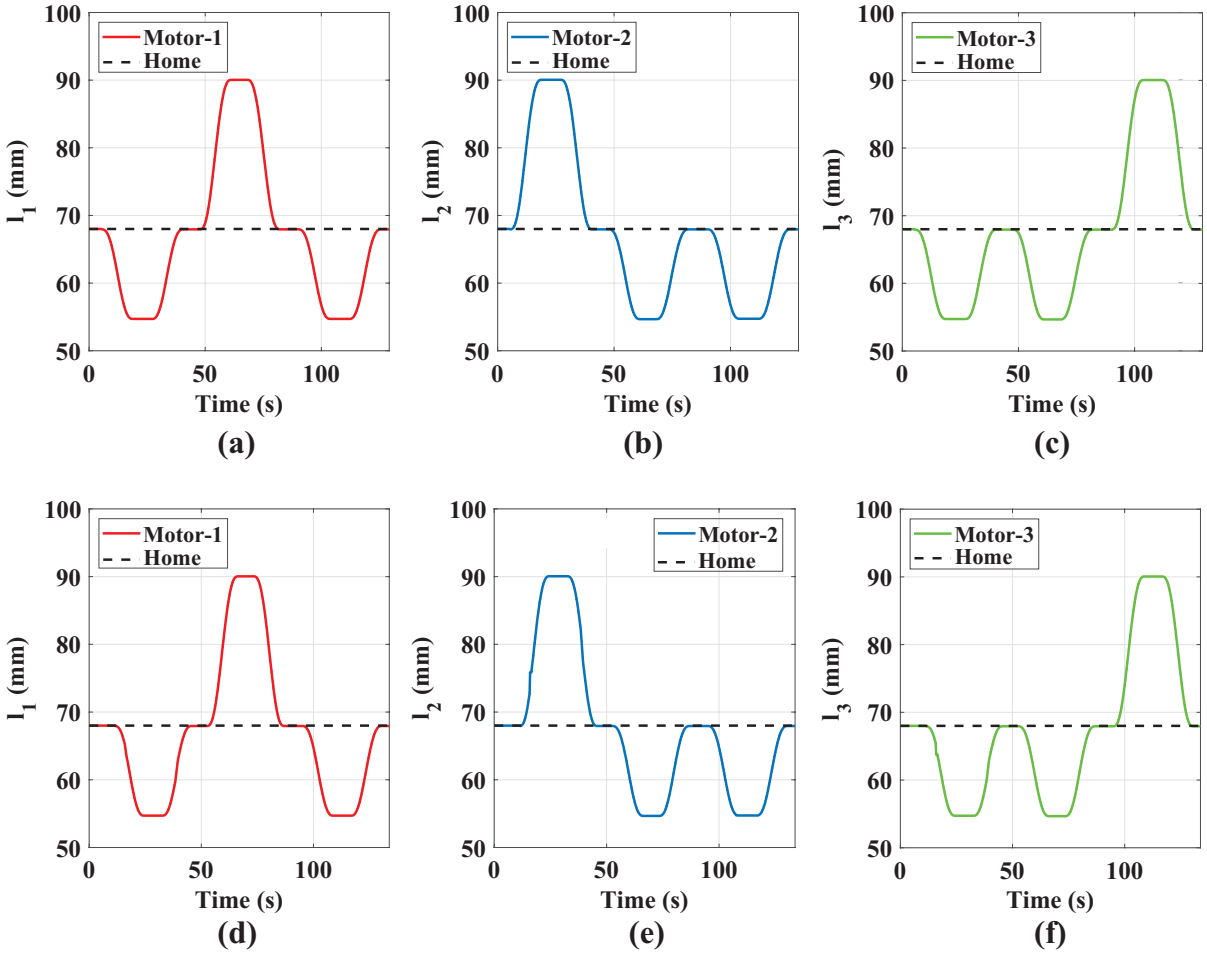


Figure 4.13 – Position of prismatic springs along the linear trajectory for two springs pull experiment in the (a),(b),(c): vertical and (d),(e),(f): horizontal orientations of the mechanism

for Figures 4.13a to 4.13f are provided in Table 4.6. By using Eq. (4.21), the error between the desired and measured angular position of the pulley is calculated. The plot of error data for the two springs pull experiments for both orientations are represented in Figure 4.14. From Figure 4.14, the global error between the measured and desired position data lies between -0.17° to -0.45° for the cable pull and 0.23° to 0.39° for cable pull and push operations in both orientations of the mechanism. Similar to the one spring pull experiments, good feedback could be observed from the PID control for the two spring pull experiments. However, noises could still be observed in the plot owing to the working condition of the BB black microcomputer. Using the current data from PID control loop, the torques on each motor are computed and their plots are represented in Figure 4.15. In the two springs pull experiment, the amount of torque required on two motors that pull the end-effector are comparatively lesser than the one spring pull experiment as the efforts are distributed among two motors. The maximum measured torque on the three motors under operation for the two springs pull experiment from Figure 4.15 is provided in Table 4.8. During the two springs pull experiment, a slight increase

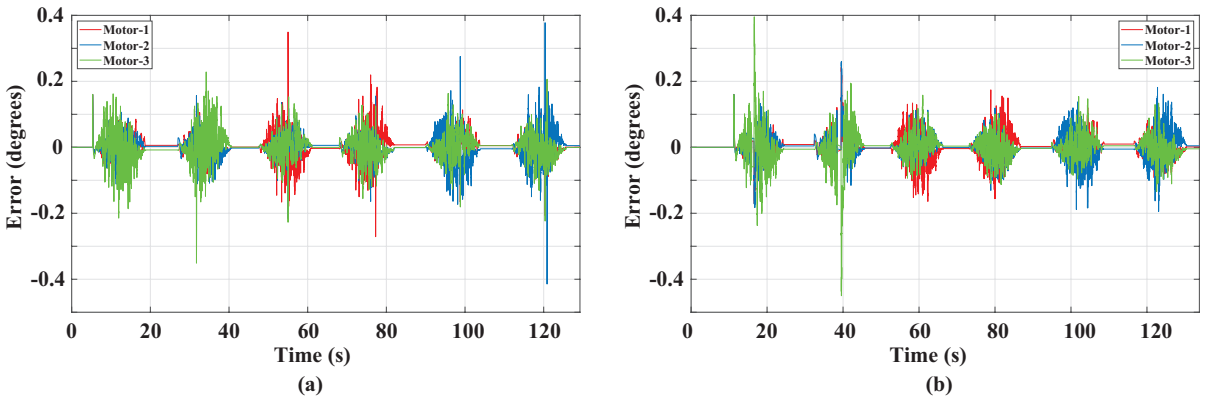


Figure 4.14 – Joint position errors along the linear trajectory in the (a) vertical and (b) horizontal orientations of the mechanism for the two springs pull experiment

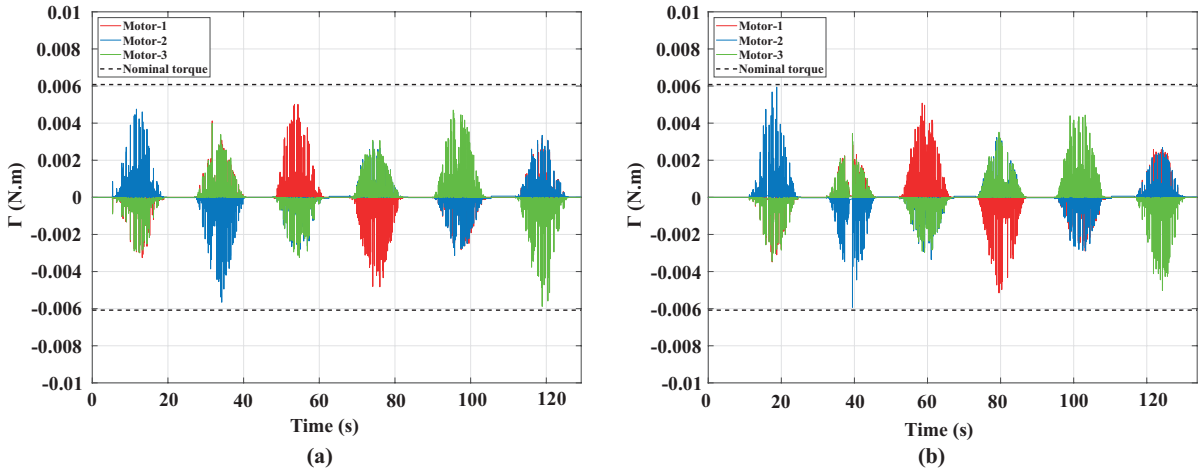


Figure 4.15 – Motor torques under operation along the linear trajectory in the (a) vertical and (b) horizontal orientations of the mechanism for the two springs pull experiment

in motor torque could be observed for the horizontal orientation of the mechanism (0.0059 N.m) over the vertical orientation (0.0047 N.m) on Motor-2 for the first cycle. This is caused by the self-weight of the mechanism during horizontal orientation as the experimental setup behaves similar to a cantilever beam. This phenomenon could be observed in almost each cycle of the experiment from Table 4.8. However, these differences are not significant and can only be better interpreted with the additional of an external load on the mechanism.

Discussions

The linear trajectory experiments were performed to understand if the mechanism could operate within the singularity free workspace. The 3-SPS-U tensegrity mechanism was made to tilt upto its maximum limit of $\pm 2\pi/15$ radians by conducting two types of experiments. In the first experiment, one of the three springs were pulled to the desired tilt limit whereas in the

Experiment type	**Orientation**	**Input tilt ranges** (rad)		**Max. Torque** (N.m)		
		$\eta_o \rightarrow \eta_f$	$\phi_o \rightarrow \phi_f$	**Motor-1**	**Motor-2**	**Motor-3**
Two springs pull	Vertical	$0 \rightarrow 7\pi/60$	$0 \rightarrow \pi/15$	-0.0032	0.0047	-0.003
	Horizontal			-0.0035	0.0059	-0.0034
	Vertical	$7\pi/60 \rightarrow 0$	$\pi/15 \rightarrow 0$	0.0025	-0.0049	0.0025
	Horizontal			0.0034	-0.0059	0.0034
	Vertical	$0 \rightarrow 0$	$0 \rightarrow -2\pi/15$	0.005	-0.0032	-0.0028
	Horizontal			0.0051	-0.003	-0.0033
	Vertical	$0 \rightarrow 0$	$-2\pi/15 \rightarrow 0$	-0.0048	0.0031	0.003
	Horizontal			-0.0051	0.0035	0.0033
	Vertical	$0 \rightarrow -7\pi/60$	$0 \rightarrow \pi/15$	-0.0028	-0.0027	0.0044
	Horizontal			-0.0029	-0.0029	0.0044
	Vertical	$-7\pi/60 \rightarrow 0$	$\pi/15 \rightarrow 0$	0.0031	0.0033	-0.0059
	Horizontal			0.0026	0.0027	-0.005

Table 4.8 – Maximum torque on each motor for two spring pull experiment during operation in the vertical and horizontal orientations of the mechanism

second experiment two springs were pulled to the desired tilt limit. In the case of one spring pull experiments, at the desired tilt position, the prismatic spring almost reached its maximum closed length. At this position, a slight movement of the fixed base was also observed and this was mainly caused by the material used for the base and end-effector of the mechanism. Rigid and lightweight metal such as aluminium can be replaced over ABS for the platforms such that the springs do not reach their minimum limits and the movements of the platform could be arrested. From the experiments, the mechanisms were able to operate smoothly and no singular configurations were reached as the input tilt limits were chosen at the boundaries of the singularity free workspace square. The error data obtained from both experiments also proved the effectiveness of the PID control algorithm. From the experiments, it was possible to identify the motor torques required for actuation of the mechanism during each cycle. Since a lightweight material was employed for the mechanism, the effects of gravity and self-weight had limited contributions to the motor torques for horizontal orientation. Thus, similar results were observed for both orientations of the mechanism. The noise data on the results of torque was refined using the Savitzky-Golay filter [sgo] in MATLAB and the corresponding plots are provided in Appendix-B. The torque range of motors is found to be around 0.002 N.m from the filtered data, which fits within the operating range of the Maxon DC-Motor [Max] employed for experiments. At each instance of the experiments, the desired input tilt angles were computed using the fifth-degree polynomial equation. At each instance, using the IKP, it could be possible to identify the actual solutions to the DKP using Eqs. (3.29) to (3.32). However, this process is computationally expensive to carry out in C or MATLAB for the entire experimental cycle.

When comparing both experiments, the prismatic retraction along one of the spring was higher for the one spring pull when compared to the two springs pull experiment. This comparison is shown in Figure 4.16. At a particular orientation of the tensegrity mechanism at the home-pose as shown in Figure 4.16b, the position of one of the prismatic links is located at a

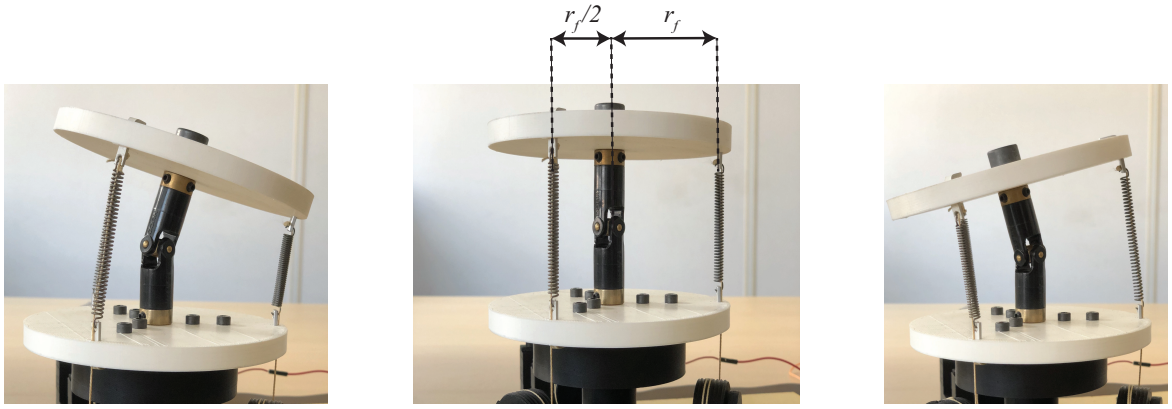


Figure 4.16 – Postures of the tensegrity mechanism during (a) One spring pull, (b) Home-pose and (c) Two springs pull phases of the experiments

distance of r_f while the other prismatic links are located at a distance of $r_f/2$ from the central axis. By the lever arm principle, during the one spring pull experiment, the prismatic link has to retract to a longer length (Figure 4.16a) when compared to the two springs pull (Figure 4.16c) experiment for attaining the desired input tilt angles. However, it has to be noted that the combined operating torques of all the motors remains the same for both experiments. The video link for the linear trajectory experiments is provided at the bottom of this page [2].

4.5.2 Circular trajectory

Followed by the linear trajectory, a circular trajectory experiment is performed on the 3-SPS-U tensegrity mechanism. At first, the theoretical curves are presented which will provide information about the simulation time with respect to the maximum angular velocity of the motor after gear reduction. Experiments are then performed for vertical and horizontal orientations of the mechanism.

Theoretical curves

For performing the circular trajectory, it is necessary to define the radius R of the circular path. This radius corresponds to the tilt limit inputs passed to the motor control units. An initial tilt along one of the motors (Motor-1) is performed using a linear trajectory. A circular path in a counterclockwise direction is then traced by the tensegrity mechanism. At the end of the circular trajectory, the linear trajectory is performed again to bring the mechanism to its home position. During the one spring pull experiment, it was observed that at the tilt limit of $\pm 2\pi/15$ radians, the prismatic springs reached its minimum limit, which is caused due to the slight movements of the base. When the circular trajectory is initiated, the mechanism will

2. Video link for the linear trajectory experiments: Click here

reach intermediate positions where each prismatic spring reaches its minimum closed length. Thus, for safer operations, a circular trajectory with a tilt radius $R{=}\pi/10$ radians, lesser than the maximum tilt limit of $\pm 2\pi/15$ radians is chosen. The circular trajectory experiment is performed in the following sequence:

Home → Motor-1: linear tilt → Circular path (counter-clockwise) → Home: linear tilt

The trace of the circular trajectory experiment for a radius of $\pi/10$ radians is generated within the workspace of the mechanism and passed as inputs for performing the experiments. This plot is represented below in Figure 4.17.

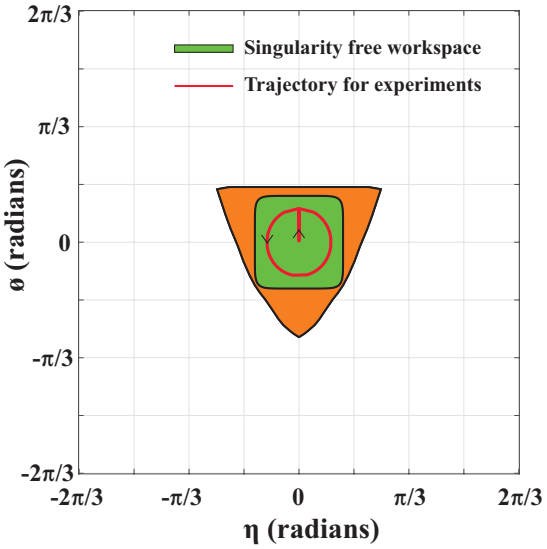


Figure 4.17 – Trace of the desired input tilt angles chosen for performing circular trajectory experiment on the tensegrity mechanism

Using Eq. (4.16), for a maximum joint velocity of 5 mm/s (calculated from motor speed), the maximum simulation time required to perform a circular trajectory is around 35 s. The total simulation time to carry out the experiments is around 53 s, taking into account the time taken for the linear tilts before and after the circular trajectory. The desired angular positions are calculated with the help of IKP and pulley radius and the corresponding values are provided below in Table 4.9.

Position	**Input tilt angles** (rad)		l_1 (mm)	θ_{d1} (rad)	l_2 (mm)	θ_{d2} (rad)	l_3 (mm)	θ_{d3} (rad)
	η	ϕ						
Home	0	0	68	0	68	0	68	0
Linear tilt	0	$\pi/10$	50	-0.9	75.9	0.39	75.9	0.39
Circular	0 → 0	$\pi/10$ → $\pi/10$	50 → 84.9 → 50	-0.9 → 0.84 → -0.9	75.9 → 84.9 → 75.9	0.39 → 0.84 → 0.39	75.9 → 84.9 → 75.9	0.395 → 0.84 → 0.39
Home: Linear	0	0	68	0	68	0	68	0

Table 4.9 – Desired angular positions of the pulley for circular trajectory with input tilt angles

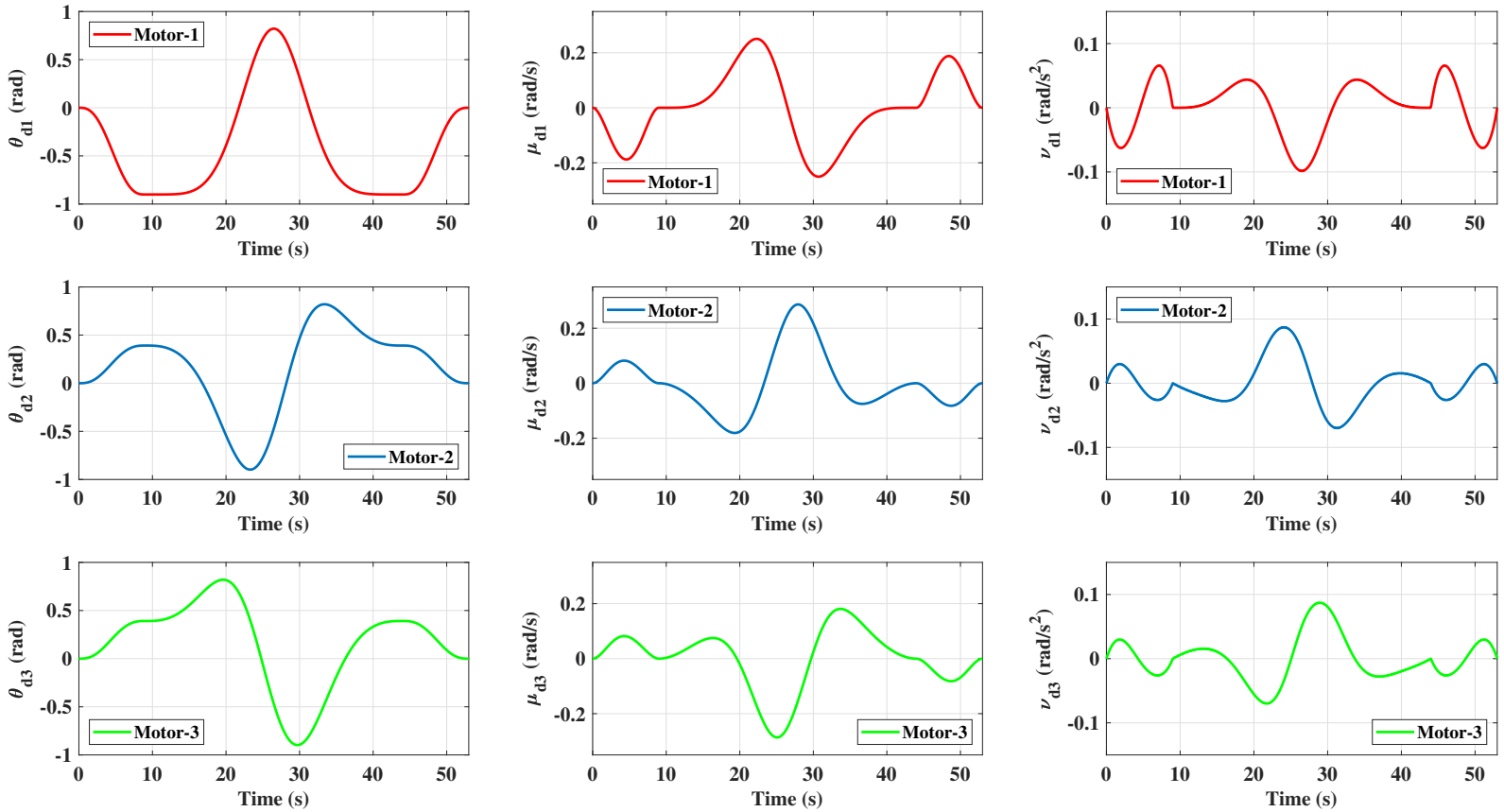


Figure 4.18 – Theoretical angular positons (θ_d), velocities (μ_d) and accelerations (ν_d) of the pulleys of Motor-1 (red) , Motor-2 (blue) and Motor-3 (green) for the circular trajectory experiment

With the help of fifth-degree polynomial equations and Jacobian matrix, the theoretical curves of the angular positions (θ_{di}), velocities (μ_{di}) and accelerations (ν_{di}) of each pulley are generated and they are represented in Figure 4.18. From Figure 4.18, it could be observed that when the mechanism follows a circular path, the maximum angular velocity of 0.25 rad/s is attained in the intermediate positions of each pulley. During the linear tilts, the maximum angular velocity is around 0.08 rad/s. The maximum angular velocities and accelerations are also calculated for the input tilt angles, which are given by $[\dot{\eta}, \dot{\phi}] = [0.1027\ ,\ 0.0892]$ rad/s and $[\ddot{\eta}, \ddot{\phi}] = [0.0387\ ,\ 0.0475]$ rad/s^2. Compared to the linear trajectory, the angular velocities and accelerations of the input tilt angles are higher. Thus, the execution times for performing the circular trajectory is comparatively higher than the linear trajectory [CCLW15]. The plots of the tilt angle positions, velocities and accelerations are provided in Appendix-B.

Results of circular trajectory experiment

The linear trajectory is performed before and after the circular trajectory for 9 s. The circular path is performed for 35 s. Owing to the sleep routines in the control loop, the time taken to initiate the circular trajectory after the initial linear tilt is longer. The total time to perform the experiments is around 124 s. The plot of the joint positions at each instance of the experiments is represented in Figure 4.19 for both orientations of the mechanism. The desired and measured solutions to the IKP for the three prismatic springs is provided in Table 4.10. The initial linear tilt is performed from 0 to $\pi/10$ radians. The circular trajectory is created from this tilt position in the counterclockwise direction and during this trajectory, the one spring pull as well as the two springs pull configurations could be observed. During the experimental cycle, at the two springs pull positions, the prismatic springs extend to a maximum length of 84.7 mm from the home position. The intermediate change in prismatic lengths could be observed in Figures 4.19(a) to 4.19(f) between 50–100 s for both orientations of the mechanism. The springs connected to each motor also reach a minimum length of 50 mm that correspond to the one spring pull experiment at $\pi/10$ radians. The zones between 75.9–84.6 mm describes the redundancy of the mechanism where one of the springs has limited effects in tilting the mechanism. Using Eq. (4.21), the error between the measured and desired position is calculated at each instance of the experiment. The plot for the error data for both orientations of the mechanism is represented in Figure 4.20. The global error data for Figure 4.20 is provided in Table 4.10. Similar to the linear trajectory, the PID control algorithm holds good for the circular trajectory as the global error range lies between -0.16° to -0.36° and 0.19° to 0.31° during cable pull and push operations for both orientations.

Using the current data obtained from the PID controller, the motor torque is calculated at each instance of the experiment and their plots are represented in Figure 4.21. The maximum measured torque on the three motors under operations during the circular trajectory experiment from Figure 4.21 is provided in Table 4.11.

Orientation	**Operation type**	**Input tilt angles** (rad)		**IKP**, l_1 (mm)		**IKP**, l_2 (mm)		**IKP**, l_3 (mm)		**Global error** (degrees)	
		η	ϕ	**Desired**	**Actual**	**Desired**	**Actual**	**Desired**	**Actual**	**Pull**	**Push**
Vertical	Initial	0	0	68	68	68	68	68	68		
Horizontal				68	68	68	68	68	68		
Vertical	Linear tilt: Motor-1	0	$\pi/10$	50	49.3	75.9	75.8	75.9	75.8		
Horizontal				50	49.9	75.9	75.7	75.9	75.7	-0.16	0.19
Vertical	Circular path	$0 \rightarrow 0$	$\pi/10 \rightarrow \pi/10$	$50 \rightarrow 84.9 \rightarrow 50$	$49.3 \rightarrow 84.1 \rightarrow 49.3$	$75.9 \rightarrow 84.9 \rightarrow 75.9$	$75.8 \rightarrow 83.9 \rightarrow 75.8$	$75.9 \rightarrow 84.9 \rightarrow 75.9$	$75.8 \rightarrow 84.1 \rightarrow 75.8$	to	to
Horizontal				$50 \rightarrow 84.9 \rightarrow 50$	$49.9 \rightarrow 84.1 \rightarrow 49.3$	$75.9 \rightarrow 84.9 \rightarrow 75.9$	$75.7 \rightarrow 84.2 \rightarrow 75.8$	$75.9 \rightarrow 84.9 \rightarrow 75.9$	$75.7 \rightarrow 83.9 \rightarrow 75.8$	-0.36	0.31
Vertical	Home position	0	$\pi/10$	68	67.94	68	67.94	68	67.94		
Horizontal				68	68.1	68	68.2	68	67.8		

Table 4.10 – Desired and measured solutions to the IKP of each prismatic joint for the 3-SPS-U tensegrity mechanism in the vertical and horizontal orientations for input tilt angles during circular trajectory experiment

Trajectory	**Orientation**	**Input tilt ranges** (rad)		**Max. Torque** (N.m)		
		$\eta_o \rightarrow \eta_f$	$\phi_o \rightarrow \phi_f$	**Motor-1**	**Motor-2**	**Motor-3**
Linear	Vertical	$0 \rightarrow 0$	$0 \rightarrow \pi/10$	-0.0049	0.0022	0.0021
	Horizontal			-0.0046	0.0022	0.0021
Circular	Vertical	$0 \rightarrow 0$	$\pi/10 \rightarrow \pi/10$	-0.0053 to 0.0053	-0.0049 to 0.0054	-0.0059 to 0.0053
	Horizontal			-0.0056 to 0.0057	-0.0056 to 0.0057	-0.0059 to 0.0056
Linear	Vertical	$0 \rightarrow 0$	$\pi/10 \rightarrow 0$	0.0042	-0.002	-0.0021
	Horizontal			0.0049	-0.0018	-0.0019

Table 4.11 – Maximum torque range on each motor during operation for the circular trajectory experiment in the vertical and horizontal orientations of the mechanism

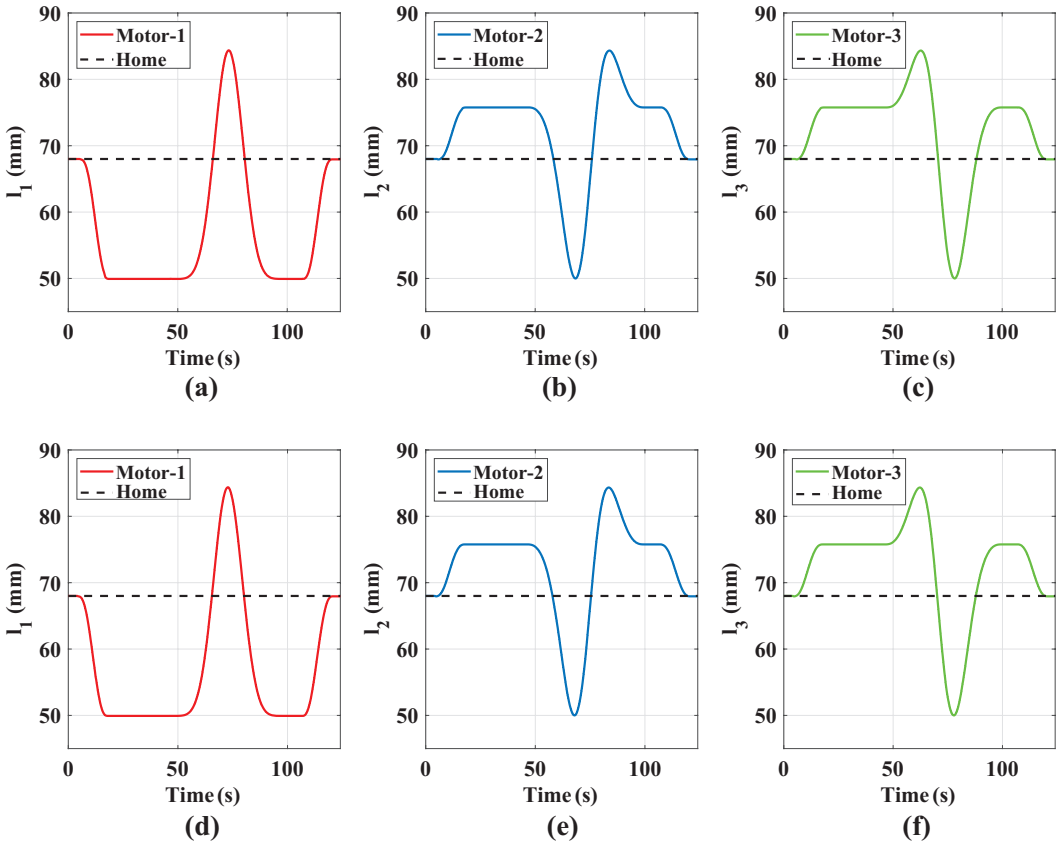


Figure 4.19 – Position of prismatic springs along the circular trajectory in the (a),(b),(c): vertical and (d),(e),(f): horizontal orientations of the mechanism

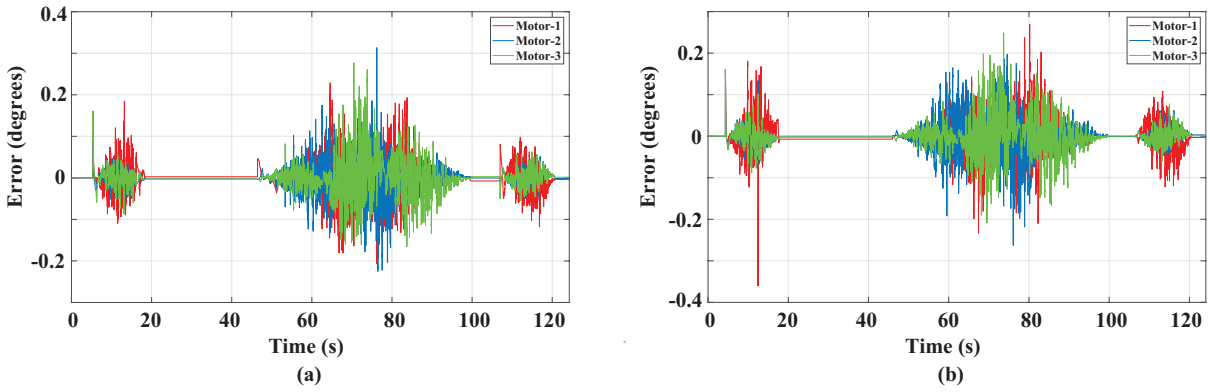


Figure 4.20 – Joint position errors along the circular trajectory for the (a) vertical and (b) horizontal orientations of the mechanism

From Figure 4.21 and Table 4.11, it could be observed that the torques measured on each motor remains similar for both orientations of the mechanism. Owing to the lightweight structure of the mechanism, there exists no significant differences in motor torques between vertical and horizontal orientations of the mechanism. The maximum torque values are observed between

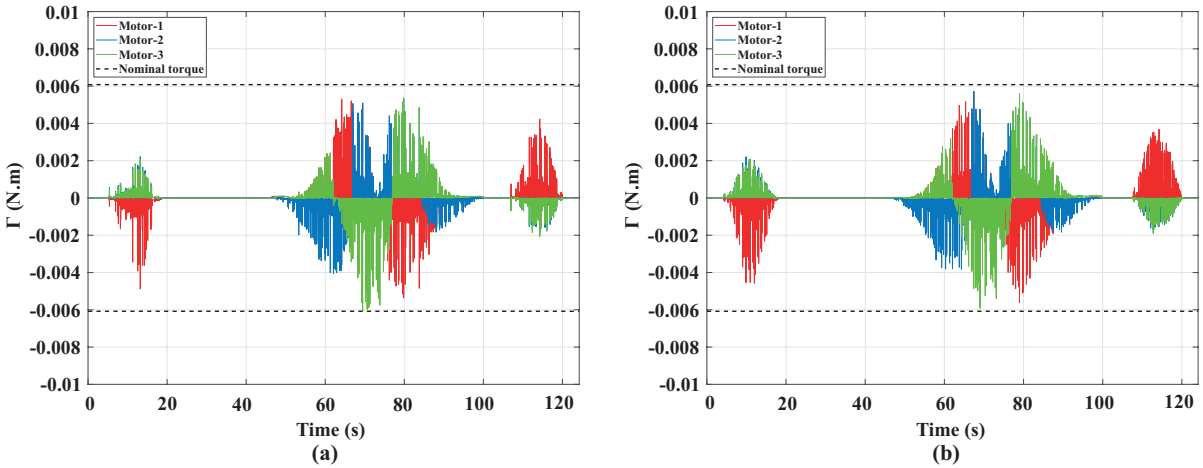


Figure 4.21 – Motor torques under operation along the circular trajectory for the (a) vertical and (b) horizontal orientations of the mechanism

-0.0059 N.m to 0.0053 N.m during the circular trajectory experiment, which is well within the nominal torque of the motor.

Discussions

The experiments with the circular trajectory allows to correlate with the theory of Tilt & Torsion. Initially, the mechanism was made to tilt along a given direction. Followed by that, the azimuth angle was chosen from 0 to -2π radians for the given tilt position to have a complete circular trajectory. The corresponding $X-Y$ Euler angle equations for this trajectory were determined and passed as inputs to the control loop to execute the circular path. Owing to the higher simulation time and incorporation of intermittent sleep periods in the control loop, the initiation time for the circular trajectory was much longer. This phenomenon could be observed in the plots of errors and torques between 25–50 s. During this period, the motor torques were completely zero. By applying Savitzky-Golay filtering to the torque data, it is found that the operating torques are around 0.002 N.m from the filtered data. The torque value after application of filters are once again within the optimum operating range of the DC-Motor. The plot of the torque data after the application of filter is provided in Appendix-B. The video link of the circular trajectory experiments is provided at the bottom of this page [3].

4.6 Conclusions

This chapter presented the experimental validation on the prototype of a 3-SPS-U tensegrity mechanism. Using DC-motors coupled with geartrain and servo controllers, the actuation of the springs of the mechanism was carried out. Through a force control algorithm written in C

3. Video link for the circular trajectory experiments: Click here

language, a simple closed-loop PID control was created to tilt the tensegrity mechanism. Two types of trajectories namely: Linear and Circular were studied and validated experimentally in vertical and horizontal orientations of the mechanism for given input tilt angles. The IKP for the tensegrity mechanism was solved experimentally through the Jacobian matrix and its time derivatives, which appeared simpler. For the linear trajectory, two sets of experiments were conducted. In the first type, one of the three motors connected to the spring was actuated to tilt the end-effector along a given linear path, whereas in the second type two of the three motors were actuated to pull the end-effector platform. Theoretical plots for positions, velocities and accelerations were presented for identifying the simulation time required with respect to the maximum velocity of motors after gear reduction. Experiments were then carried out on the prototype developed at LS2N. During the linear trajectory experiments, after each tilt operation, the mechanism was made to return to the home-pose. This was mainly performed to understand the accuracy of the PID control loop. The error between measured and desired data were lower than 0.5° and it proved the effectiveness of the PID controller. With the horizontal orientation of the mechanism, there were no significant differences observed in the motor torques. However, this parameter will not appear the same when an external loading, such as the weight of the bio-inspired robot studied in Chapter-2 is coupled along with the mechanism. It was also found that the mechanism did not attain singular configurations during the experiments as the input tilt angles were chosen at the boundaries of the singularity free workspace. However, the prismatic springs reached their minimum closed length during the one spring pull experiments as slight movements were observed on the fixed base of the tensegrity mechanism.

Followed by the linear trajectory, a circular trajectory was also performed for vertical and horizontal orientations of the mechanism. This trajectory was performed to correlate with the theory of Tilt & Torsion. Due to the movements observed on the base of the mechanism, a tilt angle radius of less than $2\pi/15$ radians within the singularity free workspace was chosen to perform a circular path. Theoretical curves were first analyzed to identify maximum simulation time, followed by which the experiments were conducted on the prototype. The results of torques on both orientations appeared similar as the self-weight had negligible effects when the mechanism was horizontal.

Owing to the frequency issue associated with the BB black microcomputer, higher noises were observed in the plots of error and torque data. Upon applying Savitzky-Golay filtering to the torque data, it was found that the operating torques were around 0.002 N.m, which is within the nominal torque value and the operating range of the Maxon DC-Motor. When the mechanism is coupled with the bio-inspired robot, the circular trajectory can be incorporated in the control law of the entire robot assembly. Based on the pipeline profile encountered, the circular trajectory can be performed to align the axis of the robot with the pipeline axis after overcoming a pipe bend or a junction.

CHAPTER 5

An optimal design of the flexible piping inspection robot

Problem statement:

1. *What is the optimum size required for the modules of a piping inspection robot to overcome a bend at 90° ?*
2. *What are the optimum sizes for the clamping modules that could be accommodated for a piping inspection robot to work inside straight and curved pipe profiles?*

5.1 Introduction

In Chapter-2, a rigid bio-inspired piping inspection robot that moves like a caterpillar was designed and analyzed. In the subsequent chapters, a tensegrity mechanism was designed and analyzed for integrating the mechanism into the bio-inspired robot for arriving at a flexible design. This robot is capable of working inside straight pipelines having diameters between 40–94 mm, provided the existing motor units and their associated spindle drives are carried over. However, if the robot encounters a pipe bend at 90° or a junction, the existing motor units prove to be oversized and this hinders the robot to overcome such bends. Another problem associated with the existing motor units is the low output velocity of 0.43 mm/s, owing to the gear reduction ratio of the spindle drive [CVB19]. This chapter focuses on determining the sizes of motor that can facilitate the robot to overcome pipe bends at 90° through an optimization

approach. Generally, optimization algorithms can be classified into deterministic and heuristic approaches [GB09]. The deterministic optimization approach solves problems in a structured manner that leads to efficient solutions with respect to design variables and constraints. On the other hand, heuristic/evolutionary algorithms operate on randomness [BB20]. Generally, the heuristic approach requires lesser computational times over the deterministic approach [RW17]. However, there are possibilities that both the algorithms can lead to solutions getting trapped in a local minimum. For the bio-inspired robot under study, a deterministic approach using *fmincon* is carried out in MATLAB. The optimization algorithm is solved within a planar test bench where the robot is constructed as multi-body blocks. Two optimization problems have been defined and solved for determining the optimal robot assembly [VCH20]. The first optimization problem deals with the determination of the motor sizes of each module without the presence of leg mechanisms. This problem helps in determining the maximum screw length of the spindle drive, which is one of the important design variables for given pipe diameter. The results coming from the first optimization problem are useful to identify suitable motor units and spindle drive from catalogue products of Maxon. The motor units and the spindle drive are chosen based on the dimensions, velocity and force factors. Also, the first optimization problem helps in determining the working diameter range for the bio-inspired robot. The second optimization problem is then carried out which deals with the sizing of leg mechanisms that could be accommodated for the results of the first optimization problem. The area of the leg mechanisms is determined through a geometric approach for the pipeline diameter range identified from the first problem. From the results of both the optimization problems, the entire bio-inspired robot is realized using CAD software.

5.2 Architecture of the flexible piping inspection robot

The architecture of the bio-inspired robot developed at LS2N is revisited in this section. The bio-inspired robot is comprised of three modules that ensure clamping and elongation phases for accomplishing the locomotion of a caterpillar [VCB19, CVB18, CVB19]. Each module consists of a brushless DC motor coupled with a spindle drive that converts rotary motion to a linear screw motion [VCB19, HCP$^+$14]. The front and rear module is composed of slot-follower leg mechanisms for having a tight contact with the pipeline walls at all instances of locomotion. The central module ensures the elongation and retraction phases which leads to robot displacement. By the study of design issues viz; passive compliance, active compliance and tilt limits [VC19b, VFCW], a tensegrity mechanism was proposed and introduced as an articulation unit for the bio-inspired robot. Through singularity and workspace analysis, two types of tensegrity mechanisms viz; 3-S$\underline{\text{P}}$S-U and 4-S$\underline{\text{P}}$S-U architectures were analyzed [VC20] in Chapter-3. The former is compatible with addressing the issue of active compliance while the latter addresses both the compliance issues. The optimization problem is carried out inside a test bench having a 90° pipe bend which is related to the issue of passive compliance. Thus, the

4-SPS-U tensegrity mechanism is considered for analysis of the bio-inspired robot based on the results identified in Chapter-3. The digital model of the bio-inspired robot that was presented in Chapter-2 as well as the 4-SPS-U tensegrity mechanism is assembled using the CATIA software to have a flexible robot assembly and it is represented below in Figure 5.1.

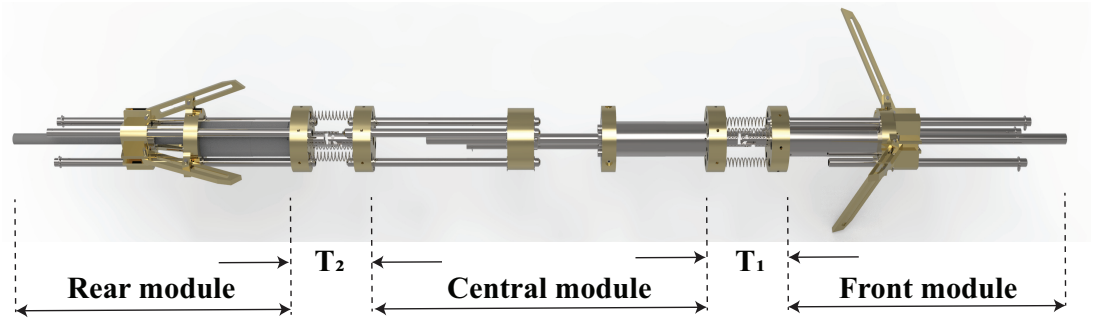


Figure 5.1 – 3D model of the existing bio-inspired robot coupled with the tensegrity mechanisms

In Figure 5.1, T_1 and T_2 represents the 4-SPS-U tensegrity mechanisms. The CAD model depicted in Figure 5.1 is constructed using the motors employed in the rigid prototype. If this robot is employed within a pipeline having diameters less than 150 mm, the modules can get trapped while encountering bends or elbows as the dimensions of the motor units prove to be oversized. For determining the dimensions of motors and leg mechanisms that can overcome such bends, an optimization approach is being implemented in this chapter.

5.3 Setting up the optimization problem

Two optimization problems are defined and solved to determine the robot assembly. The first problem deals with the determination of the motor sizes of each module without the presence of leg mechanisms. To simplify the computation, the dimensions of the tensegrity mechanism are considered fixed. The optimization problem is tested for pipeline diameters ranging from 70 to 160 mm. From the results of optimization, a suitable motor-spindle combination unit is identified from the catalogue of Maxon. Apart from the motor sizing, the existing bio-inspired robot has issues associated with velocities and cable management. Based on results obtained from the first optimization problem, the Maxon motors are identified in such a way that the following factors are addressed:

1. Velocity: An alternate spindle drive unit with a lower gear reduction ratio compared to the existing prototype (455:1) to accomplish faster displacements during locomotion
2. Hall sensors: A DC motor coupled with hall sensor units which will be useful for control phase of the robot while working inside an unknown environment [VCR19]

Once the optimal motor unit is identified, the second problem is carried out for the results of the first optimization problem to determine the size of the slot-follower leg mechanism.

5.3.1 Test bench

The optimization problems are solved inside a planar test bench. The test bench consists of a horizontal pipe section, a 90° elbow and a vertical pipe section. The optimal sizing is determined especially at the bend section wherein collision of modules against the walls of the pipelines is verified. The test bench for the optimization problem is represented below in Figure 5.2.

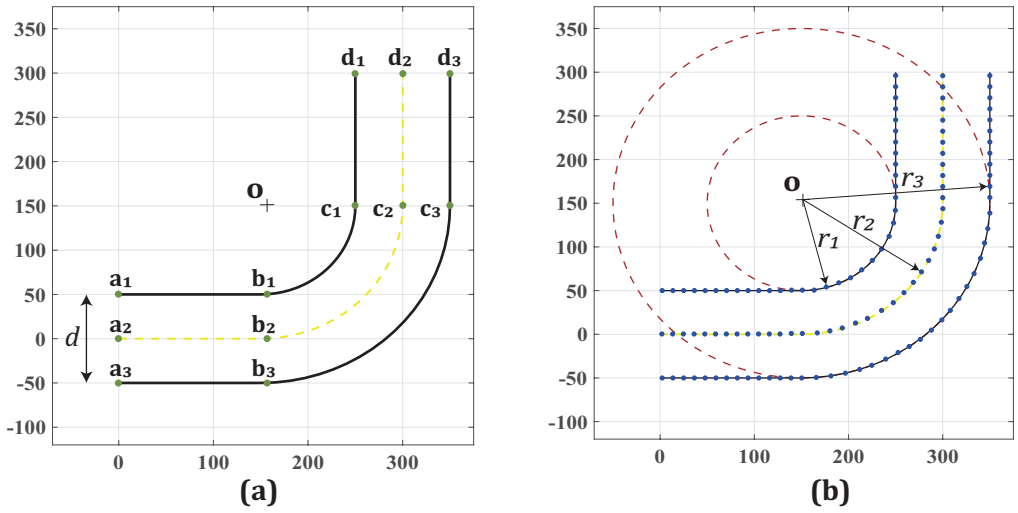


Figure 5.2 – Overview of the (a)test bench for optimization and (b) parametrization of the test bench

The test pipe represented in Figure 5.2a is generally manufactured by Numerical Control (NC) bending processes using a roller with its origin at **o** and flexible mandrels [LYZG07]. The coordinates of **o** are given by $[1.5d,\ 1.5d]$, where d is the diameter of the pipeline. With respect to **o**, the trajectories of the test bench are classified as the inner portion, centerline radius (CLR) and the outer portion. The bending radius of these trajectories represented in Figure 5.2b is given by $[r_1, r_2, r_3] = [d, 1.5d, 2d]$. The optimization problem is solved for a fixed diameter value at any instant. The values considered for d are [70, 80, 90, 100, 110, 120, 130, 140, 150, 160] mm. The pipeline geometry is constructed in MATLAB by sub-dividing into straight and bent cross-sections. The coordinate system for the test bench with respect to **o** for Figure 5.2a is given below in Table 5.1.

Coordinate	**Inner**(t=1)	**Center**(t=2)	**Outer**(t=3)
$\mathbf{a}_t$	r_1[0,0.5]	r_2[0,0]	r_3[0,-0.25]
$\mathbf{b}_t$	r_1[1.5,0.5]	r_2[1,0]	r_3[0.75,-0.25]
$\mathbf{c}_t$	r_1[2.5,1.5]	r_2[2,1]	r_3[1.75,0.75]
$\mathbf{d}_t$	r_1[2.5,3]	r_2[2,2]	r_3[1.75,1.5]

Table 5.1 – Coordinates system of pipe geometry for Figure 5.2a

A discretization is carried out between each coordinate wherein the continuous trajectory is being subdivided into discrete counterparts. This discretization is being applied to the walls of the pipelines as well as the CLR and they are represented by blue dotted lines in Figure 5.2b. The inner and outer portion discretization is employed to check for collision against modules during the locomotion of the robot. The discretization of the CLR is mainly used to move the robot as well as to rotate each module. The discretization equations for the straight sections are given by:

$$\mathbf{m} = \mathbf{a}_t + disc(\mathbf{b}_t - \mathbf{a}_t) \tag{5.1}$$

$$\mathbf{m} = \mathbf{c}_t + disc(\mathbf{d}_t - \mathbf{c}_t) \tag{5.2}$$

$$\text{where } disc = 0 : 0.01 : 1 \text{ and } t = \ 1, 2, 3$$

In Eqs. (5.1) and (5.2), $\mathbf{m}$ indicates the coordinates of discretized points between the start and end points of the straight sections. The bend section of pipe with the discretized points between $\mathbf{b}_t$-$\mathbf{c}_t$ is constructed using the equation:

$$\mathbf{m}_j = \left[r_t \sin\left(\frac{j\pi}{40}\right), -r_t \cos\left(\frac{j\pi}{40}\right) \right] \tag{5.3}$$

$$\text{where } j = 0 \text{ to } 20 \text{ and } t = \ 1, 2, 3$$

5.3.2 Modeling of robot and design variables

The robot is modeled as multi-body planar blocks with a tensegrity mechanism between each module in MATLAB. Each module comprises a Maxon brushless DC-motor coupled with a spindle drive. The digital model of a module is shown in Figure 5.3a. The robot geometry considered for the optimization problem is represented in Figure 5.3b. The modules are constructed using the geometrical equations of a rectangle and a simple computation of the same is provided in the Appendix. In Figure 5.3b, the design variables for optimization are considered in the following way:

- $[l_{k1}, w_{k1}]$: The length and width of motor and spindle drive unit (without the screw) for module k
- $[l_{k2}, w_{k2}]$: The length and width of output screw drive of spindle unit for module k

Here k indicates a module and it assumes values from 1 to 3. The spindle screw length is one of the main factors that hinder the existing bio-inspired robot to overcome a pipe bend. This is the reason why the screw unit and motor-spindle unit are considered as separate design variables. Also, it is possible to have spindle drive units with reduced screw length apart from standard sizes at Maxon [Max]. In Figure 5.3b, the reference frame of the robot $\sum_0$ is fixed at the center of gravity (CG) position of the central module. The local coordinate frames of Module-1 and Module-3 are given by $\sum_1$ and $\sum_2$. The rotation of modules about the x-axis are given by δ_1, δ_0 and δ_3 for Modules-1,2,3. The central module is rotated about $\sum_0$ whereas, for the leg

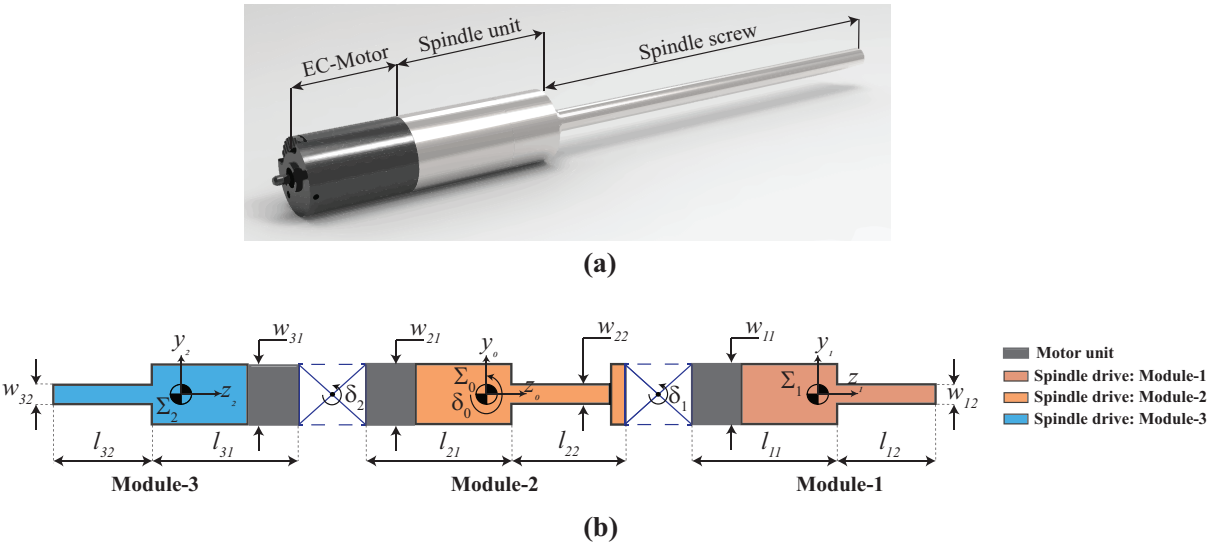


Figure 5.3 – Representation of the (a) Maxon brushless DC-motor and its spindle drive and (b) robot assembly with various design parameters for the first optimization problem

modules, the rotation is carried out about the tensegrity mechanisms. The movement of the robot is simulated in MATLAB by displacing the CG of the reference frame to each discretized point on the CLR. For any position of the robot on a given discretized point on the CLR, the rotation angle is applied to each module as well as to the tensegrity mechanisms to accomplish the rotation while passing through pipe bends. The rotation of modules during locomotion inside the test bench is accomplished in MATLAB as described below in Algorithm 3.

Algorithm 3 Rotation of modules at a given position of CLR

Inputs: Discretized CLR coordinates, Robot design parameters

Outputs: Rotation angles of modules (δ_0, δ_1, δ_2)

Initialization

for $i = 1 - n$ **do** ▷ n indicates the end point of CLR

    Placing reference frame $\sum_0$ at point i of CLR

    Extract the coordinates (z_i & y_i) of CLR point i coinciding with frame $\sum_0$

    Apply rotation angle δ_0 to Module-2 by $\delta_0 = \arctan\left(\frac{y_i}{z_i}\right)$

    Extract the coordinates of CLR point i closer to frame $\sum_1$.

    Calculate rotation angle δ_1 from the coordinates of point extracted in Step-6

    Apply rotation angle δ_1 to tensegrity mechanism coupled between Modules-1 and 2.

    Extract the coordinates of CLR point i closer to frame $\sum_2$.

    Calculate rotation angle δ_2 from the coordinates of point extracted in Step-9

    Apply rotation angle δ_2 to tensegrity mechanism coupled between Modules-2 and 3.

end for

Throughout the optimization problem, the dimensions of the tensegrity mechanism are considered to be fixed. As the orientation of the tensegrity mechanism about the z-axis is unknown at a given locomotion instant, the extreme tilt posture of the tensegrity mechanism will be considered. A configuration similar to Figure 3.17c from Chapter-3 is considered where the mounting distance between two springs is given by $2r_f$. In this pose, when one of the tilt angles reaches a maximum of $\pm\pi/4$ radians, one prismatic spring reaches a limit of 3.8 mm, while the opposing spring reaches 20.6 mm. Two other springs remain at a constant value of 12.2 mm. Thus, the design variables of the tensegrity mechanism are fixed and given by $[1.2r_f, 2r_f]$. The 2D representation of the mechanism is shown below in Figure 5.4.

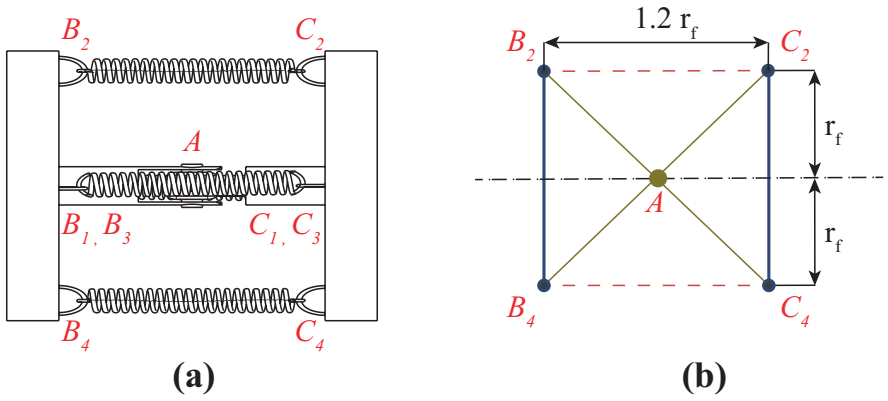


Figure 5.4 – Representation of the (a) posture considered for the 4-SPS-U mechanism in optimization problem and (b) associated fixed design parameters of the mechanism

The value of r_f indicated in Figure 5.4b is considered as 11 mm with respect to the dimensions of the flanges used in the existing bio-inspired robot [VCB19].

5.4 First optimization problem: Determination of motor sizing

A mono-objective optimization problem subject to constraints is solved using MATLAB. The leg mechanisms are not taken into account for the first optimization problem and an optimal motor sizing is determined at all discretized positions of CLR by avoiding collisions against pipeline walls. As the central module takes care of elongation and retraction phases, it is assumed for the computation that this module remains in a fully extended phase at all instances of the simulation.

5.4.1 Objective function

The objective function of the problem aims at maximizing the area of the motor units. Since the dimensions of the tensegrity mechanism are fixed, their areas are not taken into account.

The area of the robot at a given position of CLR is given by:

$$fun1 = \sum_{k=1}^{3}(l_{k1}w_{k1} + l_{k2}w_{k2}) \text{ , where } k \text{ indicates the module} \tag{5.4}$$

The area of motor units is estimated as per Eq. (5.4). The global sum of $fun1$ from Eq. (5.4) is calculated throughout the discretized points of the CLR and this will be maximized.

5.4.2 Constraint function

For each position of the robot on the CLR, the collision of modules is checked against the pipeline walls. Inequality constraints are defined in the optimization problem, which will ensure that the modules avoid collision during movement inside the test bench, especially at the bends. A complete discretization of each module is performed. The discretization equation is similar to Eqs. (5.1) and (5.2). The discretized robot modules inside the test bench are represented below in Figure 5.5a.

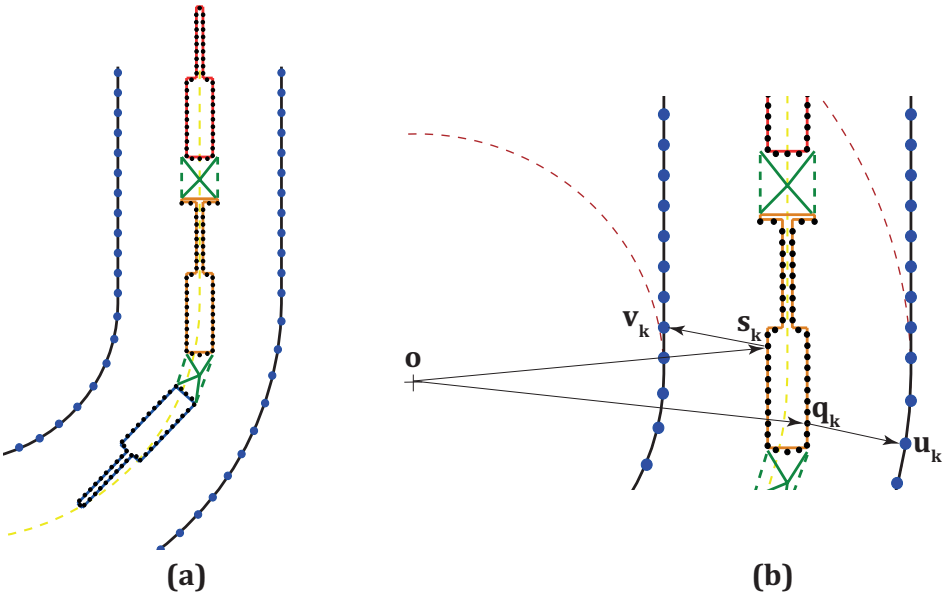


Figure 5.5 – Representation of the (a) discretized robot assembly and (b) extraction of coordinates from the discretized model for defining constraints

For each position of reference frame $\sum_0$ on the discretized CLR trajectory, the coordinates of points closer and farther on each module from **o** are extracted. These points can be extracted using the *min* and *max* function of MATLAB [HH16]. The representation of the coordinates closer ($\mathbf{s}_k$) and farther ($\mathbf{q}_k$) from **o** on a module is shown in Figure 5.5b. Followed by that, the coordinate $\mathbf{v}_k$ on the inner portion closer to $\mathbf{s}_k$ and the coordinate $\mathbf{u}_k$ on the outer portion closer to $\mathbf{q}_k$ are extracted using the *min* function. With the extracted set of coordinates, the inequality constraints are defined in MATLAB which ensures that there exist no collisions

between the modules and the pipeline walls. The constraint equations for the position of the robot assembly on a discretized CLR point are thus defined by:

$$g_k : \|\mathbf{o} - \mathbf{q}_k\| \leq \|\mathbf{o} - \mathbf{u}_k\| \tag{5.5}$$

$$g_{k+1} : \|\mathbf{o} - \mathbf{s}_k\| \geq \|\mathbf{o} - \mathbf{v}_k\| \tag{5.6}$$

$$\text{where} \quad k = 1, 2, 3$$

Throughout the locomotion sequence, the collision of each module against pipeline walls is checked using Eqs. (5.5) and (5.6). The pseudo-code for the constraint function and its working in MATLAB is provided below in Algorithm 4.

Algorithm 4 Definition of constraint equations for the first optimization problem

Inputs: Discretized CLR coordinates, Robot design parameters
Outputs: Inequality constraint equations (g_1,g_2,g_3,g_4,g_5,g_6)

Initialization
for $i = 1 - n$ **do** ▷ n indicates the end point of CLR
  for $k = 1 - 3$ **do** ▷ k indicates the module number
    for $j = 0 : 0.01 : 1$ **do** ▷ Discretization
      Discretize each face of each module
    end for
    Extract the coordinate ($\mathbf{s}_{ik}$) on module k closer to $\mathbf{o}$ using min
    Extract the coordinate ($\mathbf{q}_{ik}$) on module k farther to $\mathbf{o}$ using max
    for $l = 1 - n$ **do** ▷ For each discretized point on inner & outer portions
      Calculate $\|\mathbf{u}_l - \mathbf{q}_{ik}\|$
      Calculate $\|\mathbf{v}_l - \mathbf{s}_{ik}\|$
    end for
    Extract the coordinate from $\mathbf{u}_{ik}$ on outer portion closer to $\mathbf{q}_{ik}$
    Extract the coordinate from $\mathbf{v}_{ik}$ on inner portion closer to $\mathbf{s}_{ik}$
    Check inequality constraints (g_k): $\|\mathbf{o} - \mathbf{q}_{ik}\| \leq \|\mathbf{o} - \mathbf{u}_{ik}\|$
    Check inequality constraints (g_{k+1}): $\|\mathbf{o} - \mathbf{s}_{ik}\| \geq \|\mathbf{o} - \mathbf{v}_{ik}\|$
  end for
end for

5.4.3 Problem statement

With the objective function and constraint function being defined, the first optimization problem can be stated as:

$$\text{maximize: } \sum_{i=1}^{n} fun1(\mathbf{x})$$
$$\text{subject to the constraints: } g_1,\ g_2,\ g_3,\ g_4,\ g_5,\ g_6$$

$$\text{where } \mathbf{x} = [l_1, w_1, l_2, w_2]^T,$$
$$i = 1..n \text{ indicates the discretized CLR points}$$

The design variables for each module which were considered as $[l_{k1}, w_{k1}, l_{k2}, w_{k2}]$, resulting in 12 variables are reduced to 4 variables which are given by $[l_1, w_1, l_2, w_2]$. This is because an identical Maxon motor-spindle drive unit will be employed in the three modules. The objective and constraint functions are solved using the $fmincon$ function in MATLAB. The design variables are subject to lower and upper bounds to have a closer interpretation of the catalogue parts of Maxon. The existing prototype uses a brushless DC-motor of diameter 16 mm coupled with a spindle drive GP 16 S. The overall length of the motor-spindle unit is 58 mm with a screw length of 102 mm. The advantage of using brushless motors is that they offer hall sensors that can be useful in the control phase of the robot to determine pipeline diameters inside an unknown/closed environment [VCR19]. The lower (lb) and upper bounds (ub) for the design variables of the optimization problem are set as follows:

1. Parameter l_1: This parameter corresponds to the length of the motor-spindle unit employed in each module. The lower bound is set at 40 mm and the upper bound is set at 60 mm.
2. Parameter w_1: This parameter corresponds to the diameter of the motor-spindle unit. Since Maxon offers DC-Motors with Hall sensors starting from 16 mm, the lower bound is set at 10 mm while the upper bound is fixed at 16 mm. Values higher than 16 mm series are not considered as this may affect the length parameters of the motor-spindle drive or the screw length.
3. Parameter l_2: This corresponds to the length of the screw coming from the spindle drive. This is an essential parameter as it not only plays a vital role in avoiding collisions but also accommodates the slot-follower leg mechanism. The lower bound is set at 20 mm while the upper bound is set at 102 mm which is the maximum length offered by Maxon spindle drives.
4. Parameter w_2: This corresponds to the diameter of the screw. The lower bound is set at 2 mm and the upper bound is set at 5 mm inline with the catalogue of Maxon [Max].

Thus, the lower and upper bounds for the first optimization problem are given by lb = [40, 10, 20, 2] and ub = [60, 16, 102, 5]. Spindle drives of Maxon offer lead screws and ball screws. The existing prototype uses a ball screw type as it has a higher load carrying capacities and this could be useful when the leg mechanisms are coupled along with the drive. The optimization algorithm is executed with a constant value of diameter throughout the section depicted in Figure 5.2a. The algorithm will be evaluated one after the other for pipe diameters starting from 70 to 160 mm. For faster convergence of the problem and also to satisfy bounds at all iterations, the Sequential Quadratic Programming (SQP) algorithm [MAT] is employed within $fmincon$. The tolerance value of the guess variables and the objective function is set at 10^{-9} for the optimization problem.

5.4.4 Results and discussions

As the definition of objective and constraint functions does not have complex parameters, a faster convergence is obtained. The dimensions of motor units obtained from the optimization algorithm for the various pipe diameters are provided below in Table 5.2.

Pipe diameter d (mm)	l_1 (mm)	w_1 (mm)	l_2 (mm)	w_2 (mm)
70	41.4	12.7	20	2
80	47.44	14	24.7	2
90	50.95	16	32	5
100	52.72	16	45.17	5
110	55.02	16	58.25	5
120	58.21	16	67.33	5
130	60	16	84.4	5
140	60	16	86.54	5
150	60	16	98.8	5
160	60	16	102	5

Table 5.2 – Results obtained for the first optimization problem in MATLAB

From the results of the optimization provided in Table 5.2, a gradual increase in all design parameters could be observed with the increase in pipe diameter. During the simulation of the robot in MATLAB, the minimum and maximum lengths of the tensegrity mechanism were found to be 7.2 mm and 18.33 mm. These limits are well within the estimated numerical values of the tensegrity mechanism at $\pm\pi/4$ radians. Thus, a stacked model is not necessary for performing the simulations with a modified design. Based on the results obtained in Table 5.2, a list of possible motors that match with the dimensions are identified from the catalogue of Maxon and it is provided in Table 5.3. The results obtained for 70–80 mm diameter pipe ranges provide the least possible motor sizing. The nearest catalogue part for this series is the RE 8 DC-Motor (463223) coupled with an 8 mm diameter spindle drive (473645) with a maximum screw length of 56 mm. This combination is not compatible with the other diametrical ranges as they prove to be undersized. Also, this DC-Motor does not have Hall sensors and they require an additional encoder for having a position control algorithm. Apart from that the efficiencies and feed force factors for this series are significantly low [Max] and this might be an issue when the leg mechanisms are accommodated. Thus, the results of 70–80 mm diameter pipelines are not considered for further analysis. For the 90–120 mm diameter range pipelines, two possible motor-spindle combinations are identified. The first solution is the combination of 320178 RE 16 DC-Motor coupled with a 424731 16 mm diameter spindle drive. Another combination uses the 283828 EC-Motor coupled along with the 424731 spindle series.

Type	**Series**	l_1 (mm)	w_1 (mm)	l_2 (mm)	w_2 (mm)	**D70**	**D80**	**D90**	**D100**	**D110**	**D120**	**D130**	**D140**	**D150**	**D160**
Motor Spindle	463223 473645	31.9	8	20	3										
Motor Spindle	320178 424731	50.4	16	32	5										
Motor Spindle	283828 424731	46.4	16	32	5										
Motor Spindle	283828 424744	51.5	16	45	5										
Motor Spindle	283828 424745	62.4	16	70	5										

Table 5.3 – Identification of motor-spindle series from catalogue of Maxon [Max] based on results of optimization where red indicates over-sizing, green indicates optimal sizing and gray indicates under-sizing

Type	**Series**	**Gear ratio**	**Feed force** (N)	**Efficiency** (%)	**Length (Motor+Spindle)** (mm)
Motor Spindle	283828 424731	5.4:1	189	87	46.4
Motor Spindle	283828 424744	29:1	331	79	51.5
Motor Spindle	283828 424745	104:1	403	71	55.1

Table 5.4 – Technical specifications of motors from Table 5.3 for the results of 100 to 160 mm diameter pipes

The first solution offers a DC-Motor with a higher velocity however, the disadvantage of employing the 320178 series motor is that it requires an additional encoder if a position control algorithm is incorporated. It has to be noted that the screw length obtained for the 90 mm diameter pipe is 32 mm. This length might not be sufficient enough to accommodate the slot-follower leg mechanism, however, this can only be verified in the second optimization problem. For the 100–120 mm diameter pipeline, the 283828 EC-Motor coupled with 424744 spindle drive is identified. A screw length of 45 mm is chosen for this combination. This combination, however, appears to be undersized for the 130–160 mm diameter pipes. Another combination that uses the 283828 EC-motor along with the 424745 spindle drive with a screw length of 70 mm is identified, which is optimal for the 120–150 mm diameter pipes. From the list provided in Table 5.3, the last three motor-spindle combinations can be potentially accommodated for 90–160 mm diameter pipes provided the screw length is modified. For the 160 mm diameter pipe, all the combinations prove to be undersized. However, a motor-spindle combination with 90 mm screw length can be compatible for 150–160 mm diameter pipes. In order to have an optimal result that can be suitable for a smaller diameter pipeline ranges, the results of 100–120 mm range is considered. The screw length is thus fixed at 45 mm. The last three solutions from Table 5.3 are ideal for this diameter range. The feed force, gear ratio and efficiencies of these motor-spindle combinations are provided in Table 5.4. The EC-motor series 283828 is chosen for all these combinations. The 424731 spindle drive offers the highest velocity. However, the feed force offered by this series is low. This might pose a problem during static phases, especially when the leg mechanisms are coupled with the screw drive. The 424744 and 424745 series, on the other hand, offers better feed forces. The 424745 series offers a feed force similar to the 424749 series which is used in the existing prototype. The 424744 series, on the other hand, offers a lesser feed force but this factor will be sufficient to accommodate the leg mechanisms based on experiments conducted on the existing prototype [VC19b]. Also, the velocity offered by 424744 series is higher than 424745. This will address the issue of low velocity that is associated with the existing prototype [CVB19]. Taking into account the length, efficiency and velocity factors, the 283282 EC-motor coupled with 424744 spindle drive within a working diametrical range of 100–120 mm is considered for further analysis. For ensuring a simpler procedure for the control sequence of the prototype, the same motor-spindle combination is chosen for the central module like the leg modules.

5.5 Second Optimization problem: Design of the slot-follower leg mechanism

With the optimal motor sizing determined from the first optimization problem, the next step is to determine the geometry of leg mechanisms that could be accommodated within the screw length. The ratio of screw length to the pipe diameter has a greater influence on the geometry of the leg mechanisms. The geometry of the slot-follower leg mechanism and its assembly on

the existing robot is represented below in Figure 5.6.

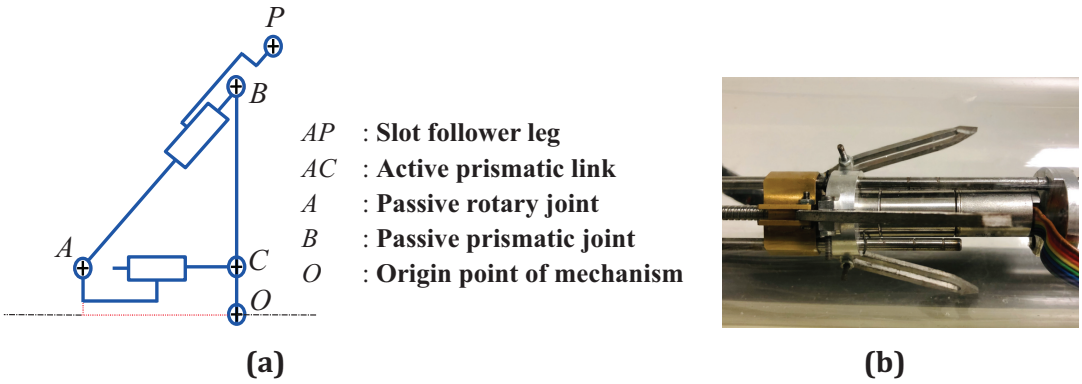


Figure 5.6 – Representation of the (a) geometry of slot-follower leg mechanism and (b) assembly of the mechanism on the existing prototype [VC19b, HCP+14]

By using the inverse kinematic model, the dimensions for the leg mechanism were identified in [HCP+14] by a multi-objective optimization approach. In this chapter, a simple mono-objective optimization problem is solved wherein the dimensions of the leg mechanisms are identified through a geometrical approach. The maximization of the area occupied by the space of leg mechanism is carried out for determining the leg length that can work inside 100–120 mm diameter pipelines.

5.5.1 Modeling of the leg mechanisms and design variables

The front and rear module assembly comprise three legs assembled at 120° with respect to each other [VC19b]. These legs ensure tight contact during static and dynamic phases of the robot. For a given locomotion instance, the orientation of the robot about its axis is unknown. As the optimization problem is solved within a planar test bench, two-dimensional projections of the leg mechanisms at two extremities are possible and these projections are represented in Figure 5.7. In the first possible extremity, the 2D projection of the robot yields a view where the distance between the central axis of the robot and the contact point of one of the legs is equal to the radius r of the pipeline. This representation is shown in Figures 5.7a and Figures 5.7b. The second possible extremity occurs when the 2D projection of the robot yields a view where two sets of legs establish contact with the walls of the pipeline. In this scenario, the distance between the central axis of the robot and the contact point with the pipeline wall is given by $r\cos(\theta)$, where θ is the angle between the leg and $y-z$ plane. This representation is shown in Figures 5.7c and Figures 5.7d. For the second optimization problem, a geometry similar to Figure 5.7c is considered such that collision will be verified on either side against the pipeline walls. For the computation, it is considered that $r\cos(\theta)$ is equal to r. Thus, the geometry of the leg mechanisms that will be assembled on the robot is depicted in Figure 5.8. For the second optimization problem, the central module dimensions are carried over from the results

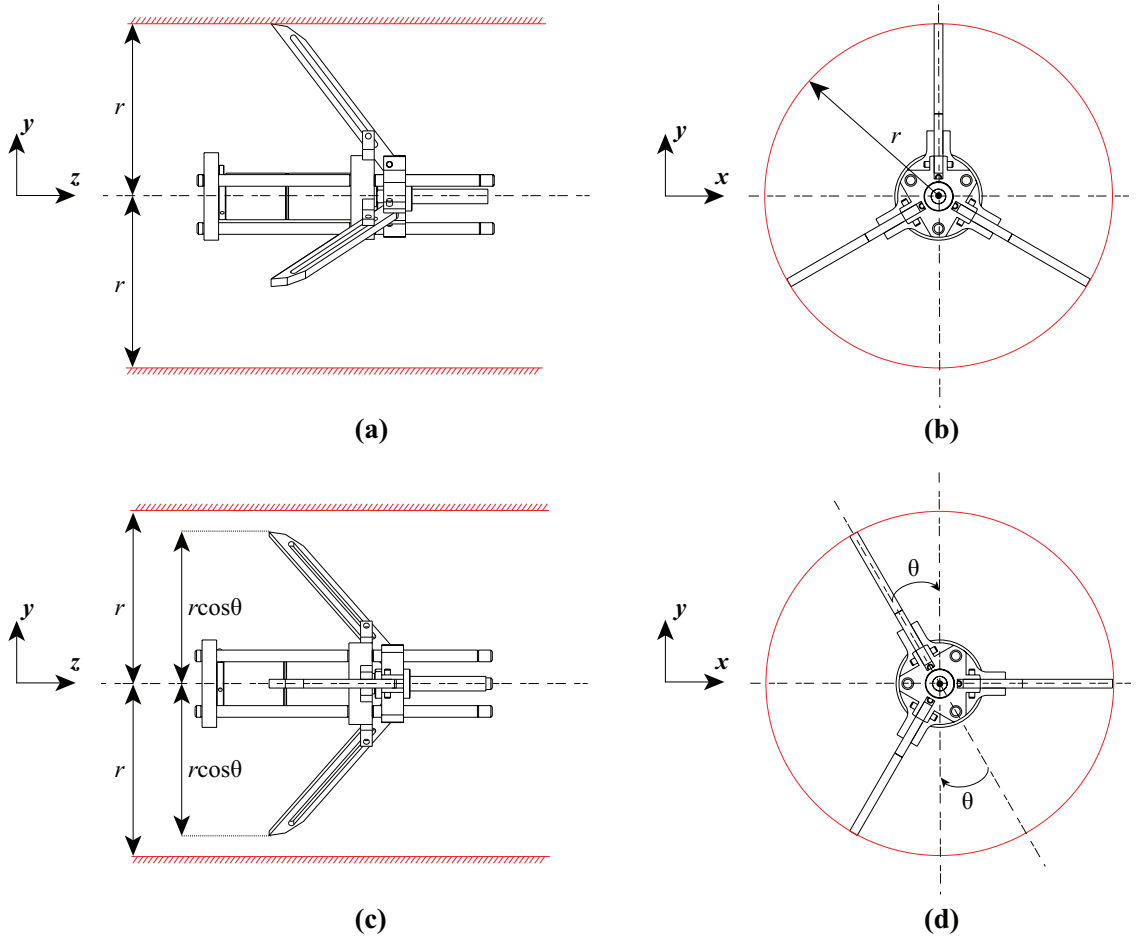


Figure 5.7 – Representation of the front and side views of the two possible orientations of the leg mechanism at the extremities caused by the rotation of the robot about z-axis

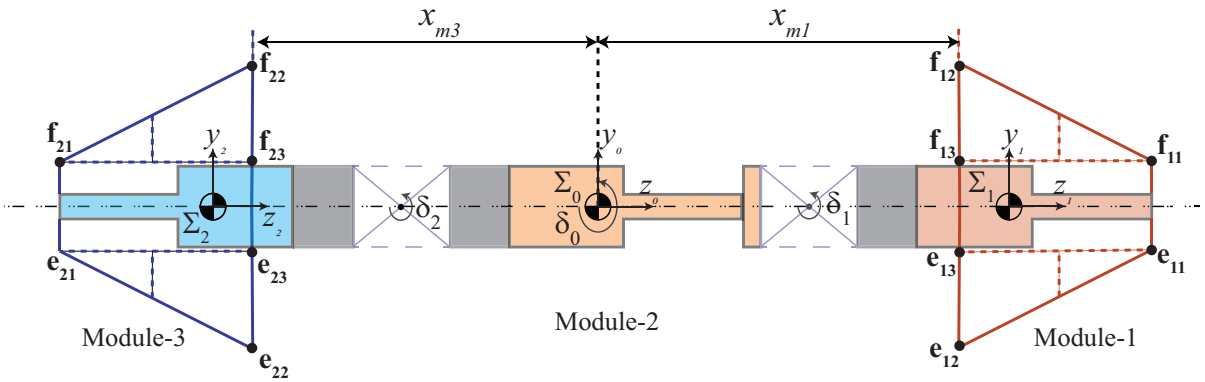


Figure 5.8 – Representation of the geometry of the slot-follower leg mechanism on the motor modules

of the first optimization problem and considered fixed. This is because the central module has no leg masses attached to them. The reference frame for the robot is fixed at $\sum_0$ in the central module. The tip of each leg mechanism is referenced from the frame $\sum_0$ through a distance x_{m1} for Module-1 and x_{m3} for Module-3. The design parameters of the slot-follower leg mechanism during declamped and clamped phases are represented below in Figures 5.9a and 5.9b.

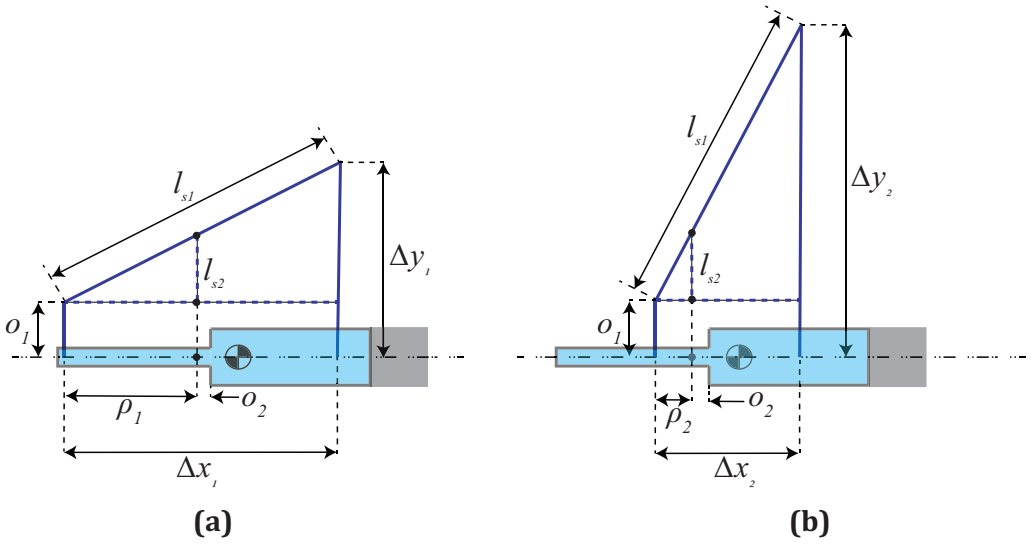


Figure 5.9 – Representation of the design parameters of the leg mechanism during (a) declamped and (b) clamped phases

The parameters of the leg mechanism from Figure 5.9a are the lengths l_{s1}, l_{s2}, the offsets o_1, o_2, the stroke lengths ρ_1, ρ_2 and the horizontal and vertical space vectors $\mathbf{\Delta x}$ and $\mathbf{\Delta y}$ [HCP+14, VC19b]. The horizontal and vertical space vectors of the leg mechanism are given by:

$$\mathbf{\Delta x} = [\Delta x_1, \Delta x_2]^T \tag{5.7}$$

$$\mathbf{\Delta y} = [\Delta y_1, \Delta y_2]^T \tag{5.8}$$

During locomotion of robot inside the pipeline, at least one set of leg mechanisms establish contact with pipeline walls. However, this sequence will be time-consuming to construct in MATLAB. In order to have faster convergence and a simpler model, it is assumed that at all positions of CLR, the leg mechanisms in both the modules remain in the declamped phase during the simulation. The design variables for the optimization problem are given by $[\Delta x_1, \Delta y_1, l_{s2}]$. The offsets o_1 and o_2 are retained as 11 mm and 7 mm, from the existing prototype as these values correspond to offsets caused by the EC-Motor unit (o_1) and spacers (o_2) used for assembly of the leg mechanisms. The parameter l_{s2} represents the vertical distance of the hinge point of the leg mechanism. This parameter plays an essential role to determine the leg length l_{s1} for a given diameter range of pipelines.

5.5.2 Objective function

The objective function of the second optimization problem aims to maximize the space occupied by the leg mechanisms. The geometry can be split into triangular and rectangular areas which are represented in Figure 5.8. The area of the leg mechanisms used in the robot at a given discretized CLR point can be calculated by:

$$fun2 = \sum_{k=1}^{2} \left(\frac{\|\mathbf{e}_{k1} - \mathbf{e}_{k3}\| \, \|\mathbf{e}_{k3} - \mathbf{e}_{k2}\|}{2} + \frac{\|\mathbf{f}_{k1} - \mathbf{f}_{k3}\| \, \|\mathbf{f}_{k3} - \mathbf{f}_{k2}\|}{2} + \|\mathbf{e}_{k1} - \mathbf{e}_{k3}\| \, \|\mathbf{e}_{k1} - \mathbf{f}_{k1}\| \right) \tag{5.9}$$

where $k = 1, 2$ indicates Module-1 and Module-3

As only two modules are being considered, the index k goes from 1 to 2, which corresponds to Modules 1 and 3. Similar to Eq. (5.4), the global sum of $fun2$ for Eq. (5.9) is estimated and it will be maximized.

5.5.3 Constraint function

Similar to the first optimization problem, the collision of the geometry of leg mechanisms is verified against the pipeline walls at each position of the CLR. A complete discretization of the leg mechanism geometry is performed using an equation similar to Eq. (5.1). The representation of the discretized leg mechanism assembled along with the optimized robot assembly is shown below in Figure 5.10a.

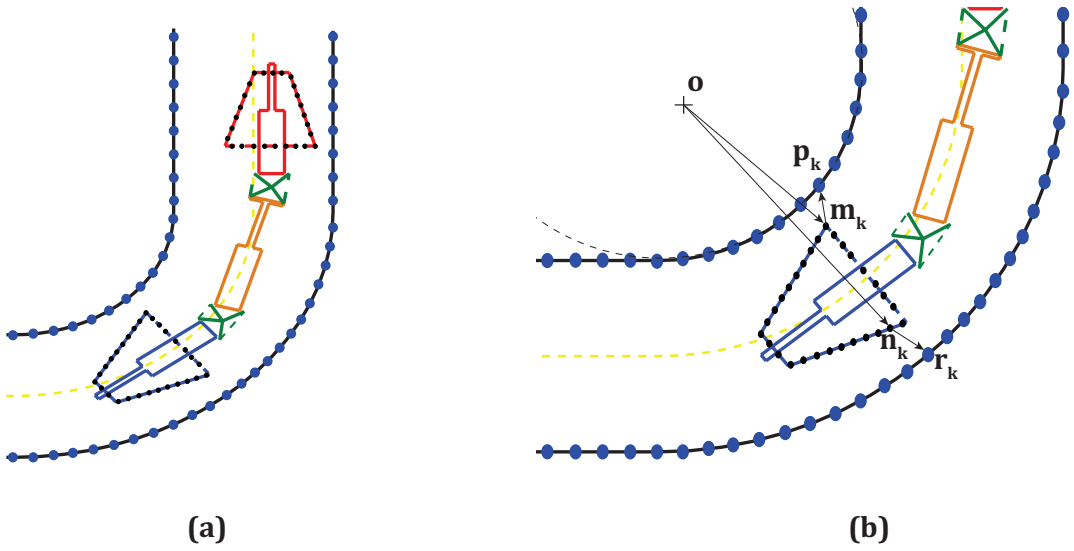


Figure 5.10 – Representation of the (a) discretized leg mechanism on the motor modules and (b) extraction of coordinates from a discretized leg mechanism for defining the constraints

For a given CLR position, the coordinates of the points closer and farther on each leg

mechanism from the point **o** are extracted. This is done using the *min* and *max* functions of MATLAB. The representation of the coordinates closer ($\mathbf{m}_k$) and farther ($\mathbf{n}_k$) on the rear leg mechanism from central coordinate **o** is shown in Figure 5.10b. The coordinates of the point $\mathbf{p}_k$ on the inner portion which is closer to $\mathbf{m}_k$ and the coordinates of the point $\mathbf{r}_k$ on the outer portion which is closer to $\mathbf{n}_k$ are extracted using the *min* function. With these extracted points, four inequality constraints are defined in MATLAB which are given by:

$$g_k : \|\mathbf{o} - \mathbf{n}_k\| \leq \|\mathbf{o} - \mathbf{r}_k\| \tag{5.10}$$

$$g_{k+1} : \|\mathbf{o} - \mathbf{m}_k\| \geq \|\mathbf{o} - \mathbf{p}_k\| \tag{5.11}$$

where $k = 1, 2$ indicates Module-1 and Module-3

In addition to the inequality constraints, two equality constraints are defined for the second optimization problem. As the simulation is performed for fully declamped phase, the length of leg mechanism l_{s1} can be calculated from Figure 5.8 by the Pythagoras theorem and it is given by:

$$l_{s1}(d) = \sqrt{(\|\mathbf{f}_{k1} - \mathbf{f}_{k3}\|)^2 + (\|\mathbf{f}_{k2} - \mathbf{f}_{k3}\|)^2} \tag{5.12}$$

where $k = 1, 2$ indicates Module-1 and Module-3

In Eq. (5.12), $l_{s1}(d)$ indicates the leg length during the declamped phase. During the clamped phase, the length l_{s1} can be calculated using Δx_2, Δy_2 and o_1 by the Pythagoras theorem from Figure 5.9b. The equation is given by:

$$l_{s1}(c) \quad = \quad \sqrt{\Delta x_2^2 + (\Delta y_2 - o_1)^2} \tag{5.13}$$

In Eq. (5.13), $l_{s1}(c)$ is the length of the leg during clamped phase. During the initialization sequence of the MATLAB algorithm for the second optimization problem, the design variable l_{s2} is calculated and set as a function of the variables Δx_1, Δy_1 during the declamped phase by similar triangles. From Figure 5.9a, the value of l_{s2} can be calculated by:

$$\frac{l_{s2}}{\rho_1} = \frac{\Delta y_1 - o_1}{\Delta x_1} \tag{5.14}$$

The dimensions of the motor units from the results of first optimization problem for the 100 mm diameter pipe from Table 5.3 is given by $[l_1, w_1, l_2, w_2]$ = [51.5, 16, 45, 5] mm. The stroke lengths ρ_1 and ρ_2 are set as 32 mm and 7 mm with reference to the 45 mm screw length of the motor unit. The reduction in stroke length is caused by the flanges and fasteners that will be used in the robot assembly. The two equality constraints for the second optimization problem are thus given by:

$$h_{k1} : \rho_1 = 32 \tag{5.15}$$

$$h_{k2} : l_{s1}(d) = l_{s1}(c) \tag{5.16}$$

where $k = 1, 2$ indicates Module-1 and Module-3

The pseudo-code of the constraint function for the second optimization problem in MATLAB is given below in Algorithm 5.

Algorithm 5 Definition of constraint equations for the second optimization problem

Inputs: Discretized CLR coordinates, Leg mechanism design parameters, Motor dimensions for 100 mm diameter pipe

Outputs: Inequality constraint equations (g_1,g_2,g_3,g_4), Equality constraint equations (h_1,h_2)

Initialization
for $i = 1 - n$ **do** ▷ n indicates the end point of CLR
  for $k = 1 - 2$ **do** ▷ k indicates the module number
    for $j = 0 : 0.01 : 1$ **do** ▷ Discretization
      Discretize each face of leg mechanism on each module
    end for
    Extract the coordinate ($\mathbf{m}_{ik}$) on module k closer to $\mathbf{o}$ using min
    Extract the coordinate ($\mathbf{n}_{ik}$) on module k farther to $\mathbf{o}$ using max
    for $l = 1 - n$ **do** ▷ For each discretized point on inner & outer portions
      Calculate $\|\mathbf{r}_l - \mathbf{n}_{ik}\|$
      Calculate $\|\mathbf{p}_l - \mathbf{m}_{ik}\|$
    end for
    Extract discretized coordinate from $\mathbf{r}_{ik}$ on outer portion closer to $\mathbf{n}_{ik}$
    Extract discretized coordinate from $\mathbf{p}_{ik}$ on inner portion closer to $\mathbf{m}_{ik}$
    Check inequality constraints (g_k): $\|\mathbf{o} - \mathbf{n}_{ik}\| \leq \|\mathbf{o} - \mathbf{r}_{ik}\|$
    Check inequality constraints (g_{k+1}): $\|\mathbf{o} - \mathbf{m}_{ik}\| \geq \|\mathbf{o} - \mathbf{p}_{ik}\|$
    Calculate $l_{s1}(d)$ using Eq. (5.12) and $l_{s1}(c)$ using Eq. (5.13)
    Check equality constraints: h_{k1} as per Eq. (5.15) and h_{k2} as per Eq. (5.16)
  end for
end for

5.5.4 Problem statement

For the second optimization problem, the diameter d of the pipeline is fixed as 100 mm. The second optimization problem which is used to estimate the size of the leg mechanism is defined by:

$$\text{maximize} \sum_{i=1}^{n} fun2(\mathbf{z})$$

subject to constraints: g_1, g_2, g_3, g_4, h_1, h_2
where $\mathbf{z} = [\Delta x_1, \Delta y_1, l_{s2}]^T$,
$i = 1..n$ indicates the discretized CLR positions

The lower and upper bounds of the design parameters for the simulation are given by lb = [40, 20, 5] , ub = [65, 40, 12]. The upper bounds for the design parameters Δx and Δy are set by considering a factor, which is 60% of the pipeline diameter. A value higher than 60% leads to convergence issues for a screw length of 45 mm. The bounds for the design parameter l_{s2} are set in line with the dimension of the existing prototype [CVB18, VCB19]. Using the *fmincon* function with the SQP algorithm, the second optimization problem is solved to determine the geometry of the leg mechanism.

5.5.5 Results and discussions

The dimensions of the slot-follower leg mechanism obtained after optimization are provided below in Table 5.5.

Parameters	**Dimensions** (mm)
Δx_1	60
Δy_1	29.3
Δy_2	62
l_{s1}	62.7
l_{s2}	9.7

Table 5.5 – Results obtained for the second optimization problem in MATLAB

Compared to the existing prototype, the parameters l_{s1} and l_{s2} are increased from 57 mm and 7 mm to 62.7 mm and 9.7 mm respectively. In the event of a straight pipeline with no bends, the slot-follower leg mechanism is capable of adjusting its size between 58.6 mm ($2\Delta y_1$) to 124 mm ($2\Delta y_2$) diameter range. When the robot is moving inside a pipeline with bends, the modules are capable of passing through such bends, provided the diameter of pipelines is strictly between 100 to 120 mm. The link for the simulation of the robot inside 100–120 mm diameter pipelines in MATLAB is provided at the bottom of this page [1]. By assembling the tensegrity mechanism, the EC-motor with spindle drive unit identified from the first optimization problem, the optimized slot-follower leg mechanism and standard fasteners such as circlips, bolts, etc., the entire robot can be realized using CATIA software. The positions of the robot inside a 100 mm diameter during its locomotion are rendered in CATIA and the sequences are represented in Figures 5.11a to 5.11e. The flexible robot resembles an "Elephant trunk" during a passive mode and it is demonstrated in Figure 5.11f.

1. Simulation of the robot assembly in MATLAB : Click here.

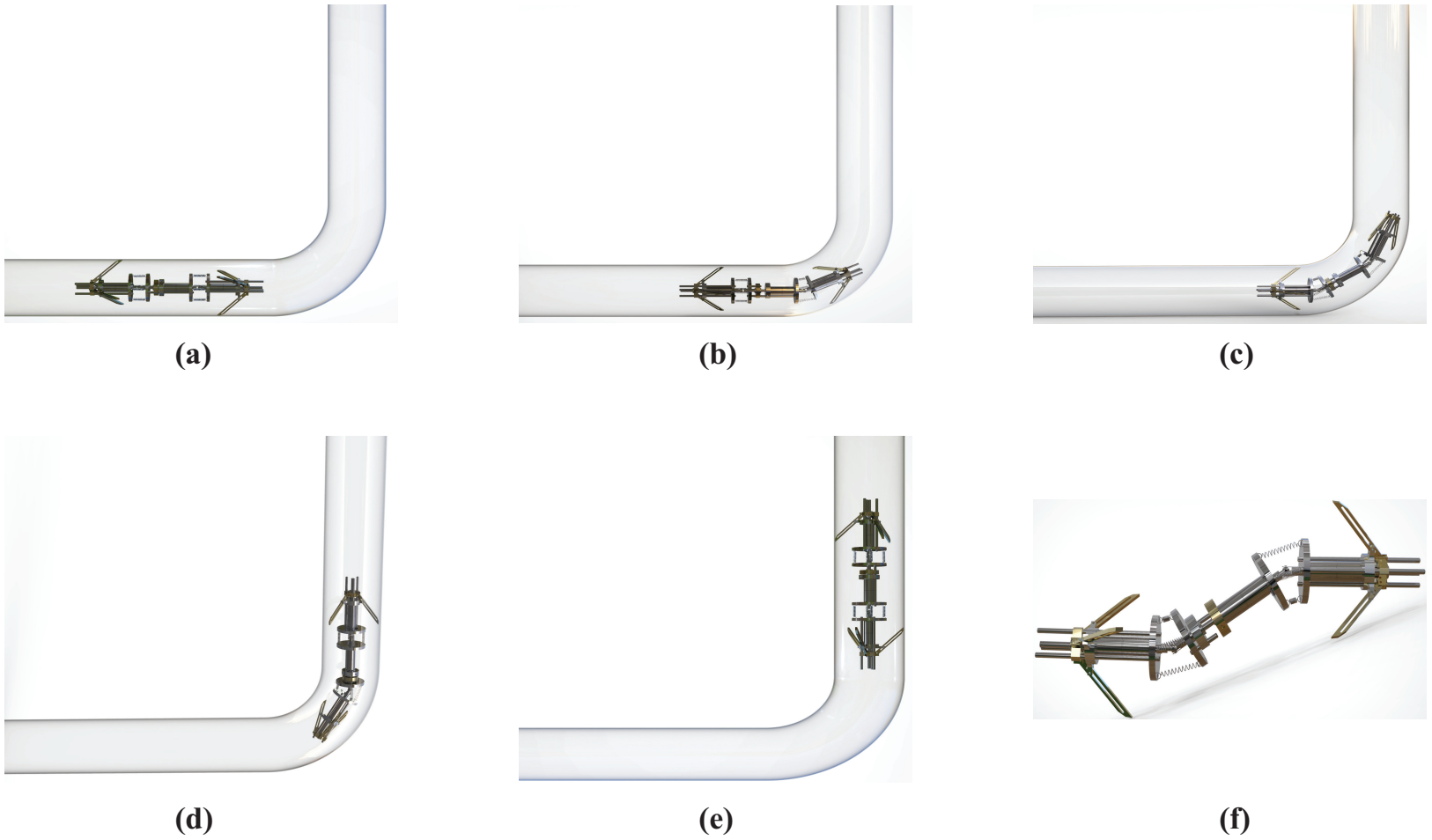


Figure 5.11 – Locomotion sequence of the optimized bio-inspired robot inside a 100 mm diameter pipe from (a) to (e) and correlation of the robot to an “Elephant trunk” in (f)

5.6 Conclusions

In this chapter, an optimization approach was employed to determine the sizing of the robot and its associated leg mechanism to pass through pipe bends. Two problems were defined and solved in this chapter, which employed the deterministic optimization approach in MATLAB. The first problem dealt with the estimation of sizes of individual motor modules without the presence of leg mechanisms inside a test bench for different pipe diameters. From the results of the first optimization problem, a series of motor-spindle combinations were identified from the catalogue of Maxon motors for a set of pipe ranges. Based on the results obtained, an intermediate-range that holds good for 100–120 mm diameter pipe was chosen. The selection of the motor-spindle unit was done based on dimensions, velocity, feed force and efficiency factors, which were not considered in the design of the initial prototype [HCP+14, CVB18, VCB19].

Followed by that, in the second optimization problem the sizing of the slot-follower leg mechanism was determined by a geometric approach that did not require the kinematic models. The dimensions of the slot-follower leg mechanism were found to be increased when compared to the existing prototype. However, with the modified architecture, the robot is capable of working inside 100–120 mm diameter pipeline range that can have straight sections and bends at 90°. The presence of leg mechanisms and tensegrity mechanism permits to have a flexible "Elephant trunk" model that allows the robot to pass through pipe bends in the event of passive compliance. A 3D model of the robot with the optimized dimensions of motors and leg mechanisms was then realized using CATIA software. The optimization algorithm can also be extended to the robotic assembly if the existing electrical actuators are replaced by a muscular pneumatic system.

CHAPTER 6

Conclusions and future works

6.1 Summary

This doctoral thesis mainly focused on the design of a bio-inspired piping inspection robot. The operational context was first defined wherein the robot must be capable of working inside pipelines having diameters less than 150 mm subject to varying diameters as well as profiles. In-line with the operational context, a rigid bio-inspired caterpillar type robot proposed by Henry et al.[HCP+14] was taken into study. Numerical force models were created for this robot and experiments were carried out on the prototype using microcontrollers inside a test pipe. A comparison was then made between the simulations and experiments. The robot was then converted into a flexible version by the incorporation of a 3-S$\underline{P}$S-U tensegrity mechanism, which can potentially operate in passive and active modes. An initial analysis using the theory of Tilt & Torsion helped in identifying a mechanism with maximum tilt limits from three models with the help of CAD algorithm and Groebner bases elimination technique in Maple. Singularity analysis with respect to Euler angles of universal joint helped in identifying the workspaces of the mechanism as well as the tilt limits which led to the proposal of the 4-S$\underline{P}$S-U tensegrity mechanism. Stability analysis under static modes was performed for both tensegrity mechanisms using an optimization approach and design parameters were modified to have stable configurations. Experiments were then carried out using a simple closed-loop PID control algorithm on the 3-S$\underline{P}$S-U tensegrity mechanism in vertical and horizontal orientations. For given input tilt angles, the IKP was solved for the mechanisms for linear and circular trajectories. An optimization approach was then introduced in the final chapter to identify the sizing of motors and the leg mechanisms for the flexible robot in order to work through 90° pipe bends. From the results of optimization, the digital model of the optimized robot assembly was realized in CAD software to demonstrate an "Elephant trunk" type model. The contributions and the solution to the research problem statements of each chapter are summarized in the following section.

6.2 Contributions

Rigid bio-inspired piping inspection robot

The second chapter defined and solved two problem statements. They are summarized as follows.

1. *What are the different ways for an inspection robot to accomplish locomotion inside a pipeline?*

In order to meet the requirements of the operational context, a rigid bio-inspired robot for piping inspection proposed by Henry et al.[HCP+14] was chosen for study. This robot accomplishes the locomotion of a caterpillar in six-steps by using DC-Motors and slot-follower leg mechanisms. From the synthesis table of locomotion techniques, which was exploited in Chapter-1, this bio-inspired robot combines the advantages of worm locomotion for movement and leg mechanisms for establishing contacts with the pipeline walls. The robot also satisfies the requirement of operation in pipelines having diameters less than 150 mm as it is capable of working inside 40–94 mm diameter straight pipelines.

2. *What are the forces required to ensure good contact with the walls of a pipeline having various orientations and diameters?*

Chapter-2 dealt with the development of static and dynamic force models of the robot proposed by Henry et al.[HCP+14]. Using Coulomb's law of dry friction, the contact forces between the legs and walls of the pipeline were determined under static modes. The contact forces also helped in identifying the forces on the actuators of the leg mechanisms using the IKP relations and force transmission factor. The forces were interpreted inside a 54 mm and a 74 mm diameter pipeline for horizontal and vertical orientations. The forces inside horizontal pipelines appeared to be larger owing to the self-weight of the robot. The dynamic force model was then developed using the recursive Newton-Euler algorithm in MATLAB. This model helped in identifying the forces on each actuator under locomotion. From the results, the dynamic effects appeared to be significantly smaller because of the low velocity of motors and low inertial forces of moving masses. Experiments were then conducted on the prototype of the bio-inspired robot inside a 74 mm PVC test pipe using a BeagleBone (BB) black microcomputer. A force control algorithm using C program was developed wherein the peak results of static force models were set as threshold limits to achieve good contact forces with pipeline walls. Higher clamping forces were observed when compared to the numerical model, which did not take into account the electromechanical factors. Also, in the experiments, the detection of final clamping forces was not rapid enough as an older version of the BB black microcomputer was employed.

Design and analysis of Tensegrity mechanisms

The third chapter addressed three problem statements and they are summarized as follows.

1. *What are the key design issues that must be addressed for the design of an articulation unit which can facilitate a robot to overcome pipe bends and junctions under passive and active modes?*

The bio-inspired robot studied in Chapter-2 was a rigid model and it is not suitable for pipelines having bends and junctions. A tensegrity mechanism that employs three tension springs and a passive universal joint was proposed as an initial solution for incorporation as an articulation unit to the bio-inspired robot. This mechanism can address the issue of passive compliance by operating passively through 90° pipe bends and through cable actuation of the end-effector platform, the issue of active compliance can be addressed in the event of junctions.

2. *What are the tilt limits for the mechanism?*

By correlating the tensegrity mechanism to a 3-S$\underline{P}$S-U architecture, the tilt limits were identified by using the theory of Tilt & Torsion through analysis of three case-study postures, namely: Pendulum, Neutral posture and Inverse pendulum. Using the CAD algorithm and Groebner bases elimination technique in Maple, the tilt limits were identified for the three postures with respect to the joint limits of the springs. This led to the selection of inverse pendulum posture owing to its higher tilt limits over the other candidates. Stability analysis was then carried out for the tilt and azimuth angles to identify design parameters which ensured that the mechanism can remain stable under static modes. For studying the workspaces of the mechanism, singularity analysis was carried out with respect to the Euler angles of the universal joint. With this analysis, the tilt limits were identified for the 3-S$\underline{P}$S-U mechanism within the singularity free workspace and were found to be less. This led to the design modification of the mechanism to a 4-S$\underline{P}$S-U architecture. From the singularity analysis, it was found that the 4-S$\underline{P}$S-U mechanism generated higher tilt limits within the singularity free workspace. Based on the comparison between both architectures, the 4-S$\underline{P}$S-U mechanism has the capability of working autonomously in the event of passive compliance whereas, for the 3-S$\underline{P}$S-U mechanism, there are chances that the mechanism can reach singular configurations during passive modes. However, the choice of tension spring also plays an essential role in these criteria. For the issue of active compliance, both architectures were ideal as cable actuation can be performed to control the mechanisms to operate within their workspaces.

3. *What are the optimal design parameters of the mechanism for having a stable configuration under static modes?*

Stability analysis for both tensegrity mechanisms was performed for the Euler angles of the universal joint. Under a fixed design parameter of $h = 1$, unstable configurations were observed for the mechanisms under static modes. By studying the Hessian matrices and second-order derivatives of tilt angles, a stochastic optimization approach using Genetic Algorithm was performed to identify optimal values of h and spring stiffness k under the presence of a preload. From the results of optimization, a common value of h= 0.6 and spring stiffness of 0.75 N/mm

was chosen for both architectures. Prototypes of the 3-SPS-U mechanism for h=1 as well as for h=0.6 were made to demonstrate the stability. Upon revisiting the singularity analysis for the modified design parameters, the tilt limits were found to be reduced for both the tensegrity mechanisms.

Experimental validation of the 3-SPS-U tensegrity mechanism

The fourth chapter addressed two problem statements and they are summarized below.

1. *What are the possible ways for actuating a 3-SPS-U tensegrity mechanism?*

The prototype of the tensegrity mechanism which was presented at the end of Chapter-3 was chosen to perform experiments. With the help of Maxon DC-Motors with encoders and servo controllers, a force control algorithm was developed in C language in BB black microcomputer for actuation of the mechanism. A planetary gearhead with a reduction ratio of 1621:1 was coupled to the DC-Motor to have a simple static model. Using pulleys, cables that pass through the springs of the mechanism were actuated from the DC-Motors for tilting the end-effector platform along a certain direction.

2. *What are the governing equations for achieving the the desired trajectory and the desired positions by the mechanism?*

As the mechanism was correlated to a parallel manipulator of type 3-SPS-U, the cartesian velocities and accelerations were possible to identify with the help of the forward kinematics Jacobian matrix and its time derivative, which appeared to be simpler. Using a simple closed-loop PID control algorithm, the IKP for the mechanism was solved for input tilt angles chosen within the singularity free workspace of the mechanism. A linear and circular trajectory was performed on the mechanism by using the fifth-degree polynomial equations and the Jacobian matrix. Theoretical curves were initially presented to identify the simulation time required to perform the experiments. Experiments were then performed on the prototype and it was found that the error between the measured data and desired data was smaller which proved that the PID control loop was effective. The experiments were performed in vertical and horizontal orientations of the mechanisms without any external loading and it was observed that the motor torques almost remained the same. The gravity forces and self-weight of the mechanism had negligible effects on the motor torques. Higher noises were observed in the plots of error and motor torque from experimental data due to the frequency issues associated with the BB black microcomputer. Upon applying a numerical filter, it was found that the motor torques were well within the operating range of the motor.

An optimal design of the flexible piping inspection robot

The fifth chapter addressed two problem statements which are summarized below.

1. *What is the optimum size required for the modules of a piping inspection robot to overcome a bend at 90° ?*

Through the addition of tensegrity mechanisms to the rigid bio-inspired robot studied in Chapter-2, a flexible design was obtained. However, the existing motor units proved to be oversized for passing through pipe bends and junctions. Chapter-5 proposed and solved two optimization problems which employed the deterministic approach in MATLAB for identifying the optimal dimensions of the motor units as well as the leg mechanisms. In the first optimization problem, the motor sizing was determined inside a planar test bench pipe that was constructed in MATLAB for pipeline diameters ranging from 70 mm to 160 mm. Based on the results of optimization, motor units were identified from the catalogue of Maxon not only based on dimensions but also based on other factors such as velocity, feed force, hall sensor feature and efficiencies. The results of 100–120 mm diameter range pipeline was chosen based on the factors specified. Also, the tensegrity mechanism was able to tilt the robot assembly and thus a stacked model was not necessary to perform the simulations.

2. *What are the optimum sizes for the clamping modules that could be accommodated for a piping inspection robot to work inside straight and curved pipe profiles?*

Followed by the identification of optimal motor units and the working diameter range of pipelines, the dimensions of the slot-follower mechanism were also determined by an optimization approach. The leg length and mounting parameters of the mechanism that could be accommodated for the results of first optimization problem were identified through a geometric approach. The dimensions were found to be a bit higher than the leg mechanisms of the existing prototype as the new model was studied for pipeline diameters ranging between 100–120 mm.

From the results of both optimization problems, the entire robot assembly was created using CAD software. An overview of the locomotion procedure through pipe bend was rendered digitally and the “Elephant trunk” posture of the flexible robot was represented.

6.3 Future perspectives

This section will summarize the open problems that has to be addressed in each chapters as well as the future perspectives of the doctoral thesis.

1. The thesis was able to address the issue of movement inside pipeline from the general issues presented in the introduction. The rest of the issues remains open. For the positioning issue, a suitable camera with a lighting source must be incorporated to understand the position of the robot inside unknown environments. A machine learning methodology can be implemented, which can train the robot to study and understand various pipe profiles, bends and deposits through multiple trails inside pipelines. Upon validation of the position issue, the inspection or intervention problems can be studied by mounting the required devices and allowing the robot to perform the desired task.

2. For the tensegrity mechanisms, only a static model was proposed to understand their stabilities. A wrench model, that takes into account the mass parameters and other forces will be done in the future to study the workspaces. Using the wrench feasible workspaces, the maximum and minimum efforts required to tilt the mechanisms can be estimated numerically. By using high-velocity motors for experiments, a dynamic model can be demonstrated and the resultant efforts can be compared with the numerical solutions.
3. The BB black microcomputer offers only three quadrature encoders for transferring information about the position. This disadvantage did not permit to perform experiments on a 4-S<u>P</u>S-U tensegrity mechanism. The latest BB blue microcomputer can address this issue as it can support four quadrature encoders.
4. The experiments on the tensegrity mechanism were performed on a scaled-up prototype in order to demonstrate the working principle. In the future, the actual size of the prototype that was considered in Chapter-3 will be realized and experiments will be performed. Owing to the COVID-19 crisis, the experiments on the tensegrity mechanism was limited to just a demonstration of simple trajectories without any additional loading. For the future, an external loading equivalent to the weight of the bio-inspired robot will be added on the scaled-up and the actual prototypes to understand their effects on the motor torques, especially when the orientation of the mechanism is horizontal.
5. The optimization approach in Chapter-5 was performed by not taking into account the dynamic parameters of the robot modules. The digital model of the robot realized in CAD software will be tested using software such as ADAMS in order to ensure that the optimized robot design can pass through pipe bends. Also, this simulation can predict if the tilt limits of the tensegrity mechanism are sufficient enough to overcome the pipe bend when the dynamic parameters of the robot assembly is taken into account.
6. The analysis of the four-bar wheel mechanism for the flexible robot has not been exploited in this thesis. For the future works, the optimization approach will be completely improved or modified in such a way that the solutions to the sizing of motors, leg mechanisms and the four-bar wheel mechanisms could be determined through a single problem statement. This might lead to the implementation of stochastic approaches using Genetic Algorithm as there are possibilities to have the definition of multiple objectives. Ideal solutions can then be chosen from the Pareto front generated by the optimization algorithm.
7. The DC-Motor with spindle drive combination was considered for the robot taking into account the nature of pipelines in nuclear industries. Pneumatic actuators are an excellent candidate for the caterpillar locomotion however, they cannot hold good inside radioactive environments. For reducing complexities of the system such as efficiency and feed force for clamping modules, an alternative solution will be to implement the artificial muscular type flexible systems for the robot modules. These systems do not require a lot of cables for their actuation and due to their flexibility, they can easily adapt to diameter changes

and overcome pipe bends. The material for the artificial muscular system must be chosen according to its compatibility with radioactive elements.

8. The BB black was employed to demonstrate and validate the locomotion principle of the robot and the tilting of the tensegrity mechanism. For the real-time application, it is difficult to have two sets of BB black operating separately for actuation of the caterpillar modules and the tensegrity mechanism of the entire robot assembly. The EPOS2 CAN programmable positioning controller of Maxon is an ideal solution when it comes to the implementation for the final industrial application. Using this controller, it is possible to have a common interface for all the ESCON controllers employed in the caterpillar modules and the tensegrity mechanisms for simultaneous operations. Another advantage is that this controller is compatible for DC-motors with incremental encoders, thereby, a position control algorithm could be incorporated for the piping inspection robot to operate inside a closed steel pipe of unknown diameter.

Personal Publications

Journal publications

PhD works

- [VCH] Swaminath Venkateswaran, Damien Chablat, and Pol Hamon. "An optimal design of a flexible piping inspection robot."*Journal of Mechanisms and Robotics* (2021): 1–37. <hal-03126584>.
- [VCB19] Swaminath Venkateswaran, Damien Chablat, and Frédéric Boyer. "Numerical and experimental validation of the prototype of a bio-inspired piping inspection robot." *Robotics* 8, no. 2 (2019): 32. <hal-02108495>.
- [CVB18] Damien Chablat, Swaminath Venkateswaran, and Frédéric Boyer. "Mechanical design optimization of a piping inspection robot." *Procedia CIRP* 70 (2018): 307-312. <hal-01763204>.

Other works

- [CMB+21] Damien Chablat, Guillaume Michel, Philippe Bordure, Swaminath Venkateswaran, and Ranjhan Jha. "Workspace analysis in the design parameter space of a 2-DOF spherical parallel mechanism for a prescribed workspace: Application to the otologic surgery." *Mechanism and Machine Theory* 157 (2021): 104224. <hal-03106584>.
- [COV19] Damien Chablat, Erika Ottaviano, and Swaminath Venkateswaran. "Self-motion conditions for a 3-PPPS parallel robot with delta-shaped base." *Mechanism and Machine Theory* 135 (2019): 109-114. <hal-02014982>.

Conference publications & Chapters in books

PhD works

- [VC20] Swaminath Venkateswaran, and Damien Chablat. "Singularity and workspace analysis of 3-SPS-U and 4-SPS-U tensegrity mechanisms." *International Symposium on Advances in Robot Kinematics*. Springer, Cham, 2020. <hal-02568290>.

- [VCH20] Swaminath Venkateswaran, Damien Chablat, and Pol Hamon. “Design of a piping inspection robot by optimization approach”. *The ASME 2020 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference,IDETC/CIE 202*0, Aug 2020, Saint-Louis, Missouri, United States. <hal-02626029v2>.
- [VCR19] Swaminath Venkateswaran, Damien Chablat, and Ramakrishnan Ramachandran. "Prototyping a piping inspection robot using a BeagleBone black board." In *24éme Congrès Français de Mécanique*. 2019. <hal-02190720>.
- [VFCW] Swaminath Venkateswaran, Matthieu Furet, Damien Chablat, and Philippe Wenger. “Design and analysis of a tensegrity mechanism for a bio-inspired robot.” In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 59230, p. V05AT07A026. American Society of Mechanical Engineers, 2019. <hal-02189437>.
- [VC19b] Swaminath Venkateswaran, and Damien Chablat. “A new inspection robot for pipelines with bends and junctions.” *IFToMM World Congress on Mechanism and Machine Science*. Springer, Cham, 2019. <hal-02098350>.
- [CVB19] Damien Chablat, Swaminath Venkateswaran, and Frédéric Boyer. “Dynamic model of a bio-inspired robot for piping inspection.” *ROMANSY 22–Robot Design, Dynamics and Control*. Springer, Cham, 2019. 42-51. <hal-01807189>.

Other works

- [CMB+20] Damien Chablat, Guillaume Michel, Philippe Bordure, Ranjhan Jha, and Swaminath Venkateswaran. “Joint space and workspace analysis of a 2-DOF Spherical Parallel Mechanism”. *European Conference on Mechanism Science*. Springer, Cham, 2020. <hal-02568293>.
- [COV18] Damien Chablat, Erika Ottaviano, and Swaminath Venkateswaran. “Self-Motion of the 3-PPPS Parallel Robot with Delta-Shaped Base.” *European Conference on Mechanism Science*. Springer, Cham, 2018. <hal-01808551>.

Posters

- [VC19a] Swaminath Venkateswaran, and Damien Chablat. “Conception d’un robot bio-inspiré pour l’inspection de la canalisation.” *Journée des Jeunes Chercheurs en Robotique*, Oct 2019, Vittel, France. <hal-02315467>.
- [VCB18] Swaminath Venkateswaran, Damien Chablat, and Frédéric Boyer. “Un robot bio-inspiré pour l’inspection de la canalisation.” *ReV Réunion-LS2N*, Sep 2018, Nantes, France. <hal-01938480>.

Appendix A: Mathematical equations

A.1 Chapter-2

Construction of the slot-follower mechanism in SIROPA:

Record[**Equations** $= [{l_0}^2 - {l_2}^2 - \rho^2, \frac{\rho\, l_1}{l_0} + x_p - \rho, \frac{l_2\, l_1}{l_0} - y_p + e, l_3 + l_0 - l_1]$, **Constraints** $= [l_2 < y_p, 0 < \rho]$, **PoseVariables** $= [y_p]$, **ArticularVariables** $= [l_1, l_2, \rho]$, **PassiveVariables** $= [l_0, l_3, x_p]$, **GeometricParameters** $= [e]$, **GenericEquations** $= [{l_0}^2 - {l_2}^2 - \rho^2, \frac{\rho\, l_1}{l_0} + x_p - \rho, \frac{l_2\, l_1}{l_0} - y_p + e, l_3 + l_0 - l_1]$, **GenericConstraints** $= [l_2 < y_p, 0 < \rho]$, **Precision** $= 4$, **PoseValues** $= []$, **ArticularValues** $= []$, **PassiveValues** $= []$, **GeometricValues** $= []$, **DefaultPlotRanges** $= []$, **Points** $= [O = [0, 0], A = [\rho, 0], B = [\rho, e], C = [0, l_2 + e], P = [x_P, y_P]]$, **Loops** $= []$, **Chains** $= [[O, C], [A, B], [B, C], [C, P]]$, **Actuators** $= [[O, A]]$, **Model** $=$ ""]

Configuration of the mechanism for $\rho = 30$ mm, $l_1 = 57$ mm and $l_2 = 7$ mm in SIROPA is shown below in Figure A.1:

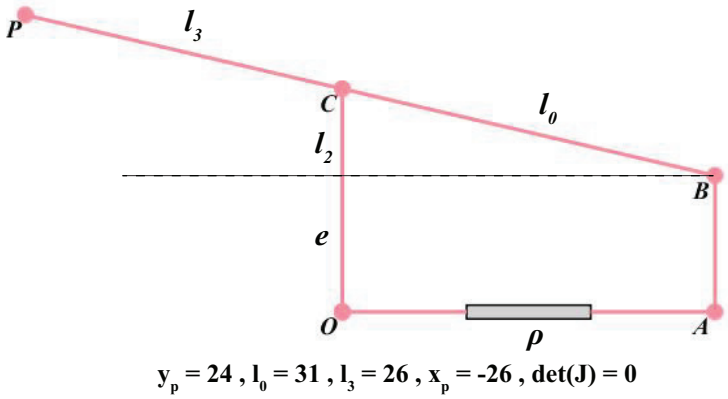


Figure A.1 – Slot-follower mechanism in SIROPA

The equations of wrench model:

$$
{}^{1}\tau_0 = \begin{bmatrix} 0 & \sin(\theta) r T_{l1} - l T_{r1} \\ T_{r1} & -\cos(\theta) r T_{l1} \\ T_{l1} & \cos(\theta) r T_{r1} \end{bmatrix}
$$

$$
{}^{2}\tau_0 = \begin{bmatrix} -\frac{\sqrt{3}T_{r2}}{2} & T_{l2}(\frac{\sqrt{3}r\cos(\theta)}{2} - \frac{r\sin(\theta)}{2}) + \frac{lT_{r2}}{2} \\ -\frac{T_{r2}}{2} & T_{l2}(\frac{\sqrt{3}r\sin(\theta)}{2} + \frac{r\cos(\theta)}{2}) - \frac{\sqrt{3}lT_{r2}}{2} \\ T_{l2} & \frac{T_{r2}}{2}(\frac{\sqrt{3}r\sin(\theta)}{2} + \frac{r\cos(\theta)}{2}) + \frac{T_{r2}\sqrt{3}}{2}(\frac{\sqrt{3}r\cos(\theta)}{2} - \frac{r\sin(\theta)}{2}) \end{bmatrix}
$$

$$
{}^{3}\tau_0 = \begin{bmatrix} \frac{\sqrt{3}T_{r3}}{2} & T_{l3}(-\frac{\sqrt{3}r\cos(\theta)}{2} - \frac{r\sin(\theta)}{2}) + \frac{lT_{r3}}{2} \\ -\frac{T_{r3}}{2} & T_{l3}(-\frac{\sqrt{3}r\sin(\theta)}{2} + \frac{r\cos(\theta)}{2}) + \frac{\sqrt{3}lT_{r3}}{2} \\ T_{l3} & \frac{T_{r3}}{2}(-\frac{\sqrt{3}r\sin(\theta)}{2} + \frac{r\cos(\theta)}{2}) + \frac{T_{r3}\sqrt{3}}{2}(\frac{\sqrt{3}r\cos(\theta)}{2} - \frac{r\sin(\theta)}{2}) \end{bmatrix}
$$

A.2 Chapter-3

The files for creating the tensegrity mechanisms using SIROPA library of Maple are provided in below links as the equation sizes are large.

- Tilt & Torsion: 3-SPS-U tensegrity mechanism: Click here
- Euler angles: 3-SPS-U tensegrity mechanism: Click here
- Euler angles: 4-SPS-U tensegrity mechanism: Click here

The SIROPA library file is available here, which has to be placed in the library folders of the installation directory to use the environment in Maple.

A.3 Chapter-4

The cartesian velocities calculated from the direct kinematics Jacobian matrix for the experiments on 3-SPS-U tensegrity mechanism:

$$
\begin{aligned}
\dot{l}_1 = &-\sqrt{2} r_f (\dot{\eta}\, h^2 \sin(\eta)\cos(\phi) + \dot{\phi}\, h^2 \cos(\eta)\sin(\phi) + \dot{\phi}\, h \cos(\phi)\cos(\eta) - \dot{\eta}\, \sin(\phi)\sin(\eta)\, h \\
&+ \dot{\phi}\, \cos(\phi)\, h - \dot{\phi}\, \sin(\phi))(2(\cos(\phi)\cos(\eta)\, h^2 - \sin(\phi)\cos(\eta)\, h - \sin(\phi)\, h + h^2 - \cos(\phi) + 1)^{1/2})^{-1}
\end{aligned}
$$

$$
\begin{aligned}
\dot{l}_2 = &\sqrt{2} r_f (-4\,\dot{\eta}\, h^2 \sin(\eta)\cos(\phi) + 2\,\dot{\eta}\,\sqrt{3}\cos(\eta)\cos(\phi)\, h - 2\,\dot{\phi}\,\sqrt{3}\sin(\eta)\sin(\phi)\, h - 4\,\dot{\phi}\, h^2 \cos(\eta) \\
&\sin(\phi) + \dot{\phi}\, \cos(\phi)\sin(\eta)\,\sqrt{3} + 2\,\dot{\phi}\, h \cos(\phi)\cos(\eta) - 2\,\dot{\eta}\, \sin(\phi)\sin(\eta)\, h + \dot{\eta}\,\sqrt{3}\cos(\eta)\sin(\phi) + \\
&2\,\dot{\eta}\, h \cos(\eta)\,\sqrt{3} + 2\,\dot{\phi}\, \cos(\phi)\, h + \dot{\phi}\, \sin(\phi) + 3\,\dot{\eta}\, \sin(\eta))(4(2\,\sqrt{3}\sin(\eta)\cos(\phi)\, h + 4\, \cos(\phi)\cos(\eta)\, h^2 \\
&+ \sqrt{3}\sin(\eta)\sin(\phi) + 2\, \sin(\phi)\cos(\eta)\, h + 2\, \sin(\eta)\, h\sqrt{3} + 2\, \sin(\phi)\, h + 4\, h^2 - \cos(\phi) - 3\, \cos(\eta) \\
&+ 4)^{1/2})^{-1}
\end{aligned}
$$

$$\dot{l}_3 = -\sqrt{2}r_f(4\,\dot{\eta}\,h^2\sin(\eta)\cos(\phi) + 2\,\dot{\eta}\,\sqrt{3}\cos(\eta)\cos(\phi)\,h - 2\,\dot{\phi}\,\sqrt{3}\sin(\eta)\sin(\phi)\,h + 4\,\dot{\phi}\,h^2\cos(\eta)\sin(\phi) + \dot{\phi}\,\cos(\phi)\sin(\eta)\,\sqrt{3} - 2\,\dot{\phi}\,h\cos(\phi)\cos(\eta) + 2\,\dot{\eta}\,\sin(\phi)\sin(\eta)\,h + \dot{\eta}\,\sqrt{3}\cos(\eta)\sin(\phi) + 2\,\dot{\eta}\,h\cos(\eta)\,\sqrt{3} - 2\,\dot{\phi}\,\cos(\phi)\,h - \dot{\phi}\,\sin(\phi) - 3\,\dot{\eta}\,\sin(\eta))(4(-2\,\sqrt{3}\sin(\eta)\cos(\phi)\,h + 4\,\cos(\phi)\cos(\eta)\,h^2 - \sqrt{3}\sin(\eta)\sin(\phi) + 2\,\sin(\phi)\cos(\eta)\,h - 2\,\sin(\eta)\,h\sqrt{3} + 2\,\sin(\phi)\,h + 4\,h^2 - \cos(\phi) - 3\,\cos(\eta) + 4)^{1/2})^{-1}$$

The cartesian accelerations calculated from the direct kinematics Jacobian matrix and its time derivative for the experiments on 3-S<u>P</u>S-U tensegrity mechanism:

$$\ddot{l}_1 = -\sqrt{2}r_f(-2\,\ddot{\phi}\,\sin(\phi) + 2\,\ddot{\phi}\,h - 2\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^2 + 2\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^4 - 2\,\cos(\phi)\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h^3 + 2\,\cos(\phi)\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h + 2\,\cos(\phi)\sin(\phi)\,(\cos(\eta))^2\,\ddot{\phi}\,h^4 - 2\,\cos(\phi)\sin(\phi)\,(\cos(\eta))^2\,\ddot{\phi}\,h^2 - 8\,\cos(\phi)\sin(\phi)\cos(\eta)\,\ddot{\phi}\,h^2 - 2\,(\cos(\eta))^2\,\ddot{\phi}\,h^3 + 2\,\cos(\phi)\ddot{\phi}\,h^3 + 2\,\ddot{\phi}\,\cos(\phi)\,h + 2\,\sin(\eta)\,\ddot{\eta}\,h^2 + 2\,\cos(\phi)\sin(\phi)\,\ddot{\phi} - 2\,\sin(\phi)\,\ddot{\phi}\,h^2 + 2\,\cos(\eta)\,\ddot{\phi}\,h - 4\,(\cos(\phi))^2\ddot{\phi}\,h - 2\,\cos(\eta)\,\ddot{\phi}\,h^3 - 4\,\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^3 + 2\,\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^2 + 4\,(\cos(\phi))^2(\cos(\eta))^2\,\ddot{\phi}\,h^3 + 4\,(\cos(\phi))^2\cos(\eta)\,\ddot{\phi}\,h^3 + 2\,\sin(\phi)\cos(\eta)\,\ddot{\phi}\,h^4 + 2\,\cos(\phi)\cos(\eta)\,\ddot{\phi}\,h^3 - 4\,(\cos(\phi))^2\cos(\eta)\,\ddot{\phi}\,h - 2\,\cos(\phi)\sin(\phi)\,\ddot{\phi}\,h^2 + 2\,\ddot{\phi}\,h^2\cos(\eta)\sin(\phi) + 2\,\ddot{\phi}\,h\cos(\phi)\cos(\eta) + 2\,\cos(\phi)\sin(\eta)\,\ddot{\eta}\,h^4 - 4\,(\cos(\phi))^2\sin(\eta)\,\ddot{\eta}\,h^2 - 2\,\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h^3 + 2\,\ddot{\eta}\,h^2\sin(\eta)\cos(\phi) - 2\,\ddot{\eta}\,\sin(\phi)\sin(\eta)\,h + 2\,\cos(\eta)\,\dot{\eta}\,h^2 - 2\,\sin(\phi)\,\dot{\phi}\,h^3 + 2\,(\cos(\eta))^2\,\dot{\phi}\,h^2 - \dot{\phi}\,(\cos(\phi))^2\,h^2 + \dot{\eta}\,h^4\,(\cos(\phi))^2 + (\cos(\eta))^2\,\dot{\eta}\,h^2 - 2\,\dot{\phi}\,\sin(\phi)\,h + \dot{\phi}\,h^4\,(\cos(\eta))^2 - (\cos(\phi))^2\,\dot{\eta}\,h^2 - 2\,\cos(\phi)\,\dot{\phi}\,h^2 + (\cos(\phi))^2\,\dot{\phi} - 2\,\cos(\phi)\sin(\phi)\,(\cos(\eta))^2\,\dot{\eta}\,h^3 - 2\,\cos(\phi)\sin(\phi)\cos(\eta)\,\dot{\eta}\,h^3 + 2\,\cos(\phi)\sin(\phi)\cos(\eta)\,\dot{\eta}\,h - 2\,\dot{\phi}\,h^3\,(\cos(\eta))^2\sin(\phi)\cos(\phi) - 2\,\dot{\phi}\,h^3\cos(\eta)\sin(\phi)\cos(\phi) + 2\,\dot{\phi}\,h\cos(\phi)\cos(\eta)\sin(\phi) + (\cos(\phi))^2\,(\cos(\eta))^2\,\dot{\eta}\,h^4 - (\cos(\phi))^2\,(\cos(\eta))^2\,\dot{\eta}\,h^2 + 2\,\cos(\phi)\cos(\eta)\,\dot{\eta}\,h^4 - 4\,(\cos(\phi))^2\cos(\eta)\,\dot{\eta}\,h^2 - 2\,\sin(\phi)\cos(\eta)\,\dot{\eta}\,h^3 + 2\,\dot{\eta}\,\cos(\phi)\cos(\eta)\,h^2 - 2\,\dot{\eta}\,\sin(\phi)\cos(\eta)\,h - 2\,\dot{\eta}\,h^3\cos(\phi)\sin(\phi) + (\cos(\phi))^2\,(\cos(\eta))^2\,\dot{\phi}\,h^4 - \dot{\phi}\,h^2\,(\cos(\phi))^2\,(\cos(\eta))^2 + 2\,\cos(\phi)\cos(\eta)\,\dot{\phi}\,h^4 - 4\,\dot{\phi}\,h^2\,(\cos(\phi))^2\cos(\eta) - 2\,\sin(\phi)\cos(\eta)\,\dot{\phi}\,h^3 + 2\,\dot{\phi}\,\cos(\phi)\cos(\eta)\,h^2 + 2\,\dot{\phi}\,\cos(\phi)\,h\sin(\phi) - 2\,\dot{\phi}\,\sin(\phi)\cos(\eta)\,h + 2\,\dot{\phi}\,h^2\cos(\eta) + \dot{\eta}\,h^2 + 2\,\dot{\phi}\,h^2 - 2\,\dot{\phi}\,\cos(\phi) + \dot{\phi})(4(\cos(\phi)\cos(\eta)\,h^2 - \sin(\phi)\cos(\eta)\,h - \sin(\phi)\,h + h^2 - \cos(\phi) + 1)(\cos(\phi)\cos(\eta)\,h^2 - \sin(\phi)\cos(\eta)\,h - \sin(\phi)\,h + h^2 - \cos(\phi) + 1)^{1/2})^{-1}$$

$$\ddot{l}_2 = -\sqrt{2}r_f(16\,\cos(\phi)\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h^3 + 16\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h^3 - 4\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h - 12\,\cos(\phi)\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h + 8\,\sin(\phi)\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h^2 + 8\,\cos(\phi)\sin(\phi)\sin(\eta)\,\sqrt{3}\dot{\eta}\,h^2 + 8\,\cos(\phi)\sin(\phi)\sin(\eta)\,\sqrt{3}\dot{\phi}\,h^2 - 4\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\phi}\,h + 16\,\cos(\phi)\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\phi}\,h^3 + 16\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\phi}\,h^3 - 12\,\cos(\phi)\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\phi}\,h$$

$$+ 8 \sin(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi} h^2 - 32 \cos(\phi) \sin(\phi) (\cos(\eta))^2 \sqrt{3} \dot{\eta} h^2 - 24 \cos(\phi) \sin(\phi) \sin(\eta)$$
$$\cos(\eta) \ddot{\eta} h - 8 \cos(\phi) \sin(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} h^2 + 16 \sin(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \ddot{\phi} h^3 - 32 (\cos(\phi))^2 \sin(\eta)$$
$$\cos(\eta) \sqrt{3} \ddot{\phi} h^2 - 12 \sin(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \dddot{\phi} h - 16 \cos(\phi) \sin(\phi) \sin(\eta) \sqrt{3} \ddot{\phi} h - 8 \cos(\phi)$$
$$\sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi}^2 h^2 - 8 \sqrt{3} \ddot{\eta} h - 6 \sin(\phi) \sqrt{3} \dot{\eta} - 2 \sin(\eta) \sqrt{3} \dddot{\phi} - 12 (\cos(\eta))^2 \ddot{\phi} h + 24 \cos(\phi) \dot{\eta} h^2$$
$$+ 12 \sin(\phi) \dot{\eta} h - 24 \sin(\eta) \ddot{\eta} + 6 \sin(\phi) \cos(\eta) \ddot{\phi} + 6 \sin(\eta) \cos(\phi) \dddot{\eta} + 12 \sin(\eta) \cos(\eta) \ddot{\eta}$$
$$+ 8 (\cos(\phi))^2 (\cos(\eta))^2 \sqrt{3} \ddot{\eta} h + 8 (\cos(\phi))^2 \cos(\eta) \sqrt{3} \ddot{\eta} h - 96 \cos(\phi) \sin(\eta) \cos(\eta) \ddot{\eta} h^2 - 48$$
$$\sin(\phi) \sin(\eta) \cos(\eta) \ddot{\eta} h + 16 \sin(\phi) \sin(\eta) \sqrt{3} \ddot{\phi} h^3 - 16 (\cos(\phi))^2 \sin(\eta) \sqrt{3} \ddot{\phi} h^2 + 12 \sin(\phi)$$
$$\sin(\eta) \sqrt{3} \ddot{\phi} h - 16 \cos(\phi) \sin(\eta) \sqrt{3} \dot{\phi} h^2 + 6 \cos(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \ddot{\phi} + 16 \sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi} h^2 +$$
$$16 \cos(\phi) \sin(\eta) \sqrt{3} \dot{\eta} h^3 - 8 \sin(\eta) \cos(\eta) \sqrt{3} \dot{\eta} h + 12 \cos(\phi) \sin(\eta) \sqrt{3} \dot{\eta} h + 16 \sin(\phi) \sin(\eta) \sqrt{3} \dot{\eta}$$
$$h^2 - 6 \sin(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \dot{\eta} - 2 \cos(\phi) \sin(\phi) \sin(\eta) \sqrt{3} \dot{\eta} - 8 (\cos(\phi))^2 \sin(\eta) \sqrt{3} \dot{\eta} h - 12$$
$$\cos(\phi) \sin(\phi) (\cos(\eta))^2 \dot{\eta} h + 16 \cos(\phi) \sin(\eta) \sqrt{3} \dot{\phi} h^3 + 12 \cos(\phi) \sin(\eta) \sqrt{3} \dot{\phi} h + 16 \sin(\phi) \sin(\eta)$$
$$\sqrt{3} \dot{\phi} h^2 - 6 \sin(\phi) \sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi} - 2 \cos(\phi) \sin(\phi) \sin(\eta) \sqrt{3} \dot{\phi} + 8 \sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi} h +$$
$$16 \sin(\eta) \cos(\eta) \sqrt{3} \dot{\phi} h^3 - 8 (\cos(\phi))^2 \sin(\eta) \sqrt{3} \dot{\phi} h - 12 \cos(\phi) \sin(\phi) (\cos(\eta))^2 \dot{\phi} h - 16 \sin(\phi)$$
$$(\cos(\eta))^2 \sqrt{3} \ddot{\eta} h^2 - 16 \cos(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} h^3 - 32 \cos(\phi) (\cos(\eta))^2 \sqrt{3} \ddot{\eta} h^3 - 32 (\cos(\phi))^2 (\cos(\eta))^2$$
$$\sqrt{3} \ddot{\eta} h^3 - 12 \cos(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} h - 16 \sin(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} h^2 + 24 \cos(\phi) (\cos(\eta))^2 \sqrt{3} \ddot{\eta} h + 2$$
$$\cos(\phi) \sin(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} + 16 \cos(\phi) \sin(\phi) \sqrt{3} \ddot{\eta} h^2 - 20 \cos(\eta) \sqrt{3} \ddot{\eta} h - 16 \cos(\eta) \sqrt{3} \ddot{\eta} h^3 + 12$$
$$\sin(\phi) (\cos(\eta))^2 \sqrt{3} \ddot{\eta} - 12 \cos(\phi) \sqrt{3} \dddot{\eta} h + 16 \cos(\phi) \sqrt{3} \ddot{\eta} h^3 + 8 \sin(\phi) \sqrt{3} \ddot{\eta} h^2 + 16 (\cos(\phi))^2$$
$$\sqrt{3} \ddot{\eta} h^3 - 4 (\cos(\phi))^2 \sqrt{3} \dddot{\eta} h + 16 (\cos(\eta))^2 \sqrt{3} \ddot{\eta} h - 8 \sin(\phi) \cos(\eta) \sqrt{3} \ddot{\eta} + 4 (\cos(\phi))^2 \sin(\eta) \sqrt{3}$$
$$\dddot{\phi} + 8 \sin(\eta) \sqrt{3} \ddot{\phi} h^2 - 8 \cos(\phi) \sin(\eta) \sqrt{3} \dddot{\phi} + 24 (\cos(\phi))^2 (\cos(\eta))^2 \dddot{\phi} h - 48 \sin(\phi) (\cos(\eta))^2 \ddot{\phi} h^2$$
$$+ 24 \cos(\phi) (\cos(\eta))^2 \dddot{\phi} h + 20 \sin(\eta) \sqrt{3} \dot{\eta} h + 16 \sin(\eta) \sqrt{3} \dot{\eta} h^3 + 8 \sin(\phi) \sin(\eta) \sqrt{3} \dot{\eta} +$$
$$24 \cos(\phi) \sin(\phi) \dot{\eta} h - 48 \cos(\phi) (\cos(\eta))^2 \dot{\eta} h^2 - 24 \sin(\phi) (\cos(\eta))^2 \dot{\eta} h + 4 \sin(\eta) \sqrt{3} \dot{\phi} h - 48$$
$$\cos(\phi) (\cos(\eta))^2 \dot{\phi} h^2 - 24 \sin(\phi) (\cos(\eta))^2 \dot{\phi} h + 6 \cos(\phi) \sin(\phi) (\cos(\eta))^2 \ddot{\phi} + 6 (\cos(\phi))^2$$
$$\sin(\eta) \cos(\eta) \ddot{\eta} - 8 \dddot{\phi} \sin(\phi) + 8 \ddot{\phi} h - 32 (\cos(\phi))^2 \sin(\eta) \cos(\eta) \ddot{\eta} h^2 + 32 (\cos(\phi))^2 \sin(\eta)$$
$$\cos(\eta) \ddot{\eta} h^4 + 16 \cos(\phi) \sin(\phi) \sin(\eta) \ddot{\eta} h^3 - 4 \cos(\phi) \sin(\phi) \sin(\eta) \ddot{\eta} h + 32 \cos(\phi) \sin(\phi) (\cos(\eta))^2$$
$$\ddot{\phi} h^4 - 32 \cos(\phi) \sin(\phi) (\cos(\eta))^2 \ddot{\phi} h^2 - 32 \cos(\phi) \sin(\phi) \cos(\eta) \ddot{\phi} h^2 + 16 (\cos(\eta))^2 \ddot{\phi} h^3 - 16$$
$$\cos(\phi) \ddot{\phi} h^3 - 28 \ddot{\phi} \cos(\phi) h - 16 \sin(\eta) \ddot{\eta} h^2 - 4 \cos(\phi) \sin(\phi) \ddot{\phi} + 16 \sin(\phi) \ddot{\phi} h^2 - 4 \cos(\eta) \ddot{\phi} h$$
$$- 16 (\cos(\phi))^2 \ddot{\phi} h + 16 \cos(\eta) \ddot{\phi} h^3 + 3 (\cos(\phi))^2 (\cos(\eta))^2 \dot{\phi} + 6 \cos(\phi) \cos(\eta) \dot{\phi} + 3 (\cos(\phi))^2$$
$$(\cos(\eta))^2 \dot{\eta} + 6 \cos(\phi) \cos(\eta) \dot{\eta} + 32 \cos(\phi) \sin(\phi) \sin(\eta) \cos(\eta) \ddot{\eta} h^3 - 16 \sin(\eta) \cos(\eta) \ddot{\eta} h^2$$
$$- 32 (\cos(\phi))^2 (\cos(\eta))^2 \ddot{\phi} h^3 - 32 (\cos(\phi))^2 \cos(\eta) \ddot{\phi} h^3 + 32 \sin(\phi) \cos(\eta) \ddot{\phi} h^4 - 16 \cos(\phi)$$
$$\cos(\eta) \ddot{\phi} h^3 + 8 (\cos(\phi))^2 \cos(\eta) \ddot{\phi} h + 16 \cos(\phi) \sin(\phi) \ddot{\phi} h^2 + 32 \ddot{\phi} h^2 \cos(\eta) \sin(\phi) - 4 \ddot{\phi} h \cos(\phi)$$
$$\cos(\eta) + 32 \cos(\phi) \sin(\eta) \ddot{\eta} h^4 - 16 (\cos(\phi))^2 \sin(\eta) \ddot{\eta} h^2 + 16 \sin(\phi) \sin(\eta) \ddot{\eta} h^3 + 32 \ddot{\eta} h^2 \sin(\eta)$$
$$\cos(\phi) + 4 \ddot{\eta} \sin(\phi) \sin(\eta) h - 6 (\cos(\phi))^2 \dot{\eta} + 6 (\cos(\eta))^2 \dot{\eta} - 6 (\cos(\eta))^2 \dot{\phi} - 24 \cos(\eta) \dot{\eta} -$$

$$16\cos(\eta)\dot{\eta}h^2+16\sin(\phi)\dot{\phi}h^3-4(\cos(\eta))^2\dot{\phi}h^2+8\dot{\phi}(\cos(\phi))^2h^2+16\dot{\eta}h^4(\cos(\phi))^2-8(\cos(\eta))^2\dot{\eta}h^2+28\dot{\phi}\sin(\phi)h+16\dot{\phi}h^4(\cos(\eta))^2+20(\cos(\phi))^2\dot{\eta}h^2+16\cos(\phi)\dot{\phi}h^2-2(\cos(\phi))^2\dot{\phi}+16\cos(\phi)\sin(\phi)(\cos(\eta))^2\dot{\eta}h^3+16\cos(\phi)\sin(\phi)\cos(\eta)\dot{\eta}h^3-4\cos(\phi)\sin(\phi)\cos(\eta)\dot{\eta}h+16\dot{\phi}h^3(\cos(\eta))^2\sin(\phi)\cos(\phi)+16\dot{\phi}h^3\cos(\eta)\sin(\phi)\cos(\phi)-4\dot{\phi}h\cos(\phi)\cos(\eta)\sin(\phi)+16\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\dot{\eta}h^2+16\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\dot{\phi}h^2+32\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\ddot{\phi}h^3-8\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\ddot{\phi}h+8\dot{\phi}\sqrt{3}\sin(\eta)\sin(\phi)+16(\cos(\phi))^2(\cos(\eta))^2\dot{\eta}h^4-16(\cos(\phi))^2(\cos(\eta))^2\dot{\eta}h^2+32\cos(\phi)\cos(\eta)\dot{\eta}h^4-16(\cos(\phi))^2\cos(\eta)\dot{\eta}h^2+16\sin(\phi)\cos(\eta)\dot{\eta}h^3+32\dot{\eta}\cos(\phi)\cos(\eta)h^2+4\dot{\eta}\sin(\phi)\cos(\eta)h+16\dot{\eta}h^3\cos(\phi)\sin(\phi)+16(\cos(\phi))^2(\cos(\eta))^2\dot{\phi}h^4-16\dot{\phi}h^2(\cos(\phi))^2(\cos(\eta))^2+32\cos(\phi)\cos(\eta)\dot{\phi}h^4-16\dot{\phi}h^2(\cos(\phi))^2\cos(\eta)+16\sin(\phi)\cos(\eta)\dot{\phi}h^3+32\dot{\phi}\cos(\phi)\cos(\eta)h^2+8\dot{\phi}\cos(\phi)h\sin(\phi)+4\dot{\phi}\sin(\phi)\cos(\eta)h+8\dot{\phi}h^2\cos(\eta)+28\dot{\eta}h^2+20\dot{\phi}h^2-8\dot{\phi}\cos(\phi)+15\dot{\eta}+7\dot{\phi})(8(2\sqrt{3}\sin(\eta)\cos(\phi)h+4\cos(\phi)\cos(\eta)h^2+\sqrt{3}\sin(\eta)\sin(\phi)+2\sin(\phi)\cos(\eta)h+2\sin(\eta)h\sqrt{3}+2\sin(\phi)h+4h^2-\cos(\phi)-3\cos(\eta)+4)(2\sqrt{3}\sin(\eta)\cos(\phi)h+4\cos(\phi)\cos(\eta)h^2+\sqrt{3}\sin(\eta)\sin(\phi)+2\sin(\phi)\cos(\eta)h+2\sin(\eta)h\sqrt{3}+2\sin(\phi)h+4h^2-\cos(\phi)-3\cos(\eta)+4)^{1/2})^{-1}$$

$$\ddot{l}_3=-\sqrt{2}r_f(16(\cos(\phi))^2\sin(\eta)\cos(\eta)\sqrt{3}\dot{\eta}h^3-4(\cos(\phi))^2\sin(\eta)\cos(\eta)\sqrt{3}\dot{\eta}h+8\sin(\eta)\sin(\phi)\cos(\eta)\sqrt{3}\dot{\eta}h^2-12\sin(\eta)\cos(\phi)\cos(\eta)\sqrt{3}\dot{\eta}h+16\sin(\eta)\cos(\phi)\cos(\eta)\sqrt{3}\dot{\eta}h^3+8\cos(\phi)\sin(\phi)\sin(\eta)\sqrt{3}\dot{\eta}h^2+16(\cos(\phi))^2\sin(\eta)\cos(\eta)\sqrt{3}\dot{\phi}h^3+8\sin(\eta)\sin(\phi)\cos(\eta)\sqrt{3}\dot{\phi}h^2-12\sin(\eta)\cos(\phi)\cos(\eta)\sqrt{3}\dot{\phi}h+16\sin(\eta)\cos(\phi)\cos(\eta)\sqrt{3}\dot{\phi}h^3-4(\cos(\phi))^2\sin(\eta)\cos(\eta)\sqrt{3}\dot{\phi}h+8\cos(\phi)\sin(\phi)\sin(\eta)\sqrt{3}\dot{\phi}h^2+24\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\ddot{\eta}h-32\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\ddot{\eta}h^3-32\cos(\phi)\sin(\phi)(\cos(\eta))^2\sqrt{3}\ddot{\eta}h^2-8\cos(\phi)\sin(\phi)\cos(\eta)\sqrt{3}\ddot{\eta}h^2-32(\cos(\phi))^2\sin(\eta)\cos(\eta)\sqrt{3}\ddot{\phi}h^2-8\sin(\eta)\cos(\phi)\cos(\eta)\sqrt{3}\ddot{\phi}h^2-16\cos(\phi)\sin(\phi)\sin(\eta)\sqrt{3}\ddot{\phi}h-12\sin(\eta)\sin(\phi)\cos(\eta)\sqrt{3}\ddot{\phi}h+16\sin(\eta)\sin(\phi)\cos(\eta)\sqrt{3}\ddot{\phi}h^3+16\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\dot{\eta}h^2+16\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\dot{\phi}h^2+32\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\ddot{\phi}h^3-8\cos(\phi)\sin(\phi)\sin(\eta)\cos(\eta)\sqrt{3}\ddot{\phi}h-12\sin(\phi)\dot{\eta}h-24\cos(\phi)\dot{\eta}h^2-3(\cos(\phi))^2(\cos(\eta))^2\dot{\eta}-6\cos(\phi)\cos(\eta)\dot{\eta}-16\dot{\eta}h^4(\cos(\phi))^2-20(\cos(\phi))^2\dot{\eta}h^2+8(\cos(\eta))^2\dot{\eta}h^2+16\cos(\eta)\dot{\eta}h^2-3(\cos(\phi))^2(\cos(\eta))^2\dot{\phi}-6\cos(\phi)\cos(\eta)\dot{\phi}-16\dot{\phi}h^4(\cos(\eta))^2+4(\cos(\eta))^2\dot{\phi}h^2-8\dot{\phi}h^2\cos(\eta)-8\dot{\phi}(\cos(\phi))^2h^2-16\sin(\phi)\dot{\phi}h^3-16\cos(\phi)\dot{\phi}h^2-28\dot{\phi}\sin(\phi)h-6\cos(\phi)\sin(\eta)\ddot{\eta}-12\sin(\eta)\cos(\eta)\ddot{\eta}+16\sin(\eta)\ddot{\eta}h^2-6\sin(\phi)\sqrt{3}\ddot{\eta}-8\sqrt{3}\ddot{\eta}h+12(\cos(\eta))^2\ddot{\phi}h-6\sin(\phi)\cos(\eta)\ddot{\phi}-16(\cos(\eta))^2\ddot{\phi}h^3+4\cos(\eta)\ddot{\phi}h-16\cos(\eta)\ddot{\phi}h^3+16\cos(\phi)\ddot{\phi}h^3+16(\cos(\phi))^2\ddot{\phi}h-16\sin(\phi)\ddot{\phi}h^2+4\cos(\phi)\sin(\phi)\ddot{\phi}+28\ddot{\phi}\cos(\phi)h-2\sin(\eta)\sqrt{3}\ddot{\phi}-15\dot{\eta}-7\dot{\phi}+6(\cos(\phi))^2\dot{\eta}-6(\cos(\eta))^2\dot{\eta}+24\cos(\eta)\dot{\eta}-28\dot{\eta}h^2+6(\cos(\eta))^2\dot{\phi}-20\dot{\phi}h^2+2(\cos(\phi))^2\dot{\phi}+8\dot{\phi}\cos(\phi)+24\sin(\eta)\ddot{\eta}-8\ddot{\phi}h+8\ddot{\phi}\sin(\phi)+4\dot{\phi}h\cos(\phi)\cos(\eta)\sin(\phi)+16\sin(\eta)\cos(\phi)\sqrt{3}\dot{\phi}h^3+12\sin(\eta)\cos(\phi))$$

$$-\sqrt{2}r_f(\sqrt{3}\dot{\phi}\,h-6\,\sin(\eta)\sin(\phi)\cos(\eta)\,\sqrt{3}\dot{\phi}-8\,(\cos(\phi))^2\sin(\eta)\,\sqrt{3}\dot{\phi}\,h+16\,\sin(\eta)\sin(\phi)\,\sqrt{3}\dot{\phi}\,h^2\cos(\eta)\,\sqrt{3}\dot{\phi}\,h-2\,\cos(\phi)\sin(\phi)\sin(\eta)\,\sqrt{3}\dot{\phi}+16\,\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\phi}\,h^3+96\,\cos(\phi)\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^2+48\,\sin(\phi)\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h+32\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^2-32\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^4-16\,\cos(\phi)\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h^3+4\,\cos(\phi)\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h-16\,\cos(\eta)\cos(\phi)\,\sqrt{3}\ddot{\eta}\,h^3-16\,\sin(\phi)(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}\,h^2+8\,(\cos(\phi))^2(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}\,h-12\,\cos(\eta)\cos(\phi)\,\sqrt{3}\ddot{\eta}\,h-32\,(\cos(\phi))^2(\cos(\eta))^2\sqrt{3}\ddot{\eta}\,h^3-32\,\cos(\phi)(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}\,h^3+8\,(\cos(\phi))^2\cos(\eta)\,\sqrt{3}\ddot{\eta}\,h+24\,\cos(\phi)(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}\,h-16\,\cos(\eta)\sin(\phi)\,\sqrt{3}\ddot{\eta}\,h^2+2\,\cos(\phi)\sin(\phi)\cos(\eta)\,\sqrt{3}\ddot{\eta}+16\,\cos(\phi)\sin(\phi)\,\sqrt{3}\ddot{\eta}\,h^2-32\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\ddot{\phi}\,h^4+32\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\ddot{\phi}\,h^2+32\,\cos(\phi)\sin(\phi)\cos(\eta)\,\ddot{\phi}\,h^2+16\,\sin(\eta)\sin(\phi)\,\sqrt{3}\ddot{\phi}\,h^3+12\,\sin(\eta)\sin(\phi)\,\sqrt{3}\ddot{\phi}\,h-16\,(\cos(\phi))^2\sin(\eta)\,\sqrt{3}\ddot{\phi}\,h^2-16\,\sin(\eta)\cos(\phi)\,\sqrt{3}\ddot{\phi}\,h^2+6\,\sin(\eta)\cos(\phi)\cos(\eta)\,\sqrt{3}\ddot{\phi}+16\,\sin(\eta)\cos(\eta)\,\sqrt{3}\ddot{\phi}\,h^2+12\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\dot{\eta}\,h-16\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\dot{\eta}\,h^3-16\,\cos(\phi)\sin(\phi)\cos(\eta)\,\dot{\eta}\,h^3+4\,\cos(\phi)\sin(\phi)\cos(\eta)\,\dot{\eta}\,h+12\,\sin(\eta)\cos(\phi)\sqrt{3}\dot{\eta}\,h-8\,\sin(\eta)\cos(\eta)\,\sqrt{3}\dot{\eta}\,h+16\,\sin(\eta)\cos(\phi)\,\sqrt{3}\dot{\eta}\,h^3-6\,\sin(\eta)\sin(\phi)\cos(\eta)\,\sqrt{3}\dot{\eta}-8\,(\cos(\phi))^2\sin(\eta)\,\sqrt{3}\dot{\eta}\,h+16\,\sin(\eta)\sin(\phi)\,\sqrt{3}\dot{\eta}\,h^2-2\,\cos(\phi)\sin(\phi)\sin(\eta)\,\sqrt{3}\dot{\eta}+12\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\dot{\phi}\,h-16\,\dot{\phi}\,h^3(\cos(\eta))^2\sin(\phi)\cos(\phi)-16\,\dot{\phi}\,h^3\cos(\eta)\sin(\phi)\cos(\phi)-6\,\cos(\phi)\sin(\phi)(\cos(\eta))^2\,\ddot{\phi}+48\,\sin(\phi)(\cos(\eta))^2\,\ddot{\phi}\,h^2-24\,\cos(\phi)(\cos(\eta))^2\,\ddot{\phi}\,h-32\,\sin(\phi)\cos(\eta)\,\ddot{\phi}\,h^4+16\,\cos(\phi)\cos(\eta)\,\ddot{\phi}\,h^3-8\,(\cos(\phi))^2\cos(\eta)\,\ddot{\phi}\,h-16\,\cos(\phi)\sin(\phi)\,\ddot{\phi}\,h^2-32\,\ddot{\phi}\,h^2\cos(\eta)\sin(\phi)+4\,\ddot{\phi}\,h\cos(\phi)\cos(\eta)+32\,(\cos(\phi))^2(\cos(\eta))^2\,\ddot{\phi}\,h^3+32\,(\cos(\phi))^2\cos(\eta)\,\ddot{\phi}\,h^3+8\,\sin(\eta)\,\sqrt{3}\ddot{\phi}\,h^2+4\,(\cos(\phi))^2\sin(\eta)\,\sqrt{3}\ddot{\phi}-8\,\sin(\eta)\cos(\phi)\,\sqrt{3}\ddot{\phi}-24\,\cos(\phi)\sin(\phi)\,\dot{\eta}\,h+48\,\cos(\phi)(\cos(\eta))^2\,\dot{\eta}\,h^2+24\,\sin(\phi)(\cos(\eta))^2\,\dot{\eta}\,h+16\,(\cos(\phi))^2(\cos(\eta))^2\,\dot{\eta}\,h^2-16\,\dot{\eta}\,h^3\cos(\phi)\sin(\phi)+16\,(\cos(\phi))^2\cos(\eta)\,\dot{\eta}\,h^2-16\,\sin(\phi)\cos(\eta)\,\dot{\eta}\,h^3-32\,\dot{\eta}\,\cos(\phi)\cos(\eta)\,h^2-4\,\dot{\eta}\,\sin(\phi)\cos(\eta)\,h-16\,(\cos(\phi))^2(\cos(\eta))^2\,\dot{\eta}\,h^4-32\,\cos(\phi)\cos(\eta)\,\dot{\eta}\,h^4+16\,\sin(\eta)\,\sqrt{3}\dot{\eta}\,h^3+20\,\sin(\eta)\,\sqrt{3}\dot{\eta}\,h+8\,\sin(\eta)\sin(\phi)\,\sqrt{3}\dot{\eta}+48\,\cos(\phi)(\cos(\eta))^2\,\dot{\phi}\,h^2+24\,\sin(\phi)(\cos(\eta))^2\,\dot{\phi}\,h+16\,\dot{\phi}\,h^2(\cos(\phi))^2\cos(\eta)-16\,\sin(\phi)\cos(\eta)\,\dot{\phi}\,h^3-32\,\dot{\phi}\,\cos(\phi)\cos(\eta)\,h^2-8\,\dot{\phi}\,\cos(\phi)\sin(\phi)\,h-4\,\dot{\phi}\,\sin(\phi)\cos(\eta)\,h-16\,(\cos(\phi))^2(\cos(\eta))^2\,\dot{\phi}\,h^4+16\,\dot{\phi}\,(\cos(\eta))^2(\cos(\phi))^2\,h^2-32\,\cos(\phi)\cos(\eta)\,\dot{\phi}\,h^4+4\,\sin(\eta)\,\sqrt{3}\dot{\phi}\,h+8\,\sin(\eta)\sin(\phi)\,\sqrt{3}\dot{\phi}-6\,(\cos(\phi))^2\sin(\eta)\cos(\eta)\,\ddot{\eta}+16\,\sin(\eta)\cos(\eta)\,\ddot{\eta}\,h^2+16\,(\cos(\phi))^2\sin(\eta)\,\ddot{\eta}\,h^2-16\,\sin(\phi)\sin(\eta)\,\ddot{\eta}\,h^3-32\,\ddot{\eta}\,h^2\sin(\eta)\cos(\phi)-4\,\ddot{\eta}\,\sin(\phi)\sin(\eta)\,h-32\,\cos(\phi)\sin(\eta)\,\ddot{\eta}\,h^4-16\,\cos(\eta)\,\sqrt{3}\ddot{\eta}\,h^3+16\,(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}\,h-20\,\cos(\eta)\,\sqrt{3}\ddot{\eta}\,h+16\,\cos(\phi)\,\sqrt{3}\ddot{\eta}\,h^3+16\,(\cos(\phi))^2\,\sqrt{3}\ddot{\eta}\,h^3-4\,(\cos(\phi))^2\,\sqrt{3}\ddot{\eta}\,h+8\,\sin(\phi)\,\sqrt{3}\ddot{\eta}\,h^2+12\,\sin(\phi)(\cos(\eta))^2\,\sqrt{3}\ddot{\eta}-12\,\cos(\phi)\,\sqrt{3}\ddot{\eta}\,h-8\,\cos(\eta)\sin(\phi)\,\sqrt{3}\ddot{\eta}-24\,(\cos(\phi))^2(\cos(\eta))^2\,\ddot{\phi}\,h)(8((2\,\sin(\eta)\cos(\phi)\,\sqrt{3}h-4\,\cos(\phi)\cos(\eta)\,h^2+\sin(\eta)\sin(\phi)\,\sqrt{3}-2\,\sin(\phi)\cos(\eta)\,h+2\,h\sin(\eta)\,\sqrt{3}-2\,\sin(\phi)\,h-4\,h^2+\cos(\phi)+3\,\cos(\eta)-4)(2\,\sin(\eta)\cos(\phi)\,\sqrt{3}h-4\,\cos(\phi)\cos(\eta)\,h^2+\sin(\eta)\sin(\phi)\,\sqrt{3}-2\,\sin(\phi)\cos(\eta)\,h+2\,h\sin(\eta)\,\sqrt{3})^{1/2}))^{-1}$$

A.4 Chapter-5

The construction of rectangular cross-section in MATLAB represented below in Figure A.2 is created as per equations provided below the figure.

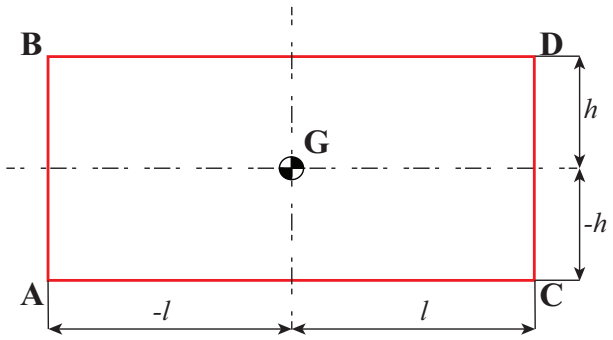


Figure A.2 – Construction of rectangle in MATLAB

For given coordinates of centroid **G**, the coordinates of the vertices are given by:

$$\mathbf{A} = \mathbf{G} + \mathbf{T}\begin{bmatrix} -l \\ -h \end{bmatrix}\ ,\ \mathbf{B} = \mathbf{G} + \mathbf{T}\begin{bmatrix} -l \\ h \end{bmatrix}\ ,\ \mathbf{C} = \mathbf{G} + \mathbf{T}\begin{bmatrix} l \\ -h \end{bmatrix}\ ,\ \mathbf{D} = \mathbf{G} + \mathbf{T}\begin{bmatrix} l \\ h \end{bmatrix}$$

$$\text{where } \mathbf{T} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

T is the rotation matrix which is used to rotate the rectangle about **G** through an angle θ.

Appendix B: Supplementary figures

B.1 Chapter-2

The plot of tangential longitudinal forces for the four length values of Table 2.5 is provided below in Figure B.1.

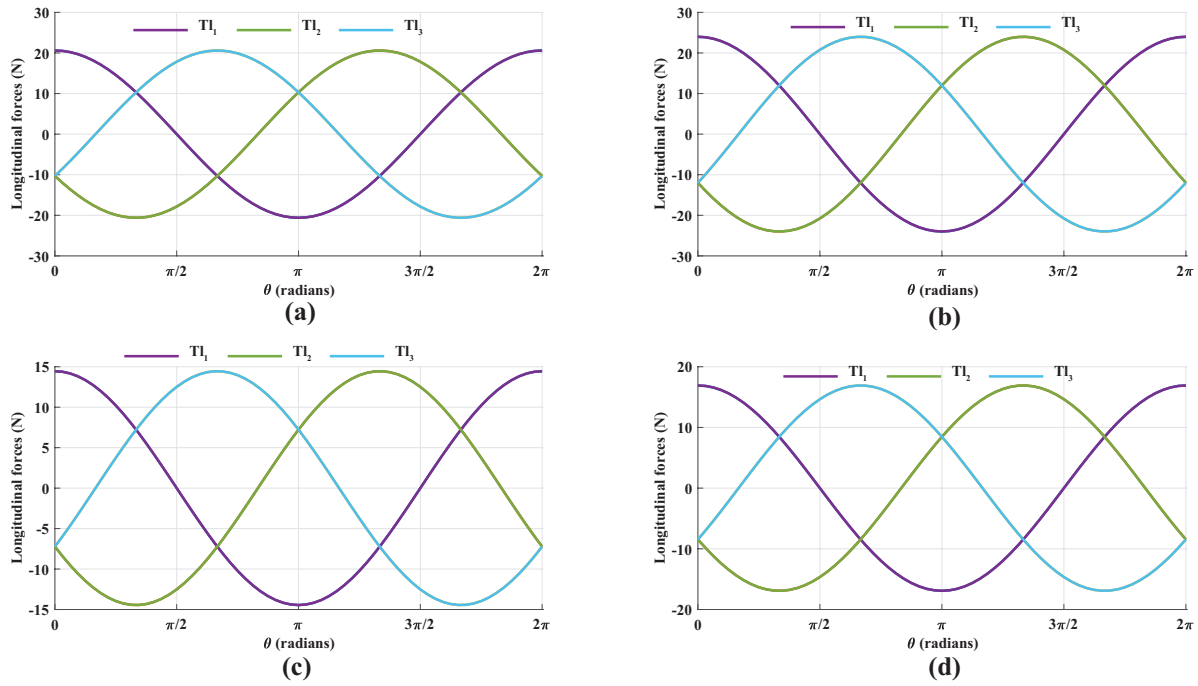


Figure B.1 – Plot of tangential longitudinal forces for the length values (ascending) of Table 2.5 in the sequence (a) to (d)

B.2 Chapter-4

The plots of theoretical angular positions, velocities and accelerations of the input tilt angles employed for performing two springs pull and circular trajectory experiments on the 3-SPS-U tensegrity mechanism are provided in Figures B.2 and B.3.

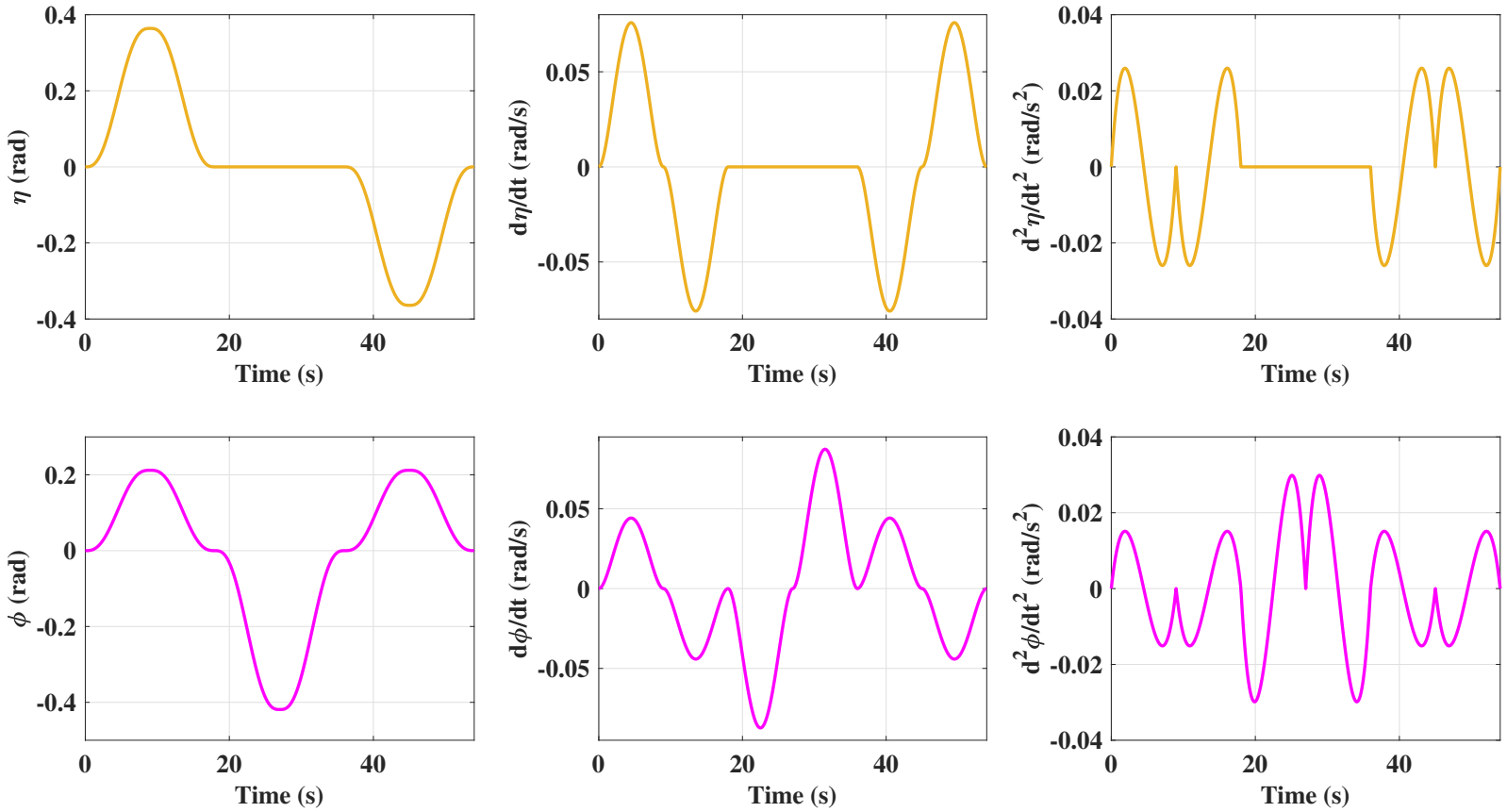


Figure B.2 – Theoretical positions, velocities and accelerations of the input tilt angles η (orange) and ϕ (magenta) for the two springs pull experiment

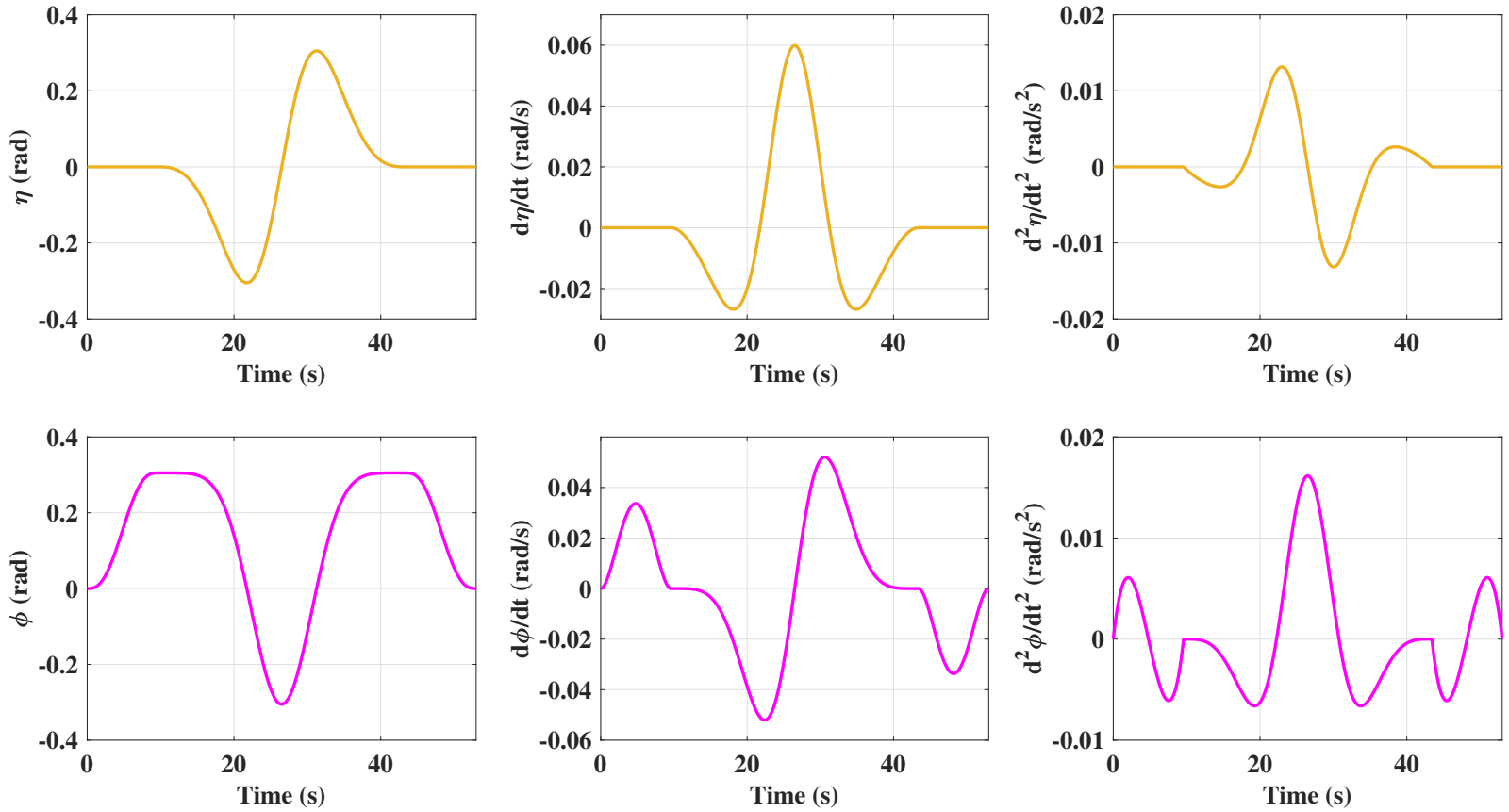


Figure B.3 – Theoretical positions, velocities and accelerations of the input tilt angles η (orange) and ϕ (magenta) for the circular trajectory experiment

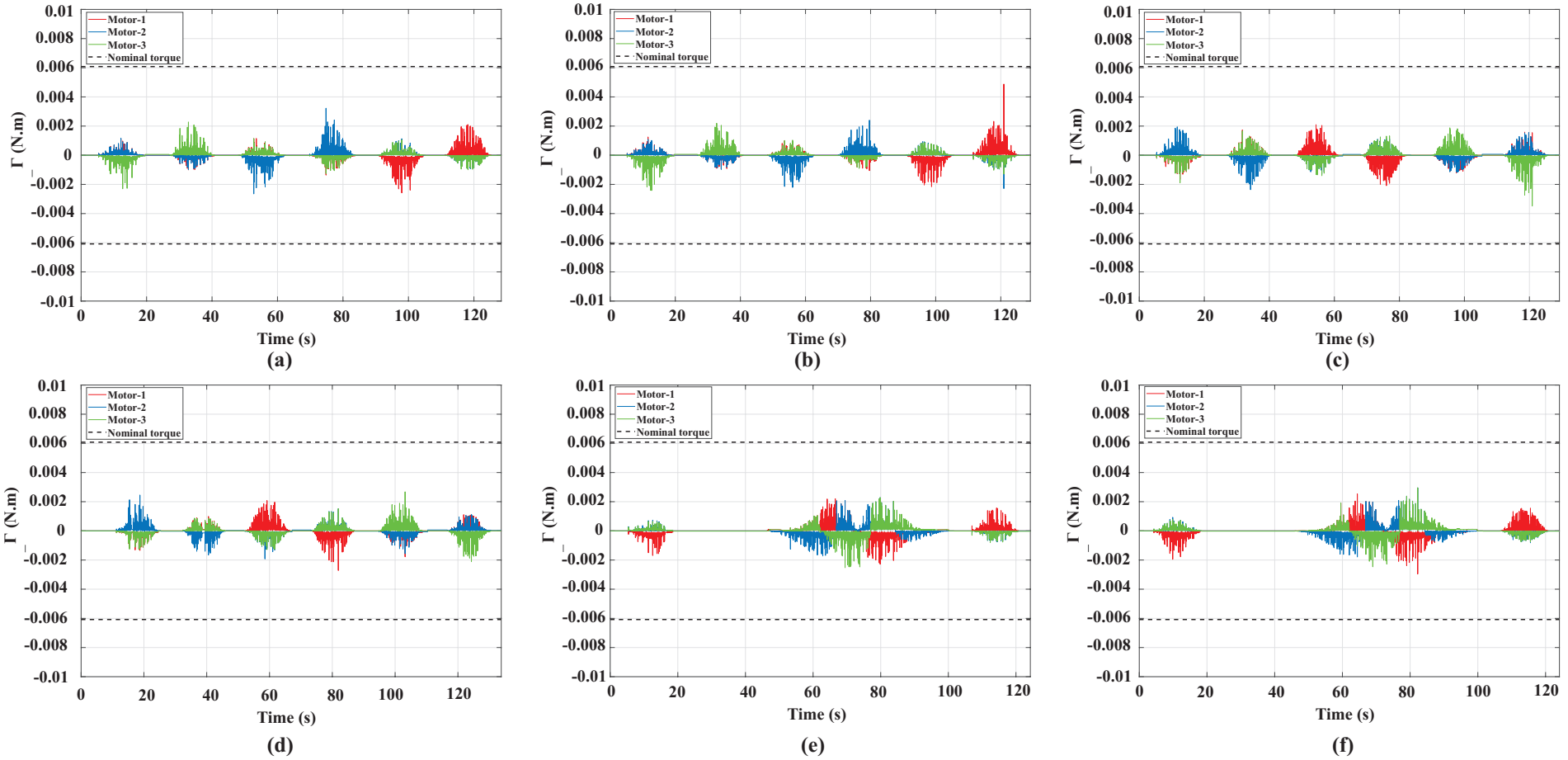


Figure B.4 – Motor torques on each motor after Savitzky-Golay filtering for (a) vertical one spring, (b) horizontal one spring, (c) vertical two springs, (d) horizontal two springs, (e) vertical circular and (f) horizontal circular experiments

B.3 Chapter-5

Flowchart for the first optimization problem algorithm written in MATLAB is provided below in Figure B.5.

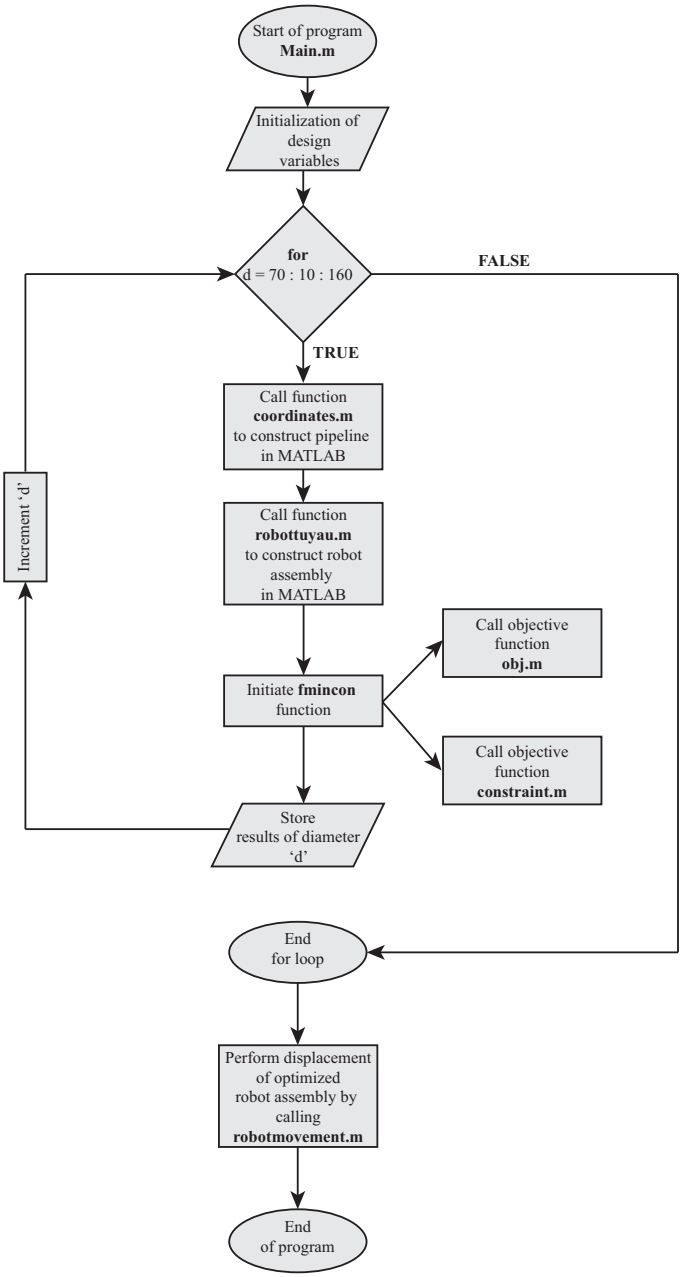


Figure B.5 – Flowchart for the first optimization problem in MATLAB

The flowchart for the second optimization problem is similar to Figure B.5, however, the design variables are different and the for loop is not employed as the diameter of pipeline is fixed at 100 mm.

Bibliography

[ACB99] Cédric Anthierens, Antoine Ciftci, and Maurice Betemps. Design of an electro pneumatic micro robot for in-pipe inspection. In *Industrial Electronics, 1999. ISIE'99. Proceedings of the IEEE International Symposium on*, volume 2, pages 968–972. IEEE, 1999.

[AG06] Marc Arsenault and Clément M Gosselin. Kinematic, static and dynamic analysis of a planar 2-dof tensegrity mechanism. *Mechanism and Machine Theory*, 41(9):1072–1089, 2006.

[ALT+00] Cédric Anthierens, Christine Libersa, Mohamed Touaibia, Maurice Bétemps, Marc Arsicault, and Nicolas Chaillet. Micro robots dedicated to small diameter canalization exploration. In *Intelligent Robots and Systems, 2000.(IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, volume 1, pages 480–485. IEEE, 2000.

[AMM+18] Mohamed Abdellatif, Hazem Mohamed, Hesham Mohamed, Abdelmoneim Ahmed, Amro Kamal, and Khaled Ahmed. Mechatronics design of an autonomous pipe-inspection robot. In *MATEC Web of Conferences*, volume 153. EDP Sciences, 2018.

[AS04] Gürsel Alici and Bijan Shirinzadeh. Topology optimisation and singularity analysis of a 3-sps parallel manipulator with a passive constraining spherical joint. *Mechanism and Machine Theory*, 39(2):215–235, 2004.

[BAB+08] Sébastien Briot, Vigen Arakelian, Ilian A Bonev, Damien Chablat, and Philippe Wenger. Self-motions of general 3-r pr planar parallel robots. *The International Journal of Robotics Research*, 27(7):855–866, 2008.

[BAV+17] Quentin Boehler, Salih Abdelaziz, Marc Vedrines, Philippe Poignet, and Pierre Renaud. Towards the control of tensegrity mechanisms for variable stiffness applications: a case study. In *New Trends in Mechanism and Machine Science*, pages 163–171. Springer, 2017.

[BB] Beaglebone black. https://elinux.org/Beagleboard:BeagleBoneBlack. Accessed: 2018-06-29.

[BB20] Fouad Bennis and Rajib Kumar Bhattacharjya. *Nature-Inspired Methods for Metaheuristics Optimization: Algorithms and Applications in Science and Engineering*, volume 16. Springer, 2020.

[BBV04] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[BCVR15] Quentin Boehler, Isabelle Charpentier, Marc S Vedrines, and Pierre Renaud. Definition and computation of tensegrity mechanism workspace. *Journal of Mechanisms and Robotics*, 7(4), 2015.

[BHPC16] Frédéric Boyer, Renaud Henry, Mathieu Porez, and Damien Chablat. Etude de faisabilité d'un robot d'intervention en milieu industriel (nucléaire et renouvable). Institut de Recherche en Communications et Cybernétique de Nantes, 2016.

[BHS+11] Alexander S Boxerbaum, Andrew D Horchler, Kendrick M Shaw, Hillel J Chiel, and Roger D Quinn. A controller for continuous wave peristaltic locomotion. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 197–202. IEEE, 2011.

[BIDM+13] R Bozzi, A Inzirillo, V Di Martino, M Sannino, F Piantadosi, D Cattaneo, U O C Chirurgia, E D Endoscopia, D Azienda, O Dei, and C Napoli. Endotics system , a new way to undergo the endoscopic valuation of lower gi tract : Our initial experience. 2013.

[BMTH15] Daan L Bakker, Daisuke Matsuura, Yukio Takeda, and Just L Herder. Design of an environmentally interactive continuum manipulator. In *Proc. 14th World Congress in Mechanism and Machine Science, IFToMM*, 2015.

[Bon02] Ilian Alexandrov Bonev. *Geometric analysis of parallel mechanisms*. Citeseer, 2002.

[BTT13] Júlia Borras, Federico Thomas, and Carme Torras. New geometric approaches to the analysis and design of stewart–gough platforms. *IEEE/ASME Transactions on Mechatronics*, 19(2):445–455, 2013.

[BZG02] Ilian A Bonev, Dimiter Zlatanov, and Clément M Gosselin. Advantages of the modified euler angles in the design and control of pkms. In *2002 Parallel Kinematic Machines International Conference*, pages 171–188. Citeseer, 2002.

[CCLW15] Stéphane Caro, Damien Chablat, Philippe Lemoine, and Philippe Wenger. Kinematic analysis and trajectory planning of the orthoglide 5-axis. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 57144, page V05CT08A004. American Society of Mechanical Engineers, 2015.

[CF95] Andrew J Chipperfield and Peter J Fleming. The matlab genetic algorithm toolbox. 1995.

[CLO13] David Cox, John Little, and Donal OShea. *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2013.

[CM06] Menon Carlo and Sitti Metin. A biomimetic climbing robot based on the gecko. *Journal of Bionic Engineering*, 3(3):115–125, 2006.

[CMB+20] Damien Chablat, Guillaume Michel, Philippe Bordure, Ranjan Jha, and Swaminath Venkateswaran. Joint space and workspace analysis of a 2-dof spherical parallel mechanism. In *European Conference on Mechanism Science*, pages 181–188. Springer, 2020.

[CMB+21] Damien Chablat, Guillaume Michel, Philippe Bordure, Swaminath Venkateswaran, and Ranjan Jha. Workspace analysis in the design parameter space of a 2-dof spherical parallel mechanism for a prescribed workspace: Application to the otologic surgery. *Mechanism and Machine Theory*, 157:104224, 2021.

[CMRW20] Damien Chablat, Guillaume Moroz, Fabrice Rouillier, and Philippe Wenger. Using maple to analyse parallel robots. In *Maple in Mathematics Education and Research: Third Maple Conference, MC 2019, Waterloo, Ontario, Canada, October 15–17, 2019, Proceedings*, volume 1125, pages 50–64. Springer Nature, 2020.

[CMW11] Damien Chablat, Guillaume Moroz, and Philippe Wenger. Uniqueness domains and non singular assembly mode changing trajectories. In *2011 IEEE International Conference on Robotics and Automation*, pages 3946–3951. IEEE, 2011.

[Col75] George E Collins. Quantifier elimination for real closed fields by cylindrical algebraic decompostion. In *Automata Theory and Formal Languages 2nd GI Conference Kaiserslautern, May 20–23, 1975*, pages 134–183. Springer, 1975.

[COM11] Damien Chablat, Erika Ottaviano, and Guillaume Moroz. A comparative study of 4-cable planar manipulators based on cylindrical algebraic decomposition. In *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 1253–1262. American Society of Mechanical Engineers, 2011.

[COV18] Damien Chablat, Erika Ottaviano, and Swaminath Venkateswaran. Self-motion of the 3-ppps parallel robot with delta-shaped base. In *European Conference on Mechanism Science*, pages 317–324. Springer, 2018.

[COV19] Damien Chablat, Erika Ottaviano, and Swaminath Venkateswaran. Self-motion conditions for a 3-ppps parallel robot with delta-shaped base. *Mechanism and Machine Theory*, 135:109–114, 2019.

[CPJ09] Changhwan Choi, Byungsuk Park, and Seungho Jung. The design and analysis of a feeder pipe inspection robot with an automatic pipe tracking system. *IEEE/ASME Transactions on Mechatronics*, 15(5):736–745, 2009.

[CVB18] Damien Chablat, Swaminath Venkateswaran, and Frédéric Boyer. Mechanical design optimization of a piping inspection robot. *Procedia CIRP*, 70:307–312, 2018.

[CVB19] Damien Chablat, Swaminath Venkateswaran, and Frédéric Boyer. Dynamic model of a bio-inspired robot for piping inspection. In *ROMANSY 22–Robot Design, Dynamics and Control*, pages 42–51. Springer, 2019.

[CW98] Damien Chablat and Philippe Wenger. Working modes and aspects in fully parallel manipulators. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, volume 3, pages 1964–1969. IEEE, 1998.

[Daw03] Paul Dawkins. Paul's online math notes. *URL: http://tutorial. math. lamar. edu*, 2003.

[DS11] Edwin Dertien and Stefano Stramigioli. Basic maneuvers for an inspection robot for small diameter gas distribution mains. In *2011 IEEE International Conference on Robotics and Automation*, pages 3447–3448. IEEE, 2011.

[DSS61] M Drapier, V Steenbrugghe, and B Successeurs. Perfectionnements aux cathéters médicaux. *France Patent*, 1, 1961.

[EBDW14] Matthew England, Russell Bradford, James H Davenport, and David Wilson. Choosing a variable ordering for truth-table invariant cylindrical algebraic decomposition by incremental triangular decomposition. In *International Congress on Mathematical Software*, pages 450–457. Springer, 2014.

[FFK+20] Benjamin Fasquelle, Matthieu Furet, Parag Khanna, Damien Chablat, Christine Chevallereau, and Philippe Wenger. A bio-inspired 3-dof light-weight manipulator with tensegrity x-joints. In *ICRA'2020*, 2020.

[fmi] Matlab, the "fmincon" function. http://fr.mathworks.com/help/optim/ug/fmincon.html. Accessed: 2020-01-12.

[Fon16] Chamberlain Fong. Squircular calculations. *arXiv preprint arXiv:1604.02174*, 2016.

[FW19] Matthieu Furet and Philippe Wenger. Kinetostatic analysis and actuation strategy of a planar tensegrity 2-x manipulator. *Journal of Mechanisms and Robotics*, 11(6), 2019.

[ga] Matlab, the "ga" function. https://fr.mathworks.com/help/gads/ga.html. Accessed: 2020-01-12.

[GA90] Clément Gosselin and Jorge Angeles. Singularity analysis of closed-loop kinematic chains. *IEEE transactions on robotics and automation*, 6(3):281–290, 1990.

[GB09] Sofiane Guessasma and David Bassir. Comparing heuristic and deterministic approaches to optimise mechanical parameters of biopolymer composite materials. *Mechanics of Advanced Materials and Structures*, 16(4):293–299, 2009.

[GCITS88] Irwin Ginsburgh, John A Carlson III, Geoffrey L Taylor, and Hamid Saghatchi. Method and apparatus for fluid propelled borescopes, April 5 1988. US Patent 4,735,501.

[GVS08] Alessandro Gasparetto, Renato Vidoni, and Tobias Seidl. Kinematic study of the spider system in a biomimetic perspective. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3077–3082. IEEE, 2008.

[GXS+10] Bingtuan Gao, Ning Xi, Yantao Shen, Jianguo Zhao, and Ruiguo Yang. Development of a low motion-noise humanoid neck: Statics analysis and experimental validation. In *2010 IEEE International Conference on Robotics and Automation*, pages 1203–1208. IEEE, 2010.

[HCP+14] Renaud Henry, Damien Chablat, Mathieu Porez, Frédéric Boyer, and Daniel Kanaan. Multi-objective design optimization of the leg mechanism for a piping inspection robot. In *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V05AT08A001–V05AT08A001. American Society of Mechanical Engineers, 2014.

[HDMP02] Mihaita Horodinca, Ioan Doroftei, Emmanuel Mignon, and André Preumont. A simple architecture for in-pipe inspection robots. In *Proc. Int. Colloq. Mobile, Autonomous Systems*, pages 61–64, 2002.

[HH16] Desmond J Higham and Nicholas J Higham. *MATLAB guide*. SIAM, 2016.

[HHY+01] JM Hyun, JL Hvung, ML Young, JP Juang, K Byungkyu, and HK Soo. Magnetic impact actuator for robotic endoscope. In *Proc. 32nd Int. Symp. Robotics, Seoul, Korea*, pages 1834–1838, 2001.

[Hib07] Russell Charles Hibbeler. *Engineering mechanics: statics*. Pearson Education India, 2007.

[HM04] Shigeo Hirose and Makoto Mori. Biologically inspired snake-like robots. In *2004 IEEE International Conference on Robotics and Biomimetics*, pages 1–7. IEEE, 2004.

[HOMS99] Shigeo Hirose, Hidetaka Ohno, Takeo Mitsui, and Kiichi Suyama. Design of in-pipe inspection vehicles for ϕ25, ϕ50, ϕ150 pipes. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 3, pages 2309–2314. IEEE, 1999.

[HPB03] Mihaita Horodinca, Andre Preumont, and Ioan Burda. The heli-pipe inspection robots architecture for curved pipes. In *Proceedings of the International Conference of Manufacturing Systems ICMS*, pages 149–154, 2003.

[Hus00] Manfred L Husty. E. borel's and r. bricard's papers on displacements with spherical paths and their relevance to self-motions of parallel manipulators. In *International Symposium on History of Machines and Mechanisms Proceedings HMM 2000*, pages 163–171. Springer, 2000.

[IS97] Gavriel J Iddan and Doron Sturlesi. In vivo video camera system, February 18 1997. US Patent 5,604,531.

[ITH88] Koji Ikuta, Masahiro Tsukamoto, and Shigeo Hirose. Shape memory alloy servo actuator system with electric resistance feedback and application for active endoscope. In *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, pages 427–430. Ieee, 1988.

[JCB+18] Ranjan Jha, Damien Chablat, Luc Baron, Fabrice Rouillier, and Guillaume Moroz. Workspace, joint space and singularities of a family of delta-like robot. *Mechanism and Machine Theory*, 2018.

[JCRM16] Ranjan Jha, Damien Chablat, Fabrice Rouillier, and Guillaume Moroz. Influence of the trajectory planning on the accuracy of the orthoglide 5-axis. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 50152, page V05AT07A039. American Society of Mechanical Engineers, 2016.

[Jha16] Ranjan Jha. *Contribution à l'analyse de performance des robots parallèles*. PhD thesis, 2016.

[Jia19] Wei Jiang. *Analysis and Design of Machine Elements*. John Wiley & Sons, 2019.

[KD04] Wisama Khalil and Etienne Dombre. *Modeling, identification and control of robots*. Butterworth-Heinemann, 2004.

[Kha10] Wisama Khalil. Dynamic modeling of robots using recursive newton-euler techniques. In *ICINCO2010*, 2010.

[Kim13] Joe Kimbrell. Fundamentals of industrial encoder sensing technologies, motion detection theory and methods, and signal output styles. *AutomationDirect. com Inc*, 9, 2013.

[KLJY08] Young-Sik Kwon, Hoon Lim, Eui-Jung Jung, and Byung-Ju Yi. Design and motion planning of a two-moduled indoor pipeline inspection robot. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3998–4004. IEEE, 2008.

[KPN+06] Irwan Kassim, Louis Phee, Wan S Ng, Feng Gong, Paolo Dario, and Charles A Mosse. Locomotion techniques for robotic colonoscopy. *IEEE Engineering in Medicine and Biology Magazine*, 25(3):49–56, 2006.

[KST+07] Sangbae Kim, Matthew Spenko, Salomon Trujillo, Barrett Heyneman, Virgilio Mattoli, and Mark R Cutkosky. Whole body adhesion: hierarchical, directional and distributed control of adhesive forces for a climbing robot. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 1268–1273. IEEE, 2007.

[KYK+95] Yoshifumi Kawaguchi, Itsuo Yoshida, H Kurumatani, Takashi Kikuta, and YAYY Yamada. Internal pipe inspection robot. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 1, pages 857–862. IEEE, 1995.

[LCT14] Fabio Leccese, Marco Cagnetti, and Daniele Trinca. A smart city application: A fully controlled street lighting isle based on raspberry-pi card, a zigbee sensor network and wimax. *Sensors*, 14(12):24408–24424, 2014.

[Lem19] Philippe Lemoine. Commande d'un moteur électrique à courant continu mettant en mouvement une charge inertielle. Le Laboratoire des Sciences du Numérique de Nantes (LS2N), 2019.

[LMS+06] Weiting Liu, Arianna Menciassi, Serio Scapellato, Paolo Dario, and Yuquan Chen. A biomimetic sensor for a crawling minirobot. *Robotics and Autonomous systems*, 54(7):513–528, 2006.

[LO09] Hun-ok Lim and Taku Ohki. Development of pipe inspection robot. In *2009 ICCAS-SICE*, pages 5717–5721. IEEE, 2009.

[LYZG07] H Li, H Yang, M Zhan, and RJ Gu. The interactive effects of wrinkling and other defects in thin-walled tube nc bending process. *Journal of Materials Processing Technology*, 187:502–507, 2007.

[MAH11] Majid M Moghaddam, Mohammadreza Arbabtafti, and Alireza Hadi. In-pipe inspection crawler adaptable to the pipe interior diameter. *International Journal of Robotics and Automation*, 26(2):135, 2011.

[Mas78] S Masuda. Apparatus for feeding a flexible tube through a conduit. *UK patent*, 1, 1978.

[MAT] Choosing the algorithm- MATLAB & Simulink- Mathworks France. https://fr.mathworks.com/help/optim/ug/choosing-the-algorithm.html. Accessed: 2020-02-02.

[Max] Maxon motors,program 2017/18. high precision drives and systems. http://epaper.maxonmotor.com/. Accessed: 2017-12-15.

[MCWR10] Guillaume Moroz, Damien Chablat, Philippe Wenger, and Fabrice Rouiller. Cusp points in the parameter space of rpr-2prr parallel manipulators. In *New Trends in Mechanism Science*, pages 29–37. Springer, 2010.

[Mei10] Leonard Meirovitch. *Fundamentals of vibrations*. Waveland Press, 2010.

[Mer06] Jean-Pierre Merlet. *Parallel robots*, volume 128. Springer Science & Business Media, 2006.

[MLN18] Myo Maung Maung, Maung Maung Latt, and Chaw Myat Nwe. Dc motor angular position control using pid controller with friction compensation. *International Journal of Scientific and Research Publications*, 8(11):149, 2018.

[MMRH11] Hamidreza Marvi, Gregory Meyers, Geoffrey Russell, and David L Hu. Scalybot: a snake-inspired robot with active control of friction. In *Dynamic Systems and Control Conference*, volume 54761, pages 443–450, 2011.

[Neu94] Werner Neubauer. A spider-like robot that climbs vertically in ducts or pipes. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, volume 2, pages 1178–1185. IEEE, 1994.

[NP14] Ankit Nayak and SK Pradhan. Design of a new in-pipe inspection robot. *Procedia Engineering*, 97:2081–2091, 2014.

[NSS+99] Hideaki Nishikawa, Takanari Sasaya, Takayuki Shibata, Takashi Kaneko, Naoki Mitumoto, Shinichirou Kawakita, and Nobuaki Kawahara. In-pipe wireless micro locomotive system. In *MHS'99. Proceedings of 1999 International Symposium on Micromechatronics and Human Science (Cat. No. 99TH8478)*, pages 141–147. IEEE, 1999.

[O'D09] Aidan O'Dwyer. *Handbook of PI and PID controller tuning rules*. Imperial college press, 2009.

[OS87] Tokuji Okada and Tsuyoshi Sanemori. Mogrer: A vehicle study and realization for in-pipe inspection tasks. *IEEE Journal on Robotics and Automation*, 3(6):573–582, 1987.

[Pap99] Georgios A Papadakis. Major hazard pipelines: a comparative study of onshore transmission accidents. *Journal of Loss Prevention in the Process Industries*, 12(1):91–107, 1999.

[PBI14] Mathieu Porez, Frédéric Boyer, and Auke Jan Ijspeert. Improved lighthill fish swimming model for bio-inspired robots: Modeling, computational aspects and experimental comparisons. *The International Journal of Robotics Research*, 33(10):1322–1341, 2014.

[PCL19] Yitao Pan, Yuan Chen, and Lin Li. Analysis of kinematic dexterity and stiffness performance based on spring's wire-driven 4-sps/u rigid–flexible parallel trunk joint mechanism. *International Journal of Structural Integrity*, 2019.

[PD14] Zoran Pandilov and Vladimir Dukovski. Comparison of the characteristics between serial and parallel robots. *Acta Technica Corvininesis-Bulletin of Engineering*, 7(1), 2014.

[PHC+11] Jungwan Park, Dongjun Hyun, Woong-Hee Cho, Tae-Hyun Kim, and Hyun-Seok Yang. Normal-force control for an in-pipe robot according to the inclination of pipelines. *IEEE transactions on Industrial Electronics*, 58(12):5304–5310, 2011.

[PHM] Phmsa: Pipeline incidents report. https://www.phmsa.dot.gov/data-and-statistics/pipeline/pipeline-incident-20-year-trends/. Accessed: 2020-06-15.

[Pop17] Valentin L Popov. Coulomb's law of friction. In *Contact Mechanics and Friction*, pages 151–172. Springer, 2017.

[Pra11] Errik Prada. Elektromagnetickỳ akčnỳ člen pre pohon lokomočnej štruktúry umelého hada. *Diplomová práca. Katedra aplikovanej mechaniky a mechatroniky, Strojnícka fakulta, Technická univerzita v Košiciach*, 2011.

[QWMD07] Marco Quirini, Robert J Webster, Arianna Menciassi, and Paolo Dario. Design of a pill-sized 12-legged endoscopic capsule robot. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 1856–1862. IEEE, 2007.

[RBC12] David Rollinson, Austin Buchan, and Howie Choset. Virtual chassis for snake robots: Definition and applications. *Advanced Robotics*, 26(17):2043–2064, 2012.

[RBR+00] SungMoo Ryew, SH Baik, SW Ryu, Kwang Mok Jung, SG Roh, and Hyouk Ryeol Choi. In-pipe inspection robot system with active steering mechanism. In *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113)*, volume 3, pages 1652–1657. IEEE, 2000.

[RC05] Se-gon Roh and Hyouk Ryeol Choi. Differential-drive in-pipe robot for moving inside urban gas pipelines. *IEEE transactions on robotics*, 21(1):1–17, 2005.

[RRYC01] SG Roh, SungMoo Ryew, JH Yang, and HR Choi. Actively steerable in-pipe inspection robots for underground urban gas pipelines. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, volume 1, pages 761–766. IEEE, 2001.

[RSS01] Elon Rimon, Shraga Shoval, and Amir Shapiro. Design of a quadruped robot for motion with quasistatic force constraints. *Autonomous Robots*, 10(3):279–296, 2001.

[RW17] R Venkata Rao and GG Waghmare. A new optimization algorithm for solving complex constrained design optimization problems. *Engineering Optimization*, 49(1):60–83, 2017.

[SAP+01] Robert E Skelton, Rajesh Adhikari, J-P Pinaud, Waileung Chan, and JW Helton. An introduction to the mechanics of tensegrity structures. In *Proceedings of the 40th IEEE conference on decision and control (Cat. No. 01CH37228)*, volume 5, pages 4254–4259. IEEE, 2001.

[sgo] Savitzky-golay filtering. https://fr.mathworks.com/help/signal/ref/sgolayfilt.html. Accessed: 2018-08-28.

[SIR] Siropa, a library for manipulator singularities analysis. http://siropa.gforge.inria.fr/. Accessed: 2018-11-20.

[Smi17] Alan G Smith. Introduction to arudino a piece of cake, 2017.

[SMKH99] Koichi Suzumori, Toyomi Miyagawa, Masanobu Kimura, and Yukihisa Hasegawa. Micro inspection robot for 1-in pipes. *IEEE/ASME transactions on mechatronics*, 4(3):286–292, 1999.

[SNK+05] Wanxin Sun, Pavel Neuzil, Tanu Suryadi Kustandi, Sharon Oh, and Victor D Samper. The nature of the gecko lizard adhesive force. *Biophysical journal*, 89(2):L14–L17, 2005.

[SPS14] Hongyun So, Albert P Pisano, and Young Ho Seo. Caterpillar locomotion-inspired valveless pneumatic micropump using a single teardrop-shaped elastomeric membrane. *Lab on a Chip*, 14(13):2240–2248, 2014.

[SSD12] SV Shah, SK Saha, and JK Dutt. Denavit-hartenberg parameterization of euler angles. *Journal of computational and nonlinear dynamics*, 7(2), 2012.

[STM07] Manuel F Silva and JA Tenreiro Machado. A historical perspective of legged robots. *Journal of Vibration and Control*, 13(9-10):1447–1486, 2007.

[Stu05] Bernd Sturmfels. What is... a grobner basis? *Notices-American Mathematical Society*, 52(10):1199, 2005.

[Tak00] Masazumi Takada. Self-propelled colonoscope, June 6 2000. US Patent 6,071,234.

[Tap12] RL Tapping. Corrosion issues in pressurized heavy water reactor (phwr/candu®) systems. In *Nuclear corrosion science and engineering*, pages 581–633. Elsevier, 2012.

[THN14] Tomoya Tanaka, Kensuke Harigaya, and Taro Nakamura. Development of a peristaltic crawling robot for long-distance inspection of sewer pipes. In *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1552–1557. IEEE, 2014.

[TMA07] O Tătar, D Mandru, and I Ardelean. Development of mobile minirobots for in pipe inspection tasks. *Mechanika*, 68(6), 2007.

[TR20] Apicit Tantaworrasilp and Robert C Richardson. Metamorphic tensegrity structure for pipe inspection. In *IOP Conference Series: Materials Science and Engineering*, volume 715, page 012087. IOP Publishing, 2020.

[TT97] Michael R Treat and William S Trimmer. Self-propelled endoscope using pressure driven linear actuators, January 21 1997. US Patent 5,595,565.

[UH14] E. Upton and G. Halfacree. *Raspberry Pi user guide*. John Wiley & Sons, 2014.

[URCCW10] Raza Ur-Rehman, Stéphane Caro, Damien Chablat, and Philippe Wenger. Multi-objective design optimization of 3-prr planar parallel manipulators. In *20th CIRP Design conference*, pages 1–10, 2010.

[Uts79] Mikio Utsugi. Tubular medical instrument having a flexible sheath driven by a plurality of cuffs, April 10 1979. US Patent 4,148,307.

[VC19a] Swaminath Venkateswaran and Damien Chablat. Conception d'un robot bio-inspiré pour l'inspection de la canalisation. In *Journée des Jeunes Chercheurs en Robotique*, 2019.

[VC19b] Swaminath Venkateswaran and Damien Chablat. A new inspection robot for pipelines with bends and junctions. In *IFToMM World Congress on Mechanism and Machine Science*, pages 33–42. Springer, 2019.

[VC20] Swaminath Venkateswaran and Damien Chablat. Singularity and workspace analysis of 3-sps-u and 4-sps-u tensegrity mechanisms. In *International Symposium on Advances in Robot Kinematics*, pages 226–233. Springer, 2020.

[VCB18] Swaminath Venkateswaran, Damien Chablat, and Frédéric Boyer. Un robot bio-inspiré pour l'inspection de la canalisation. In *ReV Réunion-LS2N*, 2018.

[VCB19] Swaminath Venkateswaran, Damien Chablat, and Frédéric Boyer. Numerical and experimental validation of the prototype of a bio-inspired piping inspection robot. *Robotics*, 8(2):32, 2019.

[VCH] Swaminath Venkateswaran, Damien Chablat, and Pol Hamon. An optimal design of a flexible piping inspection robot. *Journal of Mechanisms and Robotics*, pages 1–37.

[VCH20] Swaminath Venkateswaran, Damien Chablat, and Pol Hamon. Design of a piping inspection robot by optimization approach. In *The ASME 2020 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, IDETC/CIE 2020*, 2020.

[VCR19] Swaminath Venkateswaran, Damien Chablat, and Ramakrishnan Ramachandran. Prototyping a piping inspection robot using a beaglebone black board. In *24éme Congrès Français de Mécanique*, 2019.

[VFCW] Swaminath Venkateswaran, Matthieu Furet, Damien Chablat, and Philippe Wenger. Design and analysis of a tensegrity mechanism for a bio-inspired robot. In *ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers Digital Collection.

[VWIQ+09] Pietro Valdastri, Robert J Webster III, Claudio Quaglia, Marco Quirini, Arianna Menciassi, and Paolo Dario. A new mechanism for mesoscale legged locomotion in compliant tubular environments. *IEEE Transactions on Robotics*, 25(5):1047–1057, 2009.

[WC97] Philippe Wenger and Damien Chablat. Definition sets for the direct kinematics of parallel manipulators. In *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97*, pages 859–864. IEEE, 1997.

[WC18] Philippe Wenger and Damien Chablat. Kinetostatic analysis and solution classification of a planar tensegrity mechanism. In *Computational Kinematics*, pages 422–431. Springer, 2018.

[Wei05] Eric W Weisstein. Gröbner basis. *https://mathworld. wolfram. com/*, 2005.

[WONY09] Kenneth J Waldron, Hayato Omori, Taro Nakamura, and Takayuki Yada. An underground explorer robot based on peristaltic crawling of earthworms. *Industrial Robot: An International Journal*, 2009.

[YB17] Cihat Bora Yigit and Pinar Boyraz. Design and modelling of a cable-driven parallel-series hybrid variable stiffness joint mechanism for robotics. *Mechanical Sciences*, 8(1):65–77, 2017.

[YL16] A. Yusupov and Y. Liu. Development of a self-propelled capsule robot for pipeline inspection. In *2016 22nd International Conference on Automation and Computing (ICAC)*, pages 84–88. IEEE, 2016.

[ZBG02] Dimiter Zlatanov, Ilian A Bonev, and Clément M Gosselin. Constraint singularities of parallel mechanisms. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 1, pages 496–502. IEEE, 2002.

[ZP03] Andreas Zagler and Friedrich Pfeiffer. " moritz" a pipe crawler for tube junctions. In *2003 IEEE international conference on robotics and automation (Cat. No. 03CH37422)*, volume 3, pages 2954–2959. IEEE, 2003.

[ZZSJ10] Yanheng Zhang, Mingwei Zhang, Hanxu Sun, and Qingxuan Jia. Design and motion analysis of a flexible squirm pipe robot. In *Intelligent System Design and Engineering Application (ISDEA), 2010 International Conference on*, volume 1, pages 527–531. IEEE, 2010.

Titre : Conception d'un robot bio-inspiré pour l'inspection des canalisations

Mots clés : Inspection de la canalisation, robots bio-inspirés, tenségrité, singularités, optimisation

Résumé : Les robots d'inspection de canalisations jouent un rôle important dans des industries telles que le nucléaire, la chimie et les eaux usées. Ils peuvent opérer avec précision dans un environnement irradié ou pollué, réduisant ainsi les risques pour les humains. Cette thèse porte sur la conception d'un robot bio-inspiré pour l'inspection des canalisations.
La thèse commence par l'étude du cas d'un robot d'inspection bio-inspiré rigide qui a été développé au LS2N, France pour AREVA. Des modèles statiques et dynamiques sont développés pour comprendre les forces de serrage et les couples des actionneurs du robot. Des validations expérimentales sont également effectuées sur le prototype pour interpréter les forces d'actionnement en temps réel.
Pour améliorer sa mobilité, l'architecture du robot est rendue flexible par l'ajout d'un mécanisme de tenségrité. Deux types de mécanismes de tenségrité sont proposés et analysés avec des méthodes algébriques pour comprendre leurs limites d'inclinaison et pour connaître l'influence des paramètres de conception. Des expériences sont réalisées sur l'un des prototypes des mécanismes de tenségrité développés au LS2N avec deux types de trajectoire en positions horizontale et verticale. Ensuite, une optimisation est réalisée pour identifier les moteurs qui peuvent permettre du robot d'inspection de canalisation flexible de passer les coudes et les jonctions pour une plage donnée de diamètres de tuyaux. Une maquette numérique du robot flexible est réalisée dans un logiciel de CAO.

Title : Design of a bio-inspired robot for inspection of pipelines

Keywords : Piping inspection, Bio-inspired robots, Tensegrity, Singularities, Optimization

Abstract : Piping inspection robots play an important role in industries such as nuclear, chemical and sewage. They can perform the assigned task with better accuracy and at the same time, they can operate within an irradiated or a polluted environment thereby reducing the risks for humans. This doctoral thesis focuses on the design of a bio-inspired robot for the inspection of pipelines.
The thesis begins with the case study of a rigid bio-inspired piping inspection robot which was developed at LS2N, France for a project with AREVA. Static and dynamic force models are developed to understand the clamping forces and the torques on the actuators of the robot. Experimental validations are then done on the prototype to interpret the real-time actuator forces.
In order to improve mobility, the robot architecture is made flexible by the addition of a Tensegrity mechanism. Two types of Tensegrity mechanisms are proposed and analyzed using algebraic methods to understand their tilt limits and to identify the influences on the design parameters. Experiments are performed on one of the prototypes of the Tensegrity mechanism developed at LS2N for two types of trajectories in the vertical and horizontal orientations. An optimization approach is then being implemented to identify the sizes of motors that can permit the flexible piping inspection robot to overcome bends and junctions for a given range of pipeline diameters. A digital model of the flexible robot is then realized in CAD software.