2023

Design of Compliant Nonlinear Torsional Springs Through Parametric Finite Element Analysis

Yotsathorn Napawan

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

UNIVERSITY
OF WOLLONGONG
AUSTRALIA

Design of Compliant Nonlinear Torsional Springs Through Parametric Finite Element Analysis

Yotsathorn Napawan

Supervisors:
Chin-Hsing Kuo
Buyung Kosasih

This thesis is presented as part of the requirement for the conferral of the degree:
Master of Research
Mechatronic Engineering

University of Wollongong

School of Mechanical, Materials, Mechatronic and Biomedical Engineering

October 2023

Abstract

This paper presents the design and optimisation of compliant nonlinear torsional springs by parametric finite element analysis. The springs are created based on a single B-spline curve, which exhibits a compact, lightweight, and simpler design than existing works. The spring is created by a combination of computational finite element analysis methods and optimisation algorithms that analyse and optimise spring designs. The models and methodology of spring creation are presented with results. The constant-torque spring was able to outperform the generic constant spring design in some aspects as well as cosine-torque spring which perform an outstanding output in term of output accuracy. This thesis explores a new type of nonlinear torsional spring with advantages above generic nonlinear torsional spring as well as difficulties, limitations and recommendations of the spring design method used.

Certification

I, Yotsathorn Napawan, declare that this thesis submitted in fulfilment of the requirements for the conferral of the degree Master of Research, Mechatronic Engineering, from the University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualifications at any other academic institution.

Yotsathorn Napawan
16th October 2023

Acknowledgments

At this moment where I am about to close another chapter in my life, the chapter that took place right here in the University of Wollongong for the last six years, I am more than thrilled to be able to acknowledge those who have forever been my biggest supporters in the journey of obtaining my Bachelor and Master degree. It has been the greatest honour to have come to know and learn from every single person whom I will be mentioning below.

First and foremost, I would like to thank Professor Chin-Hsing Kuo, who was appointed to be my supervisor during my time writing my graduation thesis. From the first meeting until now, despite all of them being through remote communication platforms, he has been a teacher, a supervisor, and more than anything else, a friend that I can confide in. Despite knowing that the research topic I chose would be immensely more difficult than other topics, he would still give the utmost encouragement, guidance and a plethora of his insights about the topic, instead of directing me to another one easily. Thanks to his support, I came to enjoy every moment of researching and unearthing all the new things there are during my thesis writing. Likewise, I am especially grateful for him when he has always been trying to adjust his flexibility in time so our timetable would align due to my seldom unavailability of being both a full-time student and a restaurant owner. I would have never come to this point if it was not for him.

I would also want to thank my co-professor Buyung, who was the greatest help when it came to technical question in the ANSYS program, which was some of the few most important components I needed to write my thesis, and if it was not for him and his extensive knowledge in these programs, it would not be possible for me to commit in exploring this topic.

It would not make sense for me if I did not thank my parents, Chairat, and Yuvadee, not just for constantly encouraging me and cheering me up when I got extremely stressed because of my thesis but also for untiringly pushing me to move forward in life and to never back down from a challenge. Their love and support have contributed greatly in reminding me why I chose this path of research, and why I should never give up when being confronted by any difficulties, especially during the last few years. I thank them for always being by my side and standing by every choice I have made in my studies as well as in my life, and most of all, for believing in me.

I would also wish to extend my gratitude to a very special person who has helped me in making all of this possible, my partner, Vy. Her love and immense encouragement have always succeeded in building up my self-confidence whenever I was in doubt of my ability, or of my choice of doing research academically. Two years have passed and now when I am almost there to finish one of my most important milestones in life, I am grateful for us to be able to stay just like this, despite all the hardships from having our time cut off due to my commitment to pursuing my degree. And more than anything, I want to thank her for always being a study mate that I can learn, especially when it comes to English, it has helped me significantly in writing and perfecting this thesis in the best possible manner.

And lastly, I would like to express my utmost appreciation to the University of Wollongong, for creating this valuable opportunity for me to study in a foreign environment in the last six years. Through such opportunity, I have had friendships that can last beyond my years in Australia and created professional connections/networks with people of remarkable expertise from different fields that will surely be beneficial for my future career. And most of all, I have been able to pursue what I truly love – Mechatronic Engineering. Along with all of my friends and everyone else who have contributed to this milestone of mine, though not mentioned, it has been an absolute honour for me to meet everyone here during my university years.

Table of Contents

List of Table

List of Figure

Chapter 1: Introduction

Torsion springs are categorised under compliant design. Compliance is a material property that represents the flexibility of the material and structure. The tool that utilises the flexibility of the material to accomplish a specific task is categorized in the complaint mechanism[1]. A compliant mechanism can be difficult to design compared to rigid parts with joints and hinges. However, with more advanced technology and knowledge in the past decades, a compliant mechanism can be designed with ease by the development of new materials and higher computational capabilities. Moreover, the compliant mechanism is required in some applications in which traditional mechanisms are not applicable. Compliant mechanisms are created as one-piece and jointless. Such a design can minimize the lubricant requirements, wear, and backlash, which can be disadvantages of the generic design mechanism design[2]. The design of a compliant mechanism can help minimise production costs by assembly costs reduction, storage space reduction (reducing parts and connections) and manufacturing process simplification, such as injection moulding and metal casting. However, the disadvantages of compliant mechanisms include a limited range of motion and stress relaxation, which increase the design complexity. The compliant mechanism can be challenging to design in some applications with complex requirements and some of the trade-offs due to its nature. Compliant mechanism motions utilise the elasticity of the design and material, which is generally used in repetitive motions. The repetitive movement can result in high-stress concentration at the bending locations and can lead to fatigue failure of the mechanism in long-term usage. This makes fatigue tests and computational analysis necessary for compliant mechanism design.

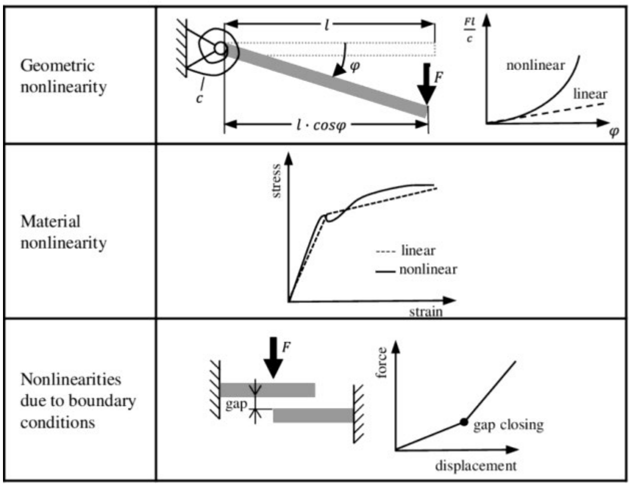


Figure 1: Structural Nonlinearities[3]

Nonlinear torsional spring is categorised under the compliant mechanism class. The spring provides nonlinear output functions such as nonlinear load displacement and nonlinear torque displacement. Nonlinear output occurs when the input energy of the nonlinear spring is partially absorbed by the compliant mechanism as strain energy, which can be called ‘Structural Nonlinearity’. The nonlinear output of the design can be achieved by three methods which are 1. Geometric nonlinearities, 2. Material nonlinearities, and 3. Change-in-status nonlinearities as Figure 1. Material nonlinearities occurred according to the material properties such as increasing temperature, material yielding, and special material synthesis such as psudoelastic material[4]. Change-in-status nonlinearities occurred due to the contact of the geometry, which provides a change in stiffness to the compliant mechanism [5]. Nonlinear torsional springs only adopt geometric nonlinearities to achieve a nonlinear result of the spring design. Nonlinear spring is generally used in many applications, such as microelectromechanical systems, constant force output, and human interface devices, but not in some other applications, such as gravity compensation. The output of the nonlinear spring can be different in each application based on the design criteria, for example, constant force nonlinear spring and S curve nonlinear spring. The output of the spring can be manipulated by changes in design geometry, which results in changes in strain energy absorption of the spring, which is one of the structural nonlinearities properties.

The term ‘Finite element analysis (FEA)” in engineering can be traced back to the 1950s through aerospace engineering application, having its first commercial computer program introduced in the 1960s [6]. Finite element analysis is the simulation of physical phenomena calculated by the finite element method (FEM). The partial differential equations (PDEs) are calculated to compute a simulation structure behaviour with specific material properties such as stress and strains. FEA can be used to find an approximate solution to the physical phenomenon, and it is commonly used to reduce the resources required for repetitive experiments for design optimisation and testing. According to [7], FEA was used to perform a design optimisation by changing the design structure based on the results until the design criteria were satisfied. FEA is a powerful simulation program that can be used to perform design optimisation without conducting practical experiments to identify the design viability, which can reduce the number of resources used in the experiment and shorten the optimisation duration.

1.1 State of the Art

Compliant design is one of the methods commonly used to create springs for a specific application, such as a human-robot interaction device, gripping device, electrical contact, and a series actuator. The type of spring used in the applications can be categorised into two major spring types based on the type of input motion, which are torsional spring and tension-compression spring. The springs can also be categorised based on the spring output, which are linear and nonlinear output. Thus, compliant design is used in both linear and nonlinear spring designs, as well as in torsional and compression springs. The majority of torsional spring designs use a design methodology that includes design optimisation and compliant mechanism design. Similarly, nonlinear tension-compression springs used a compliant design with an optimisation algorithm to achieve the desired goal as in torsional spring design studies. This method has been commonly used in many studies to create a complex torsion spring design. The spring design of the nonlinear torsional spring in each study also introduces a similar structural pattern, which represents a circular torsional spring with three main components, which are shaft, ring, and compliant links. However, this thesis aims to achieve the result in a different way. A single b-spline based-compliant nonlinear torsion spring has never been made before in constant-torque application. The spring is constructed based on a single b-spline geometry. Furthermore, the spring also has compliant mechanism property. The spring design can lower the complexity of the spring shape while maintaining its performance. Moreover, at the present, the gravity compensator is created by using a complex approach that requires a bulky design with multiple components that are not suitable for mass production. The cosine-torque spring can be used to replace some of the complex gravity compensators for lightweight, simple, and compact designs. This thesis presents a new spring design approach with lower production cost, lower design complexity, compact and high performance.

1.2 The Problems

The generic torsion spring has a limited design possibility and comes with multiple disadvantages. Generic torsion springs require multiple assembly parts, leading to a heavy and bulky system. The additional parts lead to additional manufacturing, assembly, and maintenance costs. The torsion spring with a basic design is still being used due to the lack of an alternative spring design.

This study explores the research question, ‘Is a single b-spline based-compliant torsional spring created by using parametric finite element analysis suitable for providing output in constant-torque function and cosine-torque function?’.

1.3 Aims and Objectives

The aim of this thesis is to create nonlinear torsional springs using compliant mechanisms, finite element analysis and optimisation algorithms to design a b-spline based-compliant nonlinear torsion spring for achieving constant and cosine-torque springs. Designing compliant mechanisms for nonlinear torsional springs can be challenging due to the complexity of the design. This study introduces an optimisation algorithm to assist with the compliant design methodology. Optimisation algorithms are generally used to solve complex mathematical problems, which can be expressed as a mathematical expression. By combining the optimisation method with parametric finite element analysis, geometric optimisation is created. Geometric optimisations optimise the design geometry to fulfil the set of criteria that can help overcome the complexity in the design of a compliant mechanism.

The objectives of this thesis are to conduct a literature survey on nonlinear spring designs and methodology used; to determine nonlinear spring design criteria; to create nonlinear springs; to perform experiment to verify the spring performance in practical; to determine the reliability of b-spline base compliant springs by the performance of compliant spring; to compare the performance of between each specific spring and with existing design solutions.

1.4 Thesis Organization

This chapter introduced background knowledge of compliant nonlinear springs and research aims and objectives. The next chapter reviews the relevant work related to compliant nonlinear spring design. Chapter 3 sets up the nonlinear spring problems and criteria as well as the methodology used to achieve optimal results. Chapter 4 presents the optimal design of 3 different spring types with two different materials in each spring type with computational analysis results. Chapter 5 verifies the torsion springs' performance in practice by conducting a torsional test experiment. Chapter 6 compares the performance of spring designs in this study as well as existing studies. Chapter 7 summarises the conclusion, recommendation, and future work of this thesis.

Chapter 2: Literature Review

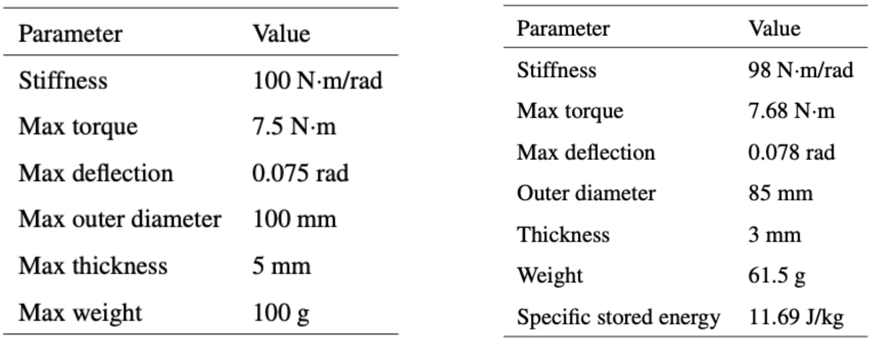
The focus of this research is to create nonlinear torsion springs with prescribed output torque. This literature review will explore related research on the design methodology of compliant torsion springs and nonlinear design. The nonlinear design literature review topic is divided into two main sections, which are nonlinear reaction torque design and nonlinear reaction force design.

2.1 Linear-Compliant Torsional Spring Design

Linear-compliant torsional spring is commonly used in soft-robotic and physical Human-Robot Interaction applications. The spring is designed to transfer the input torque of the actuator to the output structure for improving safety and dynamical adaptability in general. In detail, compliant torsional springs can reduce peak force transmissibility to prevent rigid movement during force transmission, which can cause damage to the human body and device. The spring also improves the maximum storable energy capacity of the device and provides large elastic deflection of the spring.

In physical human-robot interaction applications, the major criterion of the mechanism is to transfer the movement input to the human body without any rigid movement. Moreover, the output movements from the mechanism must follow the prescribed kinematic pattern to ensure that the device can provide safety as well as high performance. According to [7], the research aim is to create a compact torsional spring as a series of elastic actuators for assistive wearable robots. The design goal of the elastic modulus (torsion spring) is to create a compact compliant torsion spring with a suitable amount of stiffness, maximum torque, deflection, thickness, diameter, and weight of the spring, as in Table 1.

Table 1: Design requirements(left) and Design results(right) of Compliant Element [7]



Parameter	Value
Stiffness	100 N·m/rad
Max torque	7.5 N·m
Max deflection	0.075 rad
Max outer diameter	100 mm
Max thickness	5 mm
Max weight	100 g

Parameter	Value
Stiffness	98 N·m/rad
Max torque	7.68 N·m
Max deflection	0.078 rad
Outer diameter	85 mm
Thickness	3 mm
Weight	61.5 g
Specific stored energy	11.69 J/kg

The spring is designed by using geometric optimisation with 17 design parameters while 1 of the parameters is the amount of topology, and the rest are the geometric parameters of each topology. The spring design is optimized by using FEM simulation in COMSOL to evaluate the design. After the evaluation, the design topology and parameters will be adjusted according to the result until the optimal result is obtained. After 12 simulations, the optimal design of the torsional spring in Figure 3 was obtained. The spring can achieve a safety factor of 1.5 for the yield stress of the material and can store 11.69J/kg of energy. The experiment is conducted to verify the computational output of FEM, which illustrates that the design can satisfy most of the criteria with slight errors, as in Table 1.

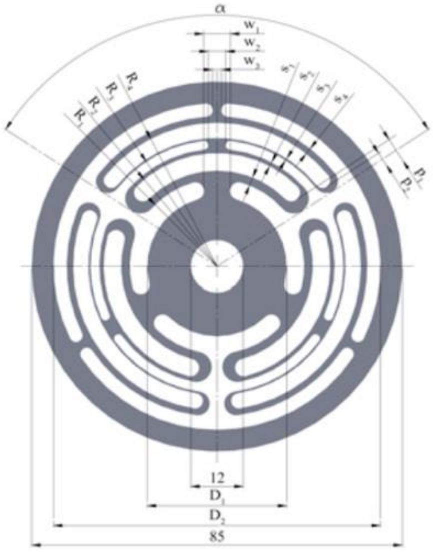


Figure 2: Design Optimisation Parameter Diagram [7]

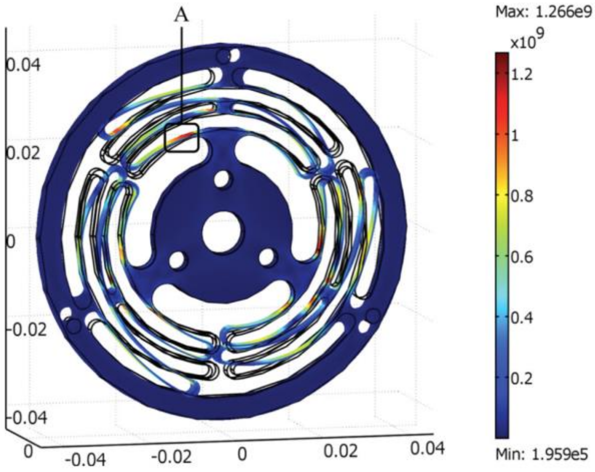


Figure 3: von Mises stress and Spring deformation [7]

In another research conducted by Accoto [8], The torsional spring was created for a series of elastic actuators for lower limb robotic orthosis. The methodology of this research is similar to the previous research conducted by Carpino [7] but for a different application. The design goal of this study is to create a series elastic actuator that can achieve high energy-storing capacity, torsional control stability and minimizing spring weight. The criteria of the design are based on the application, which includes spring stiffness, maximum torque and storable elastic energy while having spring diameter and spring thickness as design constraints.

Table 2: Design Criteria vs Design Result Comparisons [8]

Parameters	Design Criteria	Design Result
Stiffness N.M.rad^{-1}	250	235
Maximum torque (N.m.)	60	60
Storable elastic energy(J)	7.5	7.64

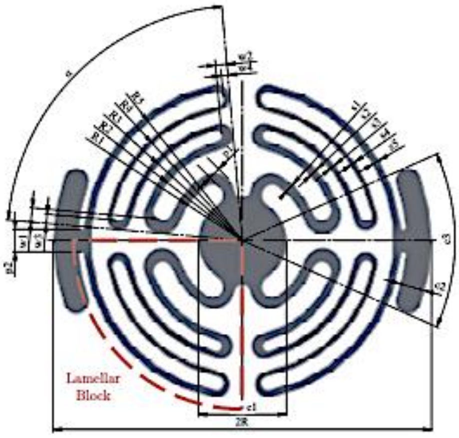


Figure 4: Design Optimisation Parameters Diagram [8]

This research methodology includes a design geometry optimisation method to achieve the optimal spring design. The spring design is based on a monolithic disc shape to minimise spring weight and footprint. However, the original spring design cannot achieve the expected storable energy level. The worm-shaped elastic elements are added and optimised to help increase the storable elastic energy of the spring. The optimisation parameters include 17 adjustable parameters for the design geometry. COMSOL program is used for FEM to evaluate each design input. The result of the optimal spring design is shown in Figure 4. After the optimal design was obtained, the experiment was conducted to verify the performance of the spring pack in real situations. The spring experimental results were acceptable and can achieve some of the criteria with minor errors, as in Table 2. Moreover, The stiffness value of the FEA result is 15.5% different from the actual experiment, which is expected due to the experiences from previous studies. The spring was able to perform correctly in terms of stiffness control for all testing frequencies.

A similar study by Negrello [9] focused on creating a high-compliance spring for robots with soft joints. The study focused on designing and comparing four different types of springs created from beam-based calculation and geometrical parameters optimisation. The initial spring design is based on a spring with a pinned end with spring parameters included N and Ns as in Figure 5, while The design constraints include an inner and outer diameter of the pinned end elastic module. The main advantages of pinned end design are the increasing spring deformation and spring modularity due to the high level of stress and stiffness of the design, which is normally obtained by increasing the axial length. The spring designs were tested using FEA, as in Figure 6; four types of spring geometries were created with Aluminium (Al 7075-T6) for practical tests. As a result, the designs have a linear behaviour, while the design with a hinged end constraint has higher compliance of 75%. Moreover, the hinged end constraint can lower the stiffness by about 11% in a higher spoke number.

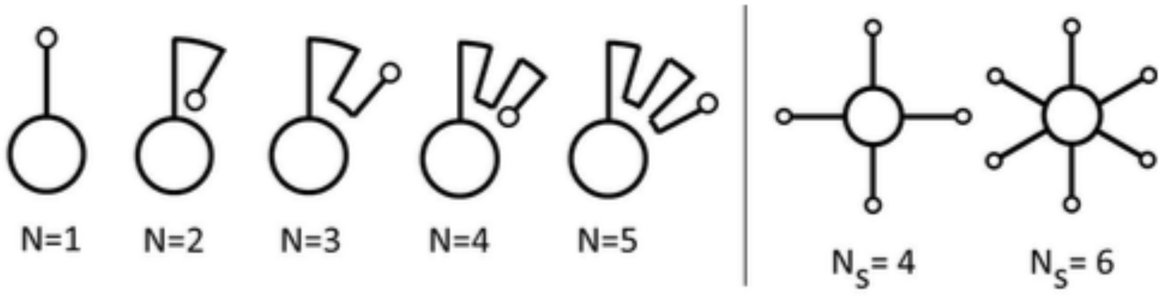


Figure 5: Spring Design Parameters' Diagram [9]

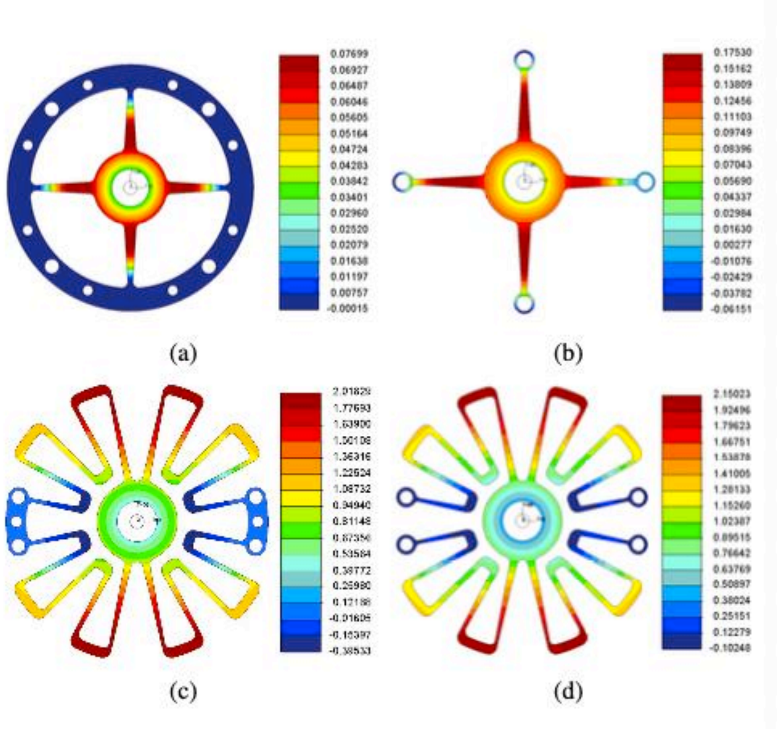


Figure 6: Spring FEA Results in 4 Different Designs [9]

The spring design methodologies of the three studies [7-9] are similar in some aspects. The design geometry optimisation was introduced in both compliance torsional spring design studies. Moreover, the optimisation parameters are similar in both studies [7] and [8]. Even though the design constraints of Accoto [8] is less flexible than the study from Carpino [7] in term of design thickness and size. On the other hand, the spring design of [9] introduced a different approach from [7] and [8]. In detail, the spring design optimisation was created based on the beam deflection and bending moment calculation, which makes the study contain a lower amount of optimisation parameters. The results of the three studies are acceptable and usable in their design application, and the methodology of each spring design can be used in the different studies.

2.2 Nonlinear spring design methods

A nonlinear spring is one of the spring types with a unique nonlinearity of load-displacement relationship function. In general, the nonlinearity of the spring is considered an error or design flaw which can provide a drawback in control of the device. However, at the same time, such nonlinearity can be used in some applications to improve the performance and overall design configuration. The nonlinear spring is intentionally designed for many applications, such as microelectromechanical systems, robotic joints, vibration absorption systems and balance mechanisms. The nonlinear spring used in each application requires a unique nonlinear load-displacement relationship and other constraints. In this literature review, nonlinear spring design methodologies are introduced and categorised into two aspects by types of output and nonlinear spring design technique. The types of output include force-displacement output and torque-displacement output. Meanwhile, the nonlinear spring design technique includes a compliant mechanism and structure, magnet, cam, gear combination, and a combination of techniques. This literature review only focuses on compliant mechanisms and structures that are directly related to the research.

2.2.1 Nonlinear Force-displacement output

In this sub-section, the overall design goal of the studies is to achieve a unique nonlinear force-displacement relationship of the designed spring. In each study, the nonlinearity criteria can be different for each design and application.

2.2.1.1 Constant Force by Compliance & Structure

The majority of the nonlinear force-displacement spring introduces a compliant mechanism and structure as a technique to achieve the spring design. One of the nonlinear force-displacement spring types is constant force spring. Constant force compliance spring design is suitable for applications that require a limited amount of output force in their application.

The study by Meaders [10] created a near-constant force spring for electrical contacts. The electrical contacts are small-scale compliant spring that connects the electrical component. The methodology used for creating the spring in this study includes non-linear finite element analysis and optimisation. The design of the compliant constant force electrical contact can be illustrated in Figure 7.

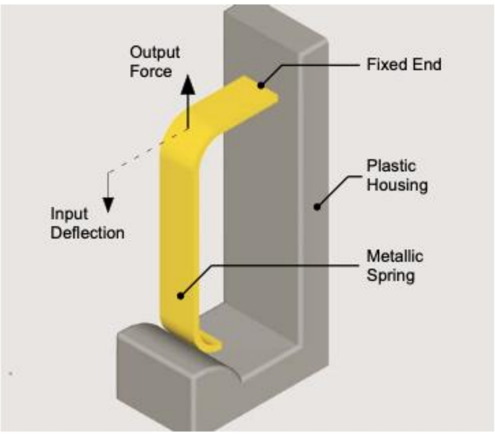


Figure 7: Compliant Constant Force Electrical Contact Model [10]

The design criteria of the constant force spring were to create a small-scale spring six by 6 mm. with constant out-of-plant depth. This study introduced two objective functions and design constraints to compare the performance of each objective function. The difference between the two objective functions was the assumption that the spring and cam contacts. The assumption can affect the performance of the optimal spring design and optimisation constraints. As a result, the optimisation results of the robust design optimisation method (Cam is not always in contact.) had a higher average percentage of constant force value than the deterministic design optimisation method (Cam is always in contact). However, the deterministic method illustrates a lower error of constant force after the preload compared to the robust objective function. The optimal design can achieve linear reaction force at 0.6 mm with a linearity of 98.2%, while the optimal design area size is 36 mm^2. The optimal design and computational test are illustrated in Figure 8. However, the study did not mention the comparison of the computational time of both objective functions, which can help determine the performance.

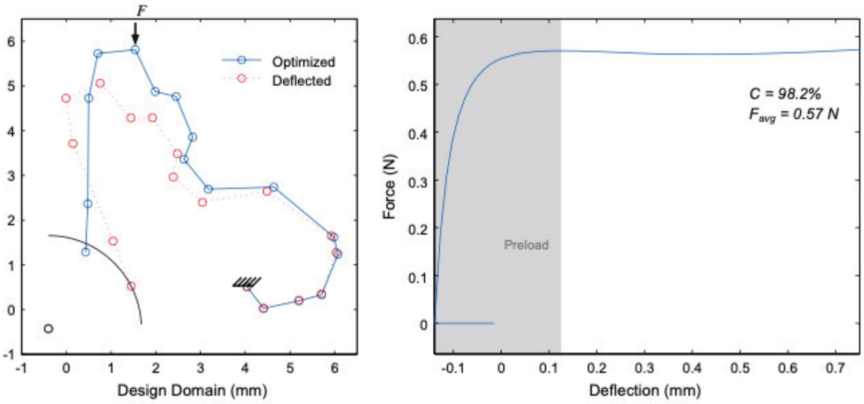


Figure 8: Constant Force Spring Control Points Before and After Deflection (Left)

Constant Force Spring Reaction Force Function (Right) [10]

Another study by Pedersen [11] created a compliant spring with constant force output for microsystem application. The spring design criteria were based on electrostatic microactuators, which are widely used in the applications. The mechanism diagram included actuator input and nonlinear spring to generate a constant force-displacement relationship of the output force, as in Figure 9.

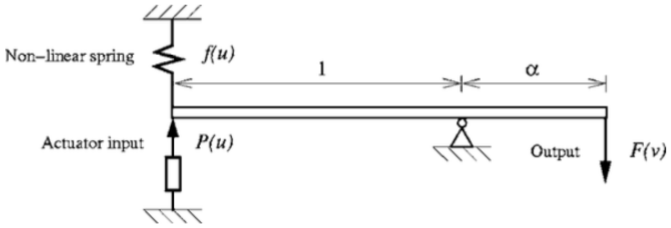


Figure 9: Spring Mechanism [11]

The methodology of spring design includes a double optimisation method for achieving optimal compliant constant force spring design. The optimisations included topology optimisation routine and size optimisation routine for achieving optimal shape and size of the compliant spring components. The topology optimisation included outer and inner iterative loops to optimise the material distribution of the geometry and to ensure that the output force is constant. The size optimisation was based on two types of beam modelling, which are slender beam idealization and stiff triangulated unit. After the optimisation, three optimal spring designs were created, as in Figure 10. However, only design B was able to fulfil the criteria of a constant force-compliant spring. Design B can achieve the most linear constant force result with a maximum input of 0.008 μN, which has an output displacement range of 10 micrometres. The footprint of the optimal design was 200*100 μmm^2. In this study, fatigue tests and practical tests can be conducted to ensure the performance of the compliant spring.

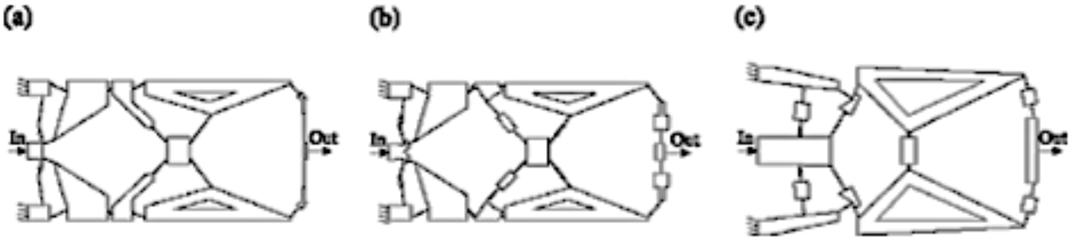


Figure 10: Compliant Constant Force Spring Geometries [11]

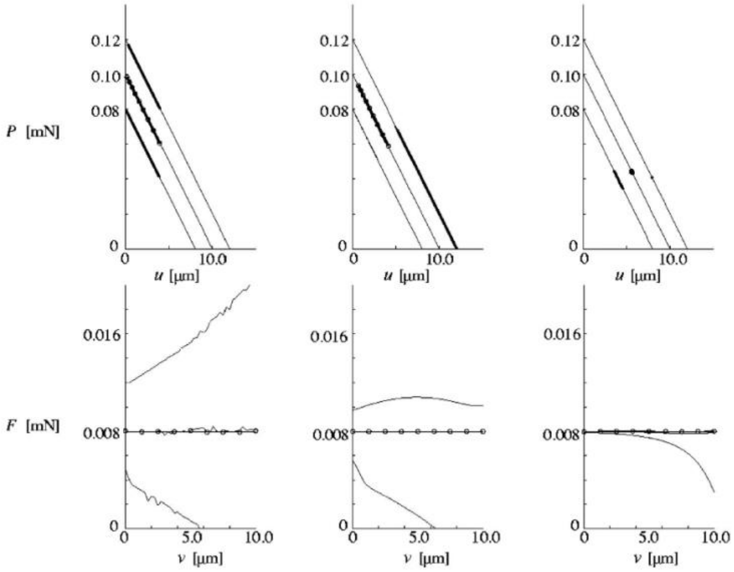


Figure 11: Compliant Constant Force Spring Output for a, b, and c designs (left, middle, right) [11]

A similar study by Liu [12] created a constant-force mechanism for a passive force regulation device that can be used in robotic automation, precision manipulation, and overload protection applications. The mechanism of the compliant constant-force mechanism is shown in Figure 12. The methodology of the study was similar to the study [11] in the aspect of double optimisation. However, the optimisations used in this study include topology and geometry optimisation methods to achieve the optimal constant-force mechanism design. The topology optimization goal is to minimize the error of output and objective forces, which results in constant output. The optimization process also includes sensitivity analysis, filter scheme, update scheme and projection scheme, which are calculated as expressions to increase the performance of the optimization method. For example, the sensitivity analysis can illustrate the gradient value of the objective function, which helps identify the direction of the next optimization loop. Moreover, this design also introduces a spring as a component in the mechanism to help achieve the constant force output.

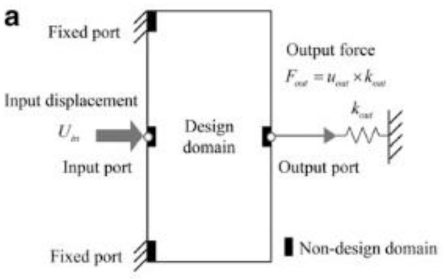


Figure 12: Constant Force Mechanism Design Diagram [12]

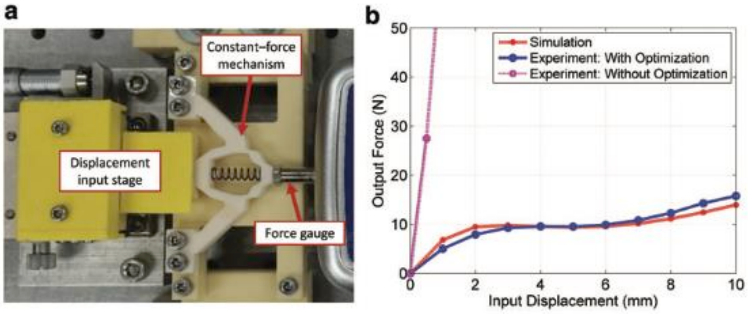


Figure 13: Constant Force Mechanism Specimen in Experimental Setup (a)

Constant Force Mechanism Results (b) [12]

After the optimisation process, the optimal design can achieve a constant-force displacement relationship. Figure 13 illustrates both the force-displacement relationship of the prototype and simulation as well as the prototype testing experiment setup. The prototype was made for testing with the electric gripper device to create a constant-force gripper. As a result, the gripper with a constant-force compliant mechanism was able to pass the gripping test in multiple object samples, as in Figures 14 and 15. The constant force mechanism can achieve 6.5 mm of constant force output, while the footprint of the optimal design was 30*60 mm.

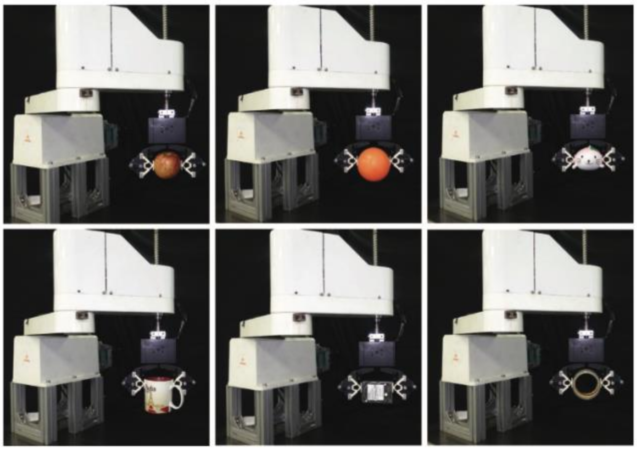


Figure 14: Constant Force Mechanism Experiments with Different Objects [12]

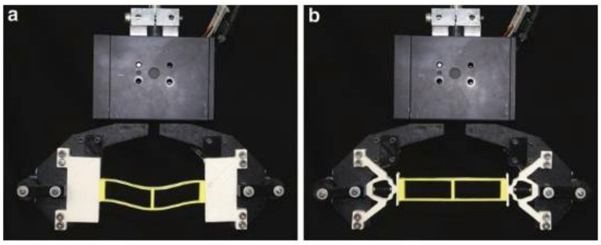


Figure 15: Experimental Object Under Compression General Setup (a)

Constant Force Mechanism (b)[12]

Wei [13] created a constant-force compliant mechanism for robot polishing devices. The mechanism can help increase the quality of the workpiece by reducing the variation and overshoot of the device. The constant-force compliant mechanism was created by using a combination of negative-stiffness and positive-stiffness beams to achieve the constant-force mechanism. The beam parameters were optimised by using an optimisation algorithm to achieve the optimal design of the constant-force mechanism. The optimization objective is to maximise the constant-force motion stroke while having the safety factor and the constant force value (7 N) as design constraints. The optimal design was tested in practical experiments and FEA analysis. Al 7075 was selected as a material for this study due to its high durability and elasticity and its creep performance. The mechanism can achieve constant-force output and reduce the vibration of the polishing device. The vibration of the polishing machine is negated by the mechanism under 900 rpm, and the performance decreases after the rotational speed exceeds 900 rpm, which is caused by the machine itself. The issue can be resolved by using a force transmission structure and separating the mechanism from the machine. The results are shown in Figure 17.

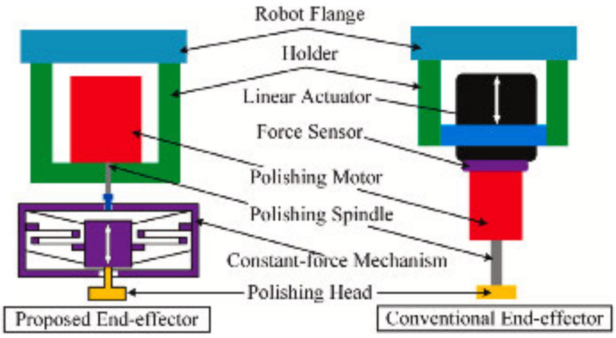


Figure 16: Comparison between polishing devices with and without constant force mechanism [13]

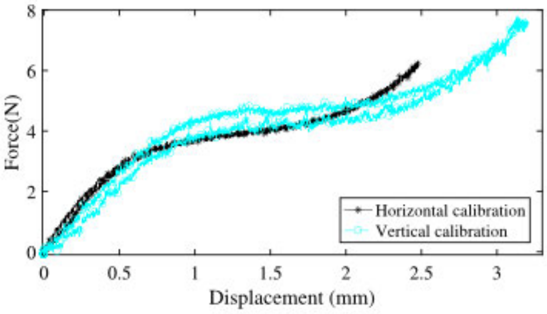


Figure 17: Experimental result of constant-force mechanism [13]

Bilancia [14] presents the compliant constant force mechanism for the manipulation system. The mechanism of this study is a beam-based monolithic long-stroke design. The beam includes a different section of thickness, which acts as a rotational joint, as in Figure 18. The study uses a combination of ANSYS and MATLAB software for FEA analysis and Genetic algorithm analysis. The genetic algorithm setting includes 100 generations with a population size of 30 candidates. The algorithm stops when the iteration exceeds 3000, or the difference in the average value of the generations is less than 10^{-2}. The feasible design from the optimisation was created by using a 3D printer with ABS material with a layer height of 0.100 mm. The constant force mechanism was able to provide a nearly constant output force without any plastic deformation in computational analysis and practical experiments. The output includes all three different configurations, as in Figure 19. As a result, the spline segment configuration provided the largest displacement, while the small-length configuration achieved the most stable constant force function. The experimental results and computational analysis results of all configurations were similar overall. However, some minor errors occurred due to manufacturing errors, technical limits of the 3D printer and friction from the experimental settings. The optimal designs can achieve up to 25mm displacement at 1.5 N maximum reaction force. The footprint of the optimal design is 100*60 mm.

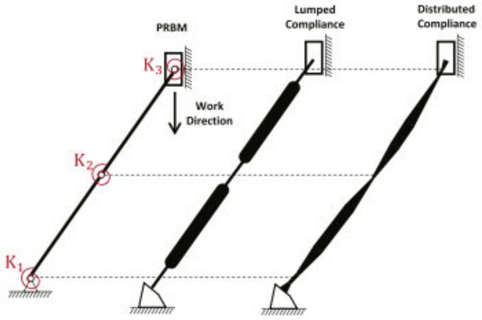
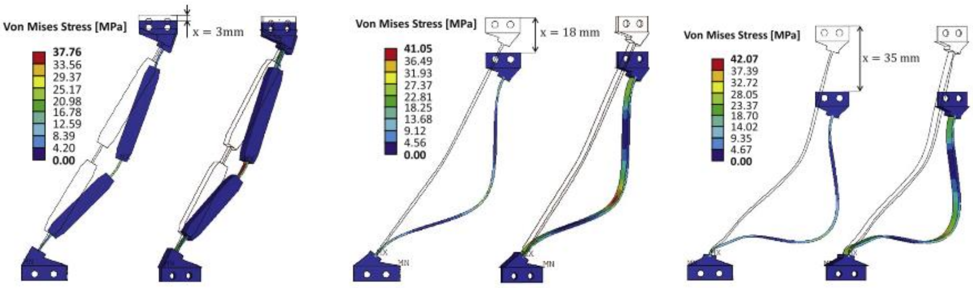


Figure 18: Constant force mechanism design [14]



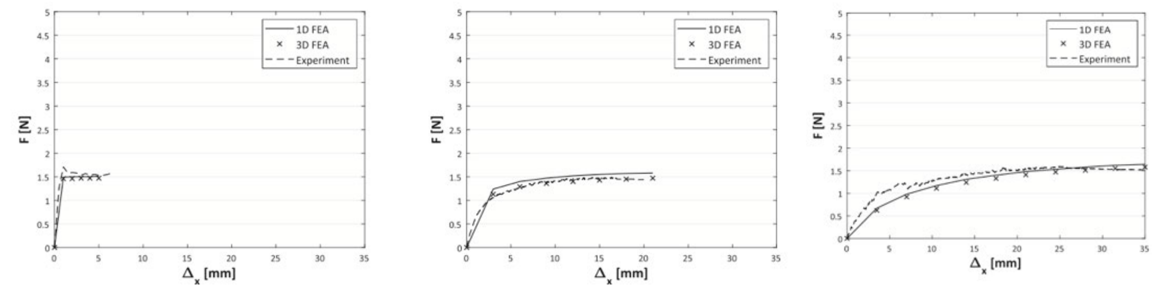


Figure 19: Design configurations and experimental results [14]

Another research by Jutte [15] created a constant-force compliant mechanism by using an optimisation process. The design approach of this study was based on the branch network of 9 compliant beams, as in Figure 20. The position of the control point on the beam was adjustable by the optimisation algorithm to generate a spline curve that can provide a constant-force output. The study requires a double optimisation method to achieve the optimal design of the spring. The first optimisation aimed to optimise the shape of the spline by adjusting the control points of the branch and subbranch until the constant-force output was achieved. The 2nd optimisation is the size and out-of-plane thickness optimisation to adjust the value of maximum torque and maximum stress until the criteria are satisfied. The algorithm used for the optimization in this study is Genetic Algorithm. After the optimisation process, the optimal design and the output result are illustrated in Figure 21. The constant force spring can achieve around 100mm of constant reaction force with a maximum load at 150 N. The footprint of the spring design is 736*685mm. In addition, prototype testing can be included to ensure the performance of the constant-force spring. The methodology of this study was also used to create other types of compliant springs, which are the J-curve and S-curve springs. The details will be discussed in the next section.

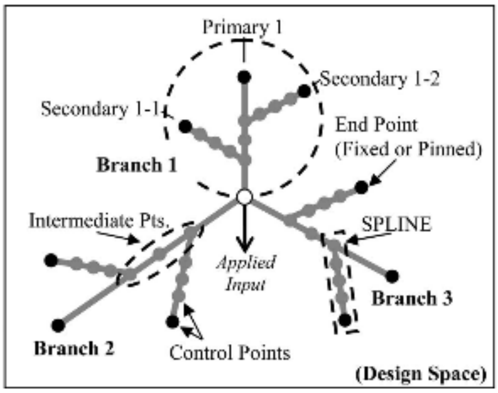


Figure 20: Branch network design topology [15]

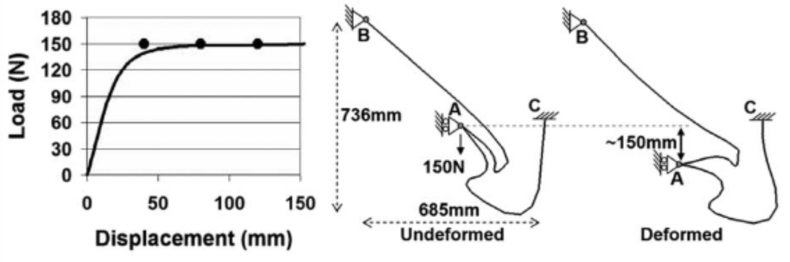


Figure 21: Final constant force geometry and result [15]

2.2.1.2 Nonlinear Force by Compliant Design

Different approaches to achieve nonlinear-force output were introduced in this literature review including compliant mechanism spring, origami-inspired constant-force mechanism design, contact pairs and helical spring.

As mentioned previously, a study by Jutte [15] presents not only constant force springs but also other types of nonlinear springs. The other type includes the J-curve spring and the S-curve spring. The spring design methods are almost identical to the constant force spring design, while the objective function of the optimisation of each other spring type was changed accordingly to the design goal while other aspects of the design optimisation setting remained unchanged. The optimal design of the S-curve spring and J-curve spring, as well as load-displacement analysis results are respectively presented in Figures 22 and 23.

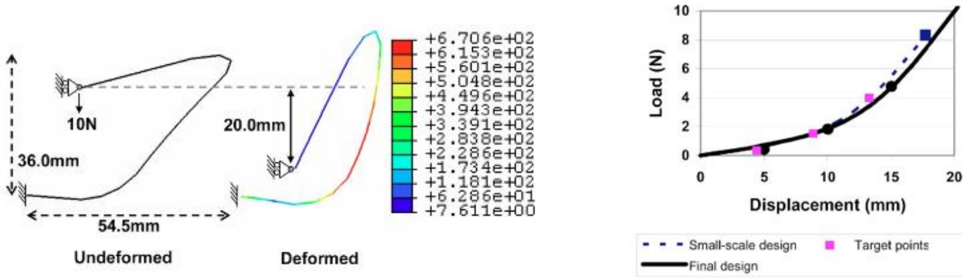


Figure 22: J-Curve spring geometry and results [15]

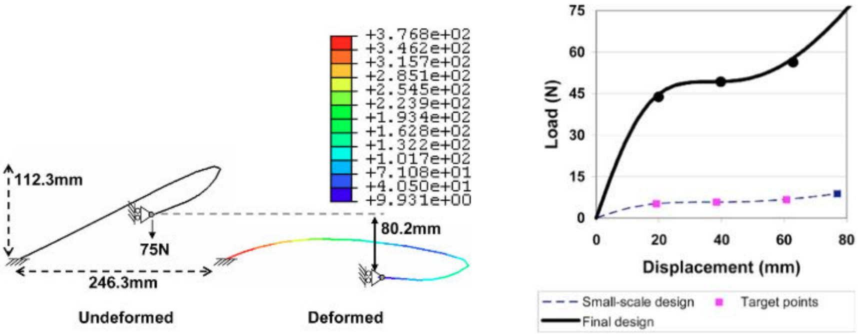


Figure 23: S-Curve spring geometry and results [15]

A study by Potkonjak [5] presents a nonlinear-force micro-spring design for a micro-system application. The nonlinear force spring was created by using a contact pair to achieve stiffness changes during the spring extension. The spring can achieve variable stiffness due to the contact pair interaction. The contact pair geometry is presented in Figure 24, which includes one set of contact pairs. After the initial design, the FEA was conducted, as in Figure 25. The Figure illustrates the difference in stiffness between before and after the contact pairs made contact. After the computational analysis, the prototype of the spring with multiple contact pairs was made for the experiment. The spring can provide a unique force-displacement function due to the interaction of the contact pairs during a certain extension range. The output force–displacement plot is shown in Figure 26.

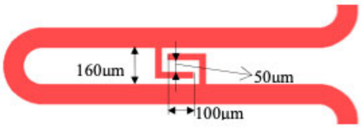
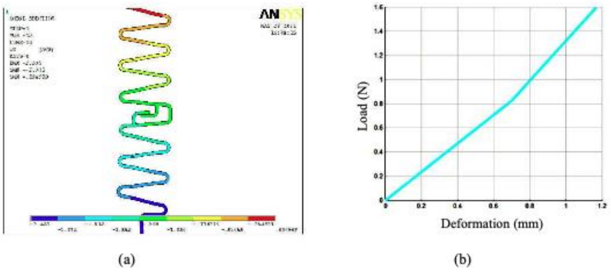


Figure 24: Contact Pair design geometry [5]



(a)

(b)

Figure 25: Computational result of spring with a contact pair [5]

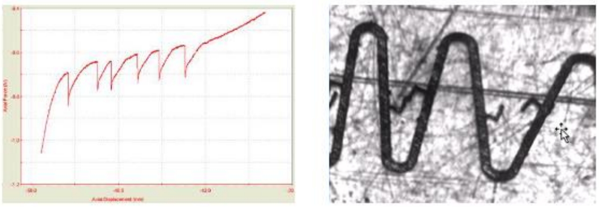


Figure 26: Experimental result of spring with multiple contact pair [5]

Fawazi [16] presents a constant-force slotted disc spring created by using a geometric design algorithm. The geometric design algorithm developed in this study can be used as a guideline for designing the desired slotted disc spring. The geometric design algorithm can be transferred as a flowchart in Figure 27. The optimization parameters of this study include spring thickness, free height, total height, inner diameter, minimum diameter and the width of the slot. Interestingly, the design constraints of this study do not include the maximum range of load and displacement. After the optimization, The final design of the slotted disc spring was tested for the load-displacement function to evaluate the performance of the spring. The spring created by the inverse geometric design algorithm can satisfy the target design, as in Figure 28. However, prototype testing and fatigue tests can be conducted to ensure the performance and reliability of the disc spring.

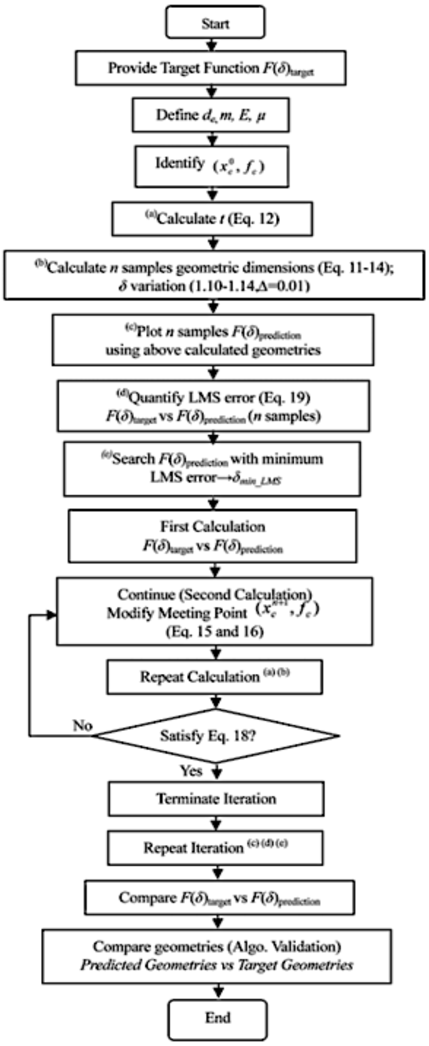

Start
Provide Target Function $F(\delta)_{target}$
Define d_e, m, E, μ
Identify (x_c^0, f_c)
(a)Calculate t (Eq. 12)
(b)Calculate n samples geometric dimensions (Eq. 11-14); δ variation (1.10-1.14,Δ=0.01)
(c)Plot n samples $F(\delta)_{prediction}$ using above calculated geometries
(d)Quantify LMS error (Eq. 19) $F(\delta)_{target}$ vs $F(\delta)_{prediction}$ (n samples)
(e)Search $F(\delta)_{prediction}$ with minimum LMS error→δ_{min_LMS}
First Calculation $F(\delta)_{target}$ vs $F(\delta)_{prediction}$
Continue (Second Calculation) Modify Meeting Point (x_c^{n+1}, f_c) (Eq. 15 and 16)
Repeat Calculation (a) (b)
No
Satisfy Eq. 18?
Yes
Terminate Iteration
Repeat Iteration (c) (d) (e)
Compare $F(\delta)_{target}$ vs $F(\delta)_{prediction}$
Compare geometries (Algo. Validation) Predicted Geometries vs Target Geometries
End

Figure 27: Comparison of Load-Displacement function of the Slotted Disc Spring [16]

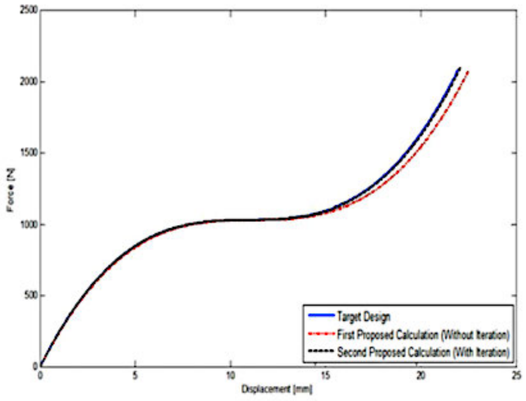
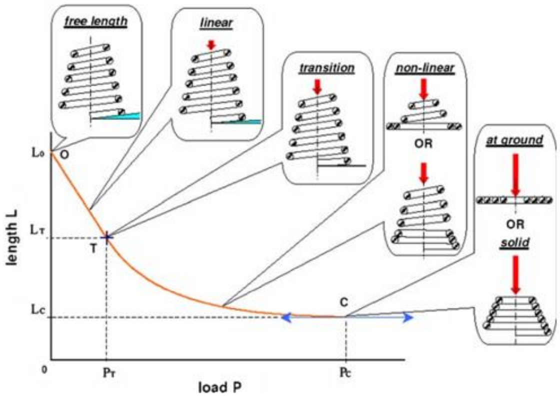
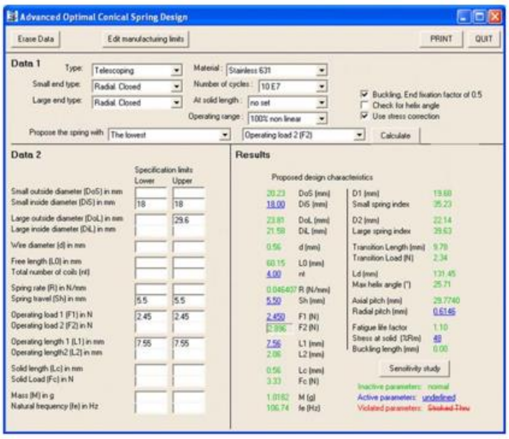


Figure 28: Geometric design Algorithm Flow Chart [16]

Another example of nonlinear design is nonlinear design by conical spring. A conical spring is a spring with a cone shape that has a nonlinear force-displacement relationship, as in Figure 29 (left). Study [17] introduced an optimisation algorithm into the conical spring design to help achieve nonlinear force for electrical contractor applications. The optimisation problem includes seven adjustable parameters and 46 design constraints, for example, diameters, number of coils and helix angle. The optimization parameter of this study includes five design parameters. Three of the parameter represents the dimensions of the spring, and two represents operating parameters. Interestingly, this optimization setting includes 46 design constraints, which limited the shape of the conical spring. The spring design optimisation also considered the telescoping state of the conical spring for better result performance. The criteria of the spring included the nonlinearity of the design and other perspectives, such as fatigue life, maximum stress, and maximum load. The sequential quadratic programming (SQP) algorithm is also implemented within this study. The study also presented the assistance tool created for the general design and calculation of the conical spring. The tool has a user-friendly interface. The input parameters, such as spring design parameters and input force can be adjusted easily, as in Figure 29 (right). The final spring design prototype was created and tested for mechanical properties. As a result, the optimal spring design prototype from the optimisation can achieve a 56% improvement in the maximum load, and the prototype was able to telescope fully without any plastic deformation.




Advanced Optimal Conical Spring Design
Erase Data
Edit manufacturing limits
PRINT
QUIT
Calculate
Sensitivity study

Data 1

Type: Telescoping
Small end type: Radial. Closed
Large end type: Radial. Closed
Material : Stainless 631
Number of cycles : 10 E7
At solid length : no set
Operating range : 100% non linear
☑ Buckling, End fixation factor of 0.5
☐ Check for helix angle
☑ Use stress correction
Propose the spring with The lowest Operating load 2 (F2)

Data 2

	Specification limits Lower	Upper
Small outside diameter (DoS) in mm		
Small inside diameter (DiS) in mm	18	18
Large outside diameter (DoL) in mm		29.6
Large inside diameter (DiL) in mm		
Wire diameter (d) in mm		
Free length (L0) in mm		
Total number of coils (nt)		
Spring rate (R) in N/mm		
Spring travel (Sh) in mm	5.5	5.5
Operating load 1 (F1) in N	2.45	2.45
Operating load 2 (F2) in N		
Operating length 1 (L1) in mm	7.55	7.55
Operating length2 (L2) in mm		
Solid length (Lc) in mm		
Solid Load (Fc) in N		
Mass (M) in g		
Natural frequency (fe) in Hz		

Results

Proposed design characteristics

20.23	DoS (mm)	D1 (mm)	19.60
18.00	DiS (mm)	Small spring index	35.23
23.81	DoL (mm)	D2 (mm)	22.14
21.58	DiL (mm)	Large spring index	39.63
0.56	d (mm)	Transition Length (mm)	9.79
		Transition Load (N)	2.34
60.15	L0 (mm)	Ld (mm)	131.45
4.00	nt	Max helix angle (°)	25.71
0.045407	R (N/mm)		
5.50	Sh (mm)	Axial pitch (mm)	29.7740
2.450	F1 (N)	Radial pitch (mm)	0.6146
2.896	F2 (N)	Fatigue life factor	1.10
7.56	L1 (mm)	Stress at solid (%Rm)	48
2.06	L2 (mm)	Buckling length (mm)	0.00
0.56	Lc (mm)		
3.33	Fc (N)		
1.0182	M (g)		
106.74	fe (Hz)		

Inactive parameters: normal
Active parameters: undefined
Violated parameters: ~~Stroked Thru~~

Figure 29: Conical Spring Design Tool (right) and Conical Spring Behaviour (left) [17]

2.2.1.3 Nonlinear Force by Mechanism

[18] present the experimental study of using an origami-inspired technique to create a constant-force mechanism (OriCFM). The OriCFM design was based on two components, which are a parallel spring and an origami mechanism. Parallel spring provided a linear reaction force, while the origami mechanism provided a nonlinear reaction force; by combining two components, the constant-force mechanism was created. Figure 30 (right) illustrates the mechanism of reaction force and its component.

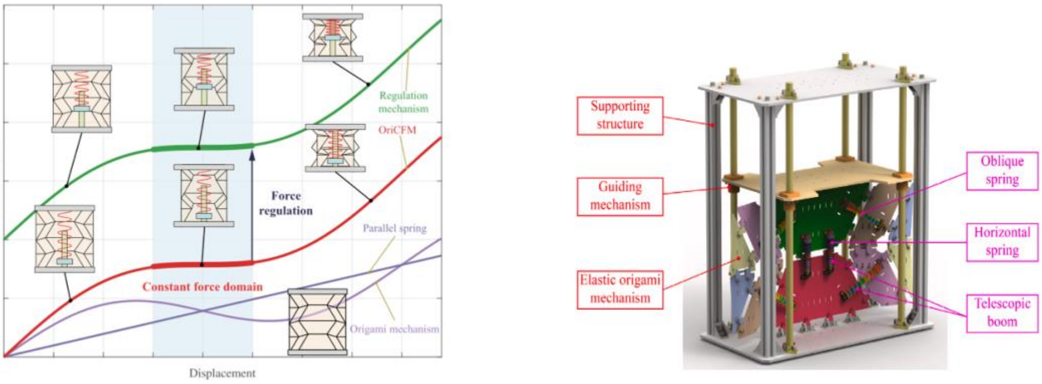


Figure 30: 3D model of origami constant force mechanism (right) and Working Principle of origami mechanism (left) [18]

The origami mechanism was created by analysing the geometric relationship and stiffness of the mechanism. After the initial mechanism was created, the mechanism was optimised according to the output force-displacement function, value of scale factor, folding angle, spring influence factor, and crease angle were optimised to create an optimal origami design as Figure 30 (left). After achieving the optimal design, the prototype was created and tested in 3 different configurations. The difference between the three experiments' setups was the number of horizontal springs (0, 2, and 4). However, the practical experiment result illustrated that the mechanism was not able to fulfil the criteria. The spring has three error factors, which are: 1st was the internal friction of the device, 2nd was the kinematic error that occurred from the paper joint, and 3rd was the gravity effect as the experiment setup. In addition, the possible solutions are listed in the journal paper.

Nonlinear reaction force studies in this literature review are mostly achieved by using compliant mechanisms and other kinds of mechanisms. Constant force design results of all studies have different design goals and materials used, which makes the results incomparable. The material used in the studies includes AL7075[13], ABS[13,14],Titanium[12] and Elastomer[5]. ABS is the most selected material for being used in prototype and testing experiments due to its ease of use and manufacturing as well as the material properties, which makes the material suitable for compliant mechanism. The studies in this literature review also have a major common concept of methodology, which is geometry optimisation. Most of the studies introduced the design optimisation method as one of the key components to achieve the optimal design. Design optimisation can help achieve the optimal design result by using the optimisation algorithm. The algorithms mentioned in this literature review include genetic algorithms and gradient-based algorithms in MATLAB software. Additionally, the software used for design optimisation in the studies are mostly MATLAB, ANSYS, ADAMS, and COMSOL.

2.2.2 Nonlinear Torque-Displacement Output

The nonlinear relationship of the torque-displacement function is another literature review topic selected for this study. The research aim of this study is to design and create a nonlinear torsional spring in order to make the torque-displacement output function designable. The literature review on this topic helps increase the knowledge and understanding of the essential components for achieving the desired result. Nonlinear torque-displacement mechanisms are used in various applications, including human-robot interaction devices, humanlike robots and other applications that require a specific limited maximum amount of output torque. This is because a nonlinear torque-displacement output device can help improve the safety of interacted objects. Various way has been used to achieve nonlinear torque-displacement output. The nonlinear torque-displacement output can be created by using different methodologies, including CAM, magnet, compliance mechanism, spiral spring, and gear system. However, this literature review only includes unique nonlinear-compliant torsional spring designs, leaf springs and spiral springs, which aligned with the aims and objectives of the study.

2.2.2.1 Nonlinear Torque by Compliant Joint

A study by Mei [19] aimed to design and create a compliant joint with variable stiffness by using a leaf branch spring to create a torsion spring. The design goal of the study is to create a high-compliance joint to be used as an elastic element in an actuator. The torsion spring design was created based on the guide rod mechanism by converting single leaf force-deflection into resultant reaction torque, as in Figure 31.

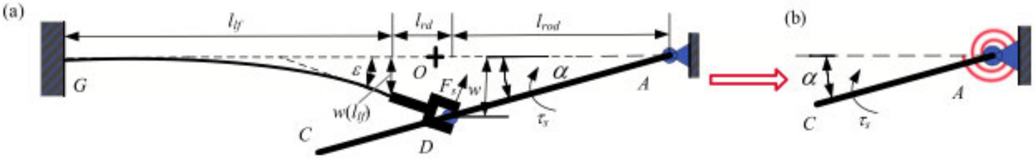


Figure 31: Leaf branch spring-based complaint joint [19]

The spring design method is based on calculation and design parameter adjustment. The principle of planar bidirectional torsion spring design is introduced to create an arrangement of the leaf structure in the torsion spring. The dimension of the leaf structure is considered as a design adjustable parameter in this study, as well as the leaf branch configuration. After the calculation, the final spring design included two components, which are rigid and flexible linkages, as in Figure 32. The spring prototype was tested and compared in terms of output torque and maximum stress. As a result, the maximum stress of the simulation is identical to the result of the experiment. Moreover, The spring can provide a nonlinear torque-displacement relationship, as in Figure 33. While T_{sa} is the simulation data and T_{sn} is the experimental result. The study can include a fatigue test to ensure the performance of the spring and help measure spring lifetime.

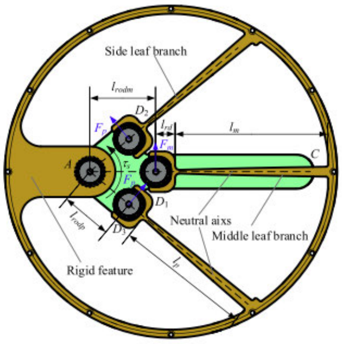


Figure 32: Final nonlinear torsional spring design [19]

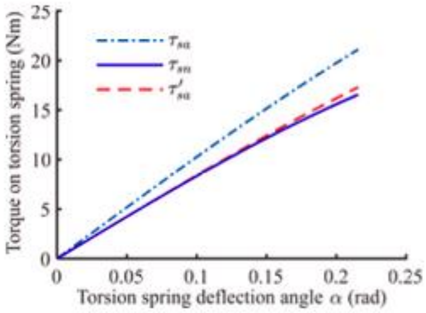


Figure 33: Result comparisons of simulation and measurement results [19]

Another study by Kuo [20] used a different approach to achieve nonlinear torque output. The goal of this study is to create a nonlinear torque output mechanism of a robotic hand to prevent an exceeding amount of force applied to the object. Interestingly, the compliant joint is used as a main component to provide a nonlinear output of this study. The compliant joint was created by using a fixed link with spacers, pulleys, and a compliant material, as in Figure 34. The compliant material was used as a spring to generate reaction torque within the elastic range.

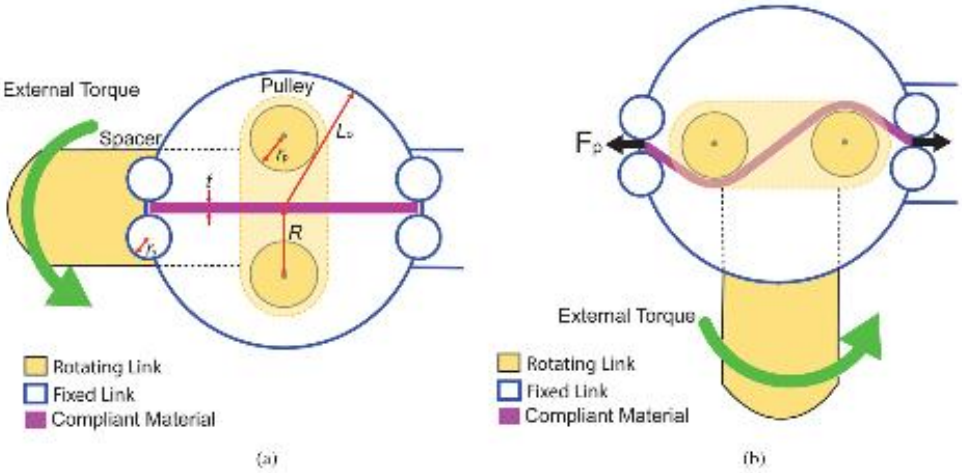


Figure 34: Joint design configuration and movement [20]

The final design of the spring was created by using design optimisation for minimising the root mean square error (RMSE). The error is a difference between the desired torque and simulation torque result value. The design constraints of the optimisation include geometrical constraints of the joint configurations and material maximum compression and the rotational range of this study, which is from 0-100 degrees. After the design optimization, the optimal torsion spring design can generate a nonlinear relationship in the computational simulation, as in Figure 35. Practical experiments are essential to verify the simulation result. The experiment of this study was designed to investigate and compare the performance of 6 different spring materials in four different pulley configurations. The experiment setting includes the DC motor, which applied sinusoidal amplitude in a range of frequencies. The rotational range of the motor is between 0 to 100 degrees at the setting frequency. As a result, differences in nonlinear material properties and pulley configuration result in different reaction torque functions. The output torque of the joint illustrates the nonlinear relationship, as expected in Figure 35. It is worth mentioning that the output torque in the clockwise rotation of the joint was acceptable. However, the output torque in a counter-clockwise direction (configuration 2) had an error due to the change of contact phase between the material and pulleys, while the issue did not appear in any other configuration of the joint.

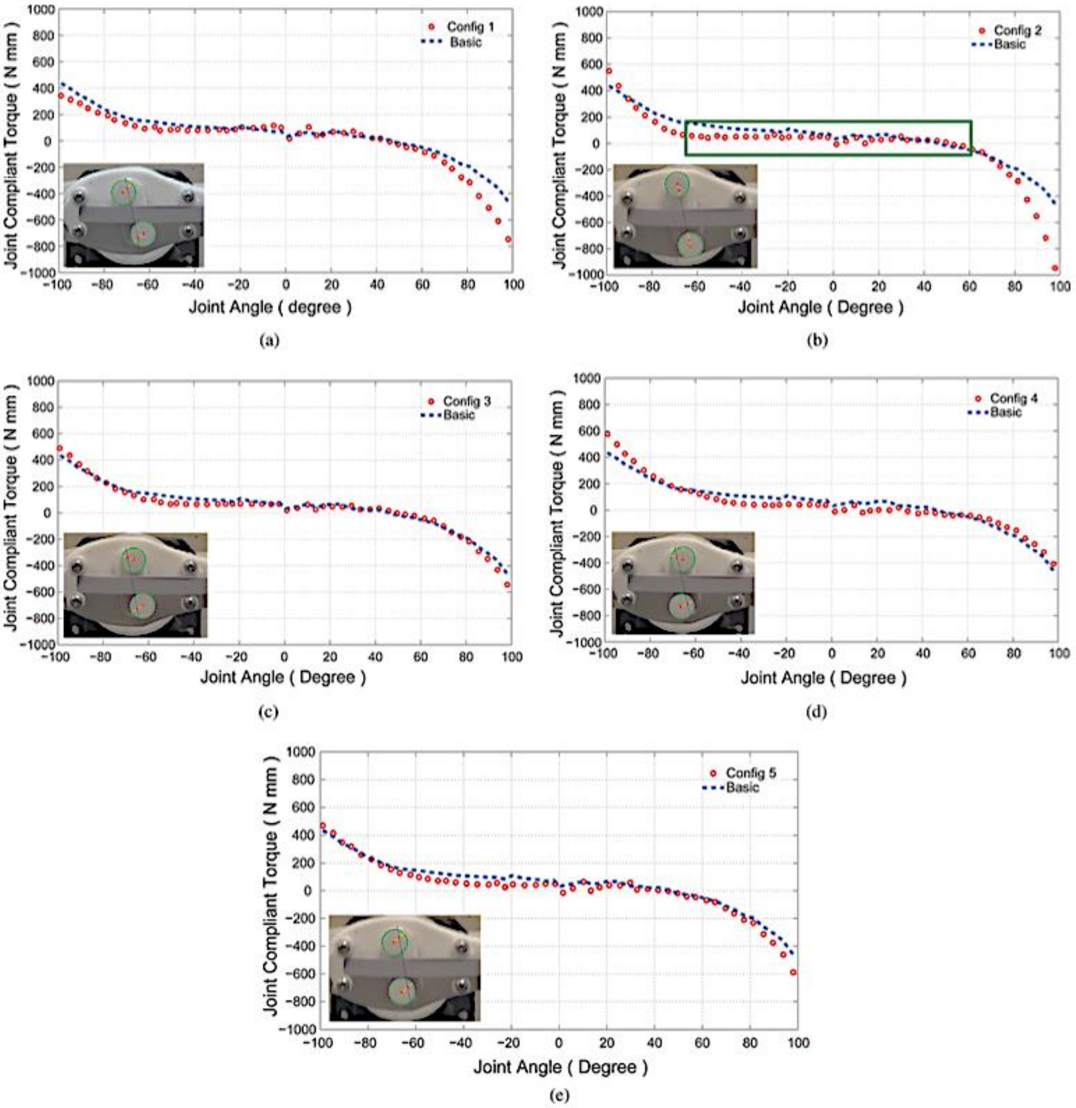


Figure 35: Experimental results in different configurations [20]

2.2.2.2 Nonlinear Torque by Compliant Torsional Spring

A study by Phan [21] focuses on creating constant-torque spring using a compliant mechanism for rehabilitation devices by using a genetic algorithm for shape optimisation. The design of the torsion spring includes a shaft, compliant beams, and an anchor, as in Figure 36. In detail, The compliant beams were created based on Bezier curves, which were constructed by input control points. Moreover, the shape of the curve can be changed by adjusting the control points' position while the positions are determined by the optimisation algorithm. The optimisation objective function was set to minimise the variation of the difference between output torque and torque criteria, while the optimization parameters include the position of control points (Bezier curve shape) and in-plane thickness of the spring. The optimisation design constraints included maximum stress, upper and lower bounds of the control points' position, as well as the design thickness. After achieving the optimal solution from geometry optimisation, the optimal spring design was tested using FEA simulation by using POM as a material. POM is a high-performance polymer used in many applications in the industry. The spring was able to provide up to 63 degrees of rotational range while the constant force range begins from 20 to 63 degrees (43-degree constant force range), as in Figure 36. The size of the optimal spring was 200 x 200 mm. In addition, the study FEA test included the equivalent stress for the thermoplastic material. However, the study did not provide other details on the maximum yield strength or specific material properties of the POM and a practical experiment was not included.

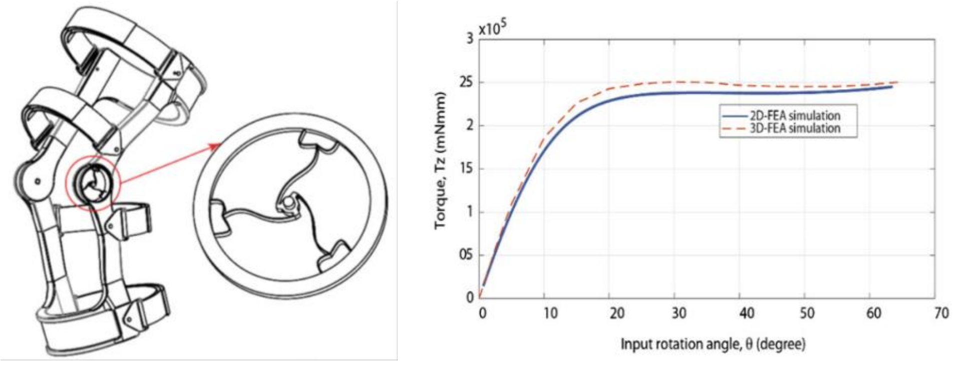


Figure 36: CTM design concept (left) and Torque-rotation results of CTM (right) [21].

Another study by Wang [22] also uses a similar methodology to create a constant-torque torsion spring. The spring includes a set of bistable beams, an anchor and a spring shaft similar to the previous study. However, the optimisation parameters of this study are the parameters of the beam, including length, thickness, width, and theta, which represent the length, in-plane width, out-of-plane thickness, and inclined angle respectively. The torsion spring design is illustrated in Figure 37. After achieving the optimal design of the spring, the prototype of the torsion spring was created and tested for two aspects, which are maximum stress and constant-torque performance, compared to the simulation results. In this study, ABS-plus was selected as a material for prototype production. The torsion spring prototype was able to generate 3 mm. constant-torque motion, which is lower than the simulation result (4.00 mm). The spring was able to perform an equal amount of angular displacement without plastic deformation, as in Figure 38. The cause of lower constant-torque performance occurred by the fabrication tolerance of the 3D printer, which can be solved by using other methods in prototype production such as metal casting, injection moulding and water cutter.

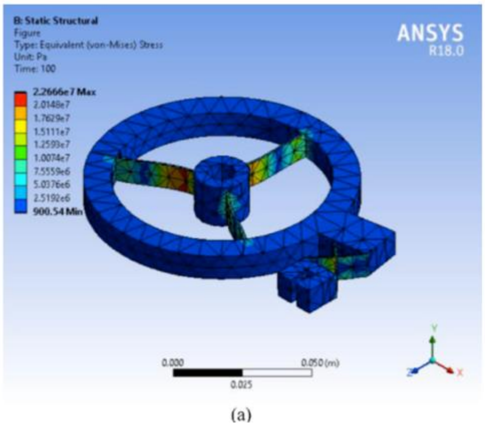


Figure 37: FEA result of optimal spring design [22]

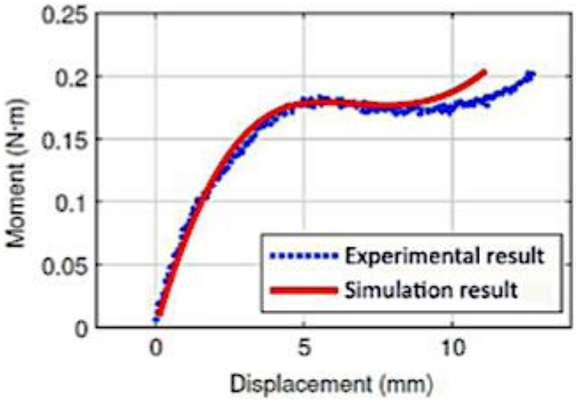


Figure 38: Comparisons of simulation and experimental results [22]

A similar study by Hou [23], the study aimed to create a joint mechanism that can provide constant-torque output. The design application of this study included dynamic and static balancing of machines and human-robot interaction devices such as joint rehabilitation devices and mobility assistive devices. The design criterion of this study is to maximise the constant-torque output region provided by the torsion spring without having spring plastic deformation. Similar to the previous studies, the torsion spring includes three main sections, which are the spring shaft, anchor, and compliant beams. The compliant beams in this study were constructed based on the design schematic in Figure 39.

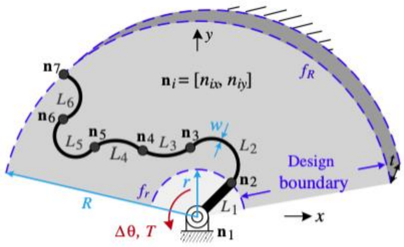


Figure 39: Compliance model schematic [23]

Design optimisation was used to help achieve the spring optimal design in this study. The optimisation goal was to maximise the constant-torque output region. The design constraints of the optimisation problem included the radius of the arc segment, the angle between the segment, node position, optimisation space and maximum stress of the compliant segment. The optimal design of the compliant segment was obtained in Figure 40. However, for ease of mass production, the design was simplified in the section between segments 3 and 4 as a straight line, as in Figure 41. The maximum footprint of the optimal design was 60*60 mm.

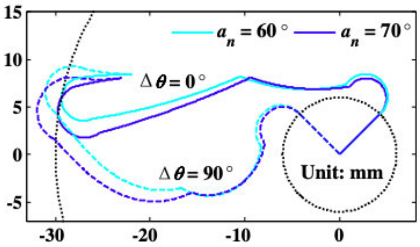


Figure 40: Optimal design and deformation shapes (type I) [23]

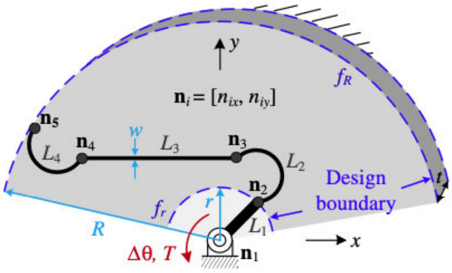


Figure 41: Optimal design shape (type II) [23]

After the model simplification, another optimisation was performed to ensure that the optimal result was obtained. The optimisation objective function of 2nd optimisation remains unchanged as the previous objective function while the design constraints were changed due to the model adjustment. After the optimal design was obtained, FEA analysis of the optimal design was conducted by using three different materials, including POM, PEEK and PEEK-GF30. After the FEA verification, the optimal design was created as a prototype for experimental verification. As a result, the torsion spring can provide constant-torque output in all types of material while generating a 50 degrees of rotation range of constant-torque output as in Figure 42. However, the flatness of the constant output region varied in different types of material, as in Table 3.

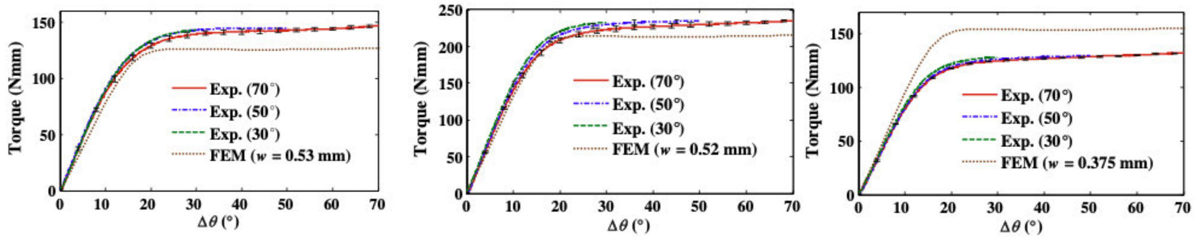


Figure 42: Experimental Torque-Theta curve of POM (left), PEEK (middle) and PEEK-GF30 (right) [23]

Table 3: Comparisons of experimental vs computational results in 3 different materials [23]

Material	T_c (Nmm)		Experiment flatness
	Experiment	FEM	
POM	141.812	126.268	87.67%
PEEK	227.205	213.942	88.52%
PEEK-GF30	127.058	153.812	89.10%

The difference in material selection does not only affect the flatness of the constant-torque region during the experiment but also the output error during the unloading state. Figure 43 illustrates the loading curve vs unloading curve of the torsion spring with three different material types. As a result, PEEK-GF30 has the lowest unloading error compared to other materials.

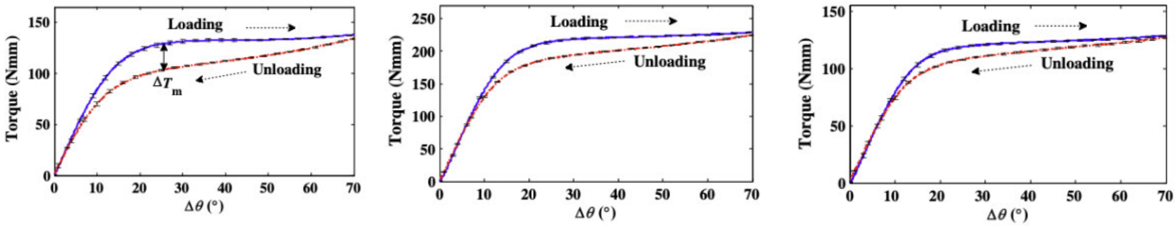
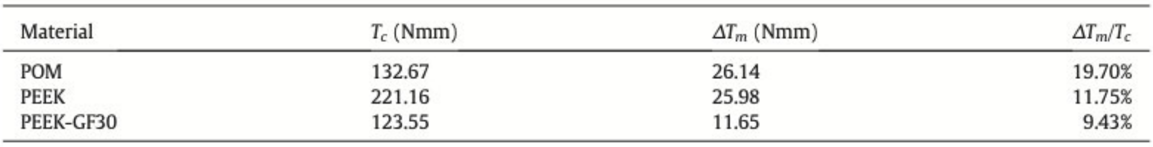


Figure 43: Experimental result of loading vs unloading torque in 3 different materials.

[POM (left), PEEK (middle), and PEEK-GF30 (right)] [23]

Table 4: Comparison of polymer hysteresis [23]



Material	T_c (Nmm)	ΔT_m (Nmm)	$\Delta T_m/T_c$
POM	132.67	26.14	19.70%
PEEK	221.16	25.98	11.75%
PEEK-GF30	123.55	11.65	9.43%

Moreover, a Stress relaxation study of torsion springs with three different materials was also included in this study. Polymer material can suffer from stress relaxation over time, which can affect the performance of torsion spring output. The experiment setup included applying a rotation of 0-70 degrees with a fixed inner shaft at 70 degrees. The output torque value of the spring was recorded every 5 seconds until the change of torque was unnoticeable. As a result, after 4 hours, PEEK-GF30 has the best performance in terms of stress relaxation at a 5.45% decline rate. The study also illustrated that the constant-torque property is insensitive to some parameters such as in-plane thickness, Poisson's ratio, or modulus variation. Moreover, the experiment also shows the essence of practical testing of each material, which was shown in both hysteresis and unloading experiments.

Another constant-torque torsion spring design study by Prakashah [24] focused on creating a torsion spring, which is similar to the previous studies. This study introduced geometric optimisation to create a compliant module (curve beam) of the torsion spring between the ring and shaft section, as in Figure 44. The compliant models were created by using a cubic spline curve expression with five adjustable control points.

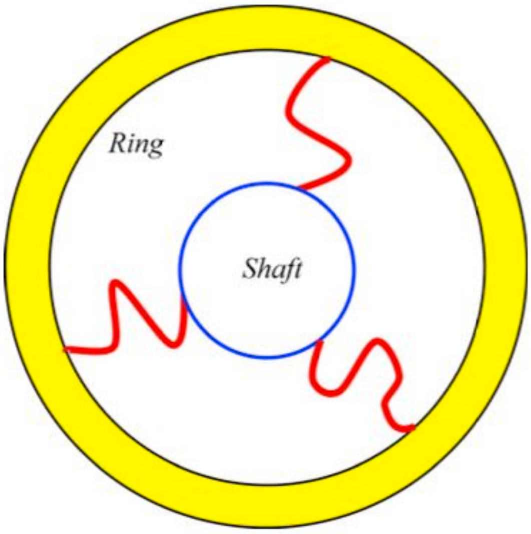
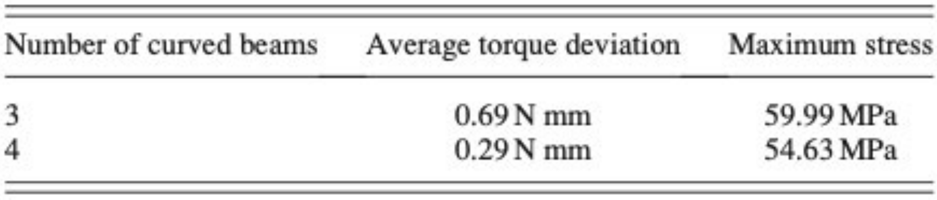


Figure 44: Spring design configuration [24]

Table 5: Result comparisons between three and four curved beams [24]



Number of curved beams	Average torque deviation	Maximum stress
3	0.69 N mm	59.99 MPa
4	0.29 N mm	54.63 MPa

The programs used for the design optimisation in this study included ANSYS APDL and MATLAB. ANSYS APDL software was used for FEA to determine the torque reaction function and maximum stress of the model, while MATLAB was used for optimising geometry and determining the viability of the model. Global optimisation was selected as an optimisation algorithm, and its parameters in this study include 12 independent parameters designed to represent the curve beam. After the optimal compliant model was obtained by the optimisation process, the model was used for further analysis in terms of the relationship between the number of curve beams and the performance of the constant-torque spring. The torsion spring with 3 and 4 constant-torque-compliant mechanisms was created for the analysis. Both torsion springs have dimensions of 80 x 80 mm. As a result, the torsion spring with four curve beams performed better in terms of average torque deviation while having a lower amount of maximum stress, as in Table 5. Finally, the optimal torsion spring design with three curve beams was created as a prototype by using a 3D printer with a 600 dpi solution. The material used for the prototype was engineering plastic. The spring prototype was tested by using a torque screwdriver, as in Figure 45. The optimal spring's torque function is presented in Figure 46, which verifies the performance of the experimental result vs computational simulation result. As a result, the prototype can provide a constant-torque output with some deviation. However, the experiment cannot provide maximum stress at maximum rotation or performance in multiple uses.

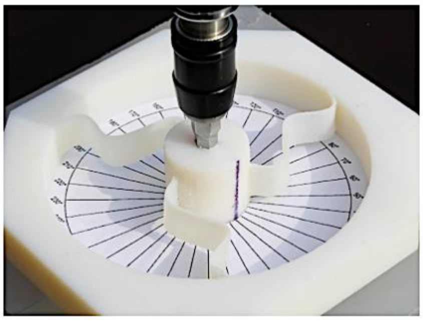


Figure 45: Torsional spring experiment setup [24]

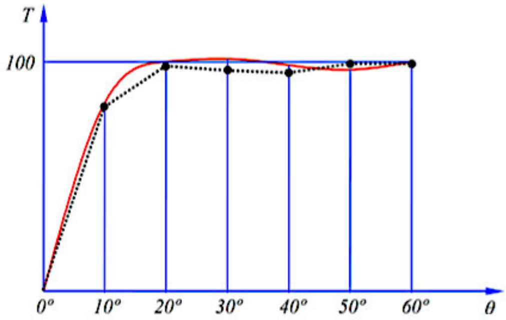


Figure 46: Torsional spring experimental result vs computational analysis result [24]

The study by Reddy [25] aimed to develop the constant-torque spring to another level. The generic constant-torque torsion spring, such as in [24], can only provide constant reaction torque in positive rotation. However, [25] aimed to develop a torsion spring that can provide constant reaction torque in both positive and negative rotation by using a similar design methodology as in [24]. This constant-torque torsion spring can be called "The synthesized bidirectional constant-torque spring". The main components of the torsion spring in this study included a ring, shaft, and curve beam, as in the previous study. However, the curved beam in this study includes two different sets of designs for four beam curves as in Figure 47. The curve beam designs were created by eight control points as shown in Figure 48.

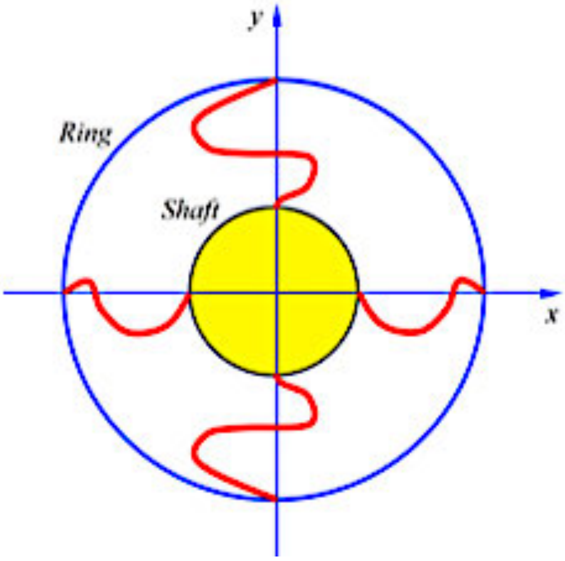


Figure 47: Bidirectional torsion spring configuration [25]

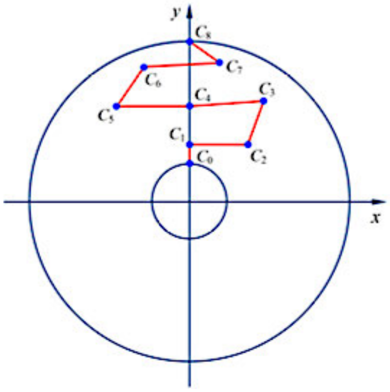


Figure 48: Control points of compliant components [25]

Another difference in the methodology of this study is the geometry optimisation method. The beam design control points were optimised by using the ANSYS optimization toolbox directly without using the MATLAB program for optimisation. The area constraint of the torsion spring optimisation was 100 x 100 mm. After achieving the optimal result by optimisation, the optimal design was created as a prototype for a practical experiment to determine the performance of a constant-torque torsion spring compared to the simulation result. The material used for the prototype was engineering plastic. The prototype of a bidirectional constant-torque spring can achieve constant-torque results in both positive and negative directions with the same amount of resultant torque and displacement. However, similarly to the [24], the practical experiments for maximum stress, plastic deformation, fatigue test and material comparison were not included in this study. The experiments can help determine the performance of the bidirectional constant-torque spring in long-term usage.

Nonlinear reaction torque designs in this literature review are mostly based on the torsional design with three sections, which are shaft, compliant mechanism, and ring, which makes compliant mechanism design of the spring the only optimisable component in each study. Another common aspect of the studies is the optimisation process. All the studies in this nonlinear reaction torque design literature review introduced an optimisation method for compliant mechanism geometry optimisation to achieve the result. However, 2 of the studies introduced different approaches to achieve the nonlinear reaction torque result. [13] introduced leaf spring to create a unique compliant joint, while [20] introduced compliant material as an elastic unit to generate torque. Another interesting aspect is the material selection. POM was the most selected material in this nonlinear reaction torque literature review, while PEEK, PEEK-GF30 and ABS-plus were only selected once. In addition, [24] and [25] use engineering plastic as a material but without specifying the material type. Moreover, the study [23] conducted experiments that others did not, such as the effect of material properties on design performance, difference in loading and unloading reaction torque, and polymer hysteresis. The experiments can help ensure optimal performance in different materials and long-term usage.

This literature review explored studies related to nonlinear compliant torsion spring designs, which can be divided into three sub-sections, including compliant torsion spring design, nonlinear reaction force design and nonlinear reaction torque design. Three different subsections illustrate the methodology and experiments required to verify the performance of the spring and material selection of each study. The common design method in this literature review is geometric optimisation, which is the key tool for finding an optimal result that has better performance than general calculation only in some cases due to the complexity of the calculation. Another common design method found was compliant mechanism design. Compliant mechanism design is used for creating the optimal design in most of the studies included in this literature review. The compliant mechanism can help obtain the desired force and motion by using the flexibility of the mechanism parts. The powerful design method was created by combining geometry optimisation and a compliant mechanism. Another common aspect of this literature review was the prototype production. Most of the prototypes in this literature review were created by using a 3D printer due to its convenience, which leads to a limited type of material selection. Polymer materials were the most selected type of material, which includes POM and ABS. However, according to [22], using a 3D printer can cause a decrease in the performance of the prototype due to the resolution of the 3D printer. It is worth mentioning that some studies include experiments in other aspects that are different from others. As mentioned previously, [23] conducted experiments to verify the performance of the optimal design, including a difference in loading-unloading torque, which can ensure the amount of reaction force in both stages; polymer hysteresis, which helps determine the decrease in performance over time in some of the material types and the difference in performance in different material used. It is worth mentioning that none of the studies perform fatigue tests even though the compliant mechanism has a higher risk of fatigue failure than other types of mechanisms. However, some of the studies include safety factors as their standard, which can help prevent fatigue failure and other types of failure at the same time. Interestingly, none of the nonlinear spring designs focused on gravity compensation application, which requires the cosine function of torque displacement that is categorised in nonlinear spring. Moreover, most of the torsion spring designs have similar spring configurations, which results in a limited number of possible solutions of the spring.

Chapter 3: Methodology

This research aims to create a torsion spring for gravity compensation (cosine-torque spring) and biomedical (constant-torque spring) applications. In this research, the spring designs were created by integrating design optimisations and the finite element analysis method (FEA). The integration method is required due to the non-availability of a generalised design methodology for compliant spring design. The methodology in this study begins with design objectives, followed by methodology overview, design scope and specifications, optimisation methods, genetic algorithm optimisation setting, and lastly, the process to achieve optimisation result and double optimisation.

3.1 Design objectives

The design objective of this study is to create a torsion spring that can generate constant-torque and cosine-torque output. The overall specifications of the optimal spring design are listed below.

1. The spring must provide a specific reaction torque.
    a. Constant-torque
    b. Cosine-torque
2. The minimum rotational range of the spring must be at least 90 degrees.
3. The spring must reach maximum rotation without performing any plastic deformations.

The spring specifications are introduced based on the general requirement of gravity compensation and biomedical applications as in the literature reviews, which will be discussed in this chapter.

3.2 Methodology Overview

The methodology of this study can be simply transferred into the diagram as in Figure 49. The methodology includes two optimisation steps, which are Geometric optimisation and Scaling optimisation. Geometric optimisation aims to optimise the optimal shape of the B-spline curve, which can provide a specific desired output. On the other hand, Scaling optimisation aims to adjust the size of the optimised shape from Geometric optimisation to adjust both maximum stress and output torque. Each optimisation process of this study includes five primary elements, which are Design optimisation, B-spine generation, 3D geometry creation, Finite

Element Analysis (FEA), and Design Verification. The overview of the optimisation process is shown in Figure 49, while red represents the MATLAB software process, and yellow represents the ANSYS software process.

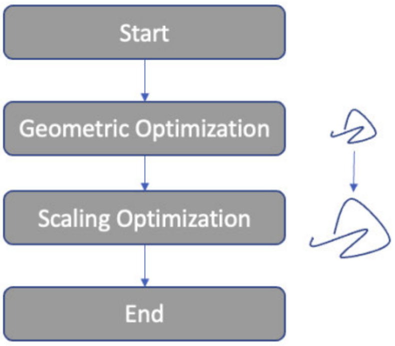


Figure 49: Spring Design Methodology Overview

MATLAB software is used in three different tasks for this study, which are design optimization, B-spline generation, and design verification. The design optimisations are performed using MATLAB optimisation algorithms and determining the optimisation setting such as design parameters, design constraints and objective function. In addition, MATLAB software is used for pre-processing and post-processing. Pre-processing includes B-spline generation, which not only creates a B-spline shape from the input control points but also ensures that the generated B-spline shape is suitable for the optimisation process. On the other hand, ANSYS software is used for both 3D geometry creation and Finite Element Analysis. After passing the suitability check from MATLAB, the B-spline curve is transferred to SpaceClaim software to create a 3D geometry of the torsion spring and passed on to Workbench software. Workbench software will perform a FEA on the geometry and send the results, which include maximum stress and output torque to MATLAB for Design verification. The design verification process validates the suitability of each design in terms of maximum stress, size, and output accuracy of the spring in a different factor for each criterion. The score of each design is sent to the optimisation algorithm to perform design optimisation and further analysis.

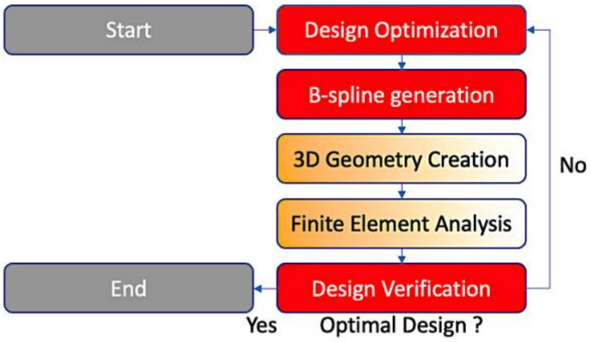


Figure 50: Design Optimisation Overview

3.3 Design scope and specifications

The optimisation algorithm requires proper design constraints and optimisation goals to be able to achieve the optimal result without excessive time consumption. This study's design scopes and specifications were created based on the design objectives of constant-torque and cosine-torque applications. Setting the design scope and specification can help limit the possible amount of the optimisation population and ensure the final design's suitability for usage in each application.

3.3.1 Spring Maximum Dimension

Spring size is one of the main criteria of this study. Minimising the spring's size can help reduce the machine's overall footprint and help create a compact and lightweight design. For example, a generic sprint gravity compensation device requires multi-springs configuration and other components such as a gear system and pulley, and excessive components can result in the bulky and excessive weight of the device. B-spline-based spring can reduce the amount of components required in the gravity compensator and the overall system's size. In this study, the maximum dimension of the length and width of the spring are limited to 500 mm. while the cross-sectional area is limited to 5 mm^2.

3.3.2 Spring Material

Material properties of the selected material must be suitable for the application used for the reliability of the springs. As in the design objective, the spring must be able to provide a rotational range of 90 degrees. In order to achieve such a criterion, a suitable material must be selected to ensure that the spring can perform large deformation without exceeding the material’s yield strength. The material selection of this study is based on literature reviews and material availability. Structural Steel and POM are the candidates for the material selection in this study. Structural Steel was chosen for the material because of its material properties and accessibility of the material. On the other hand, POM was not selected for the material due to the downside of using a 3D printer as a production method, which is unreliable for this design. In detail, Additive manufacturing can cause design inaccuracy as well as uneven distribution of material between each layer, which can reduce the performance of the specimen.

3.3.3 Viable Range of the Result

The viability of the result is another factor that requires to be set as a criterion. The viable range of the result can be different in each design generated. The optimisation goal is to maximise the viable range of the result for achieving the highest performance design possible. For example, as in Figure 51, an example torque output plot is illustrated. The result's viable range is from 0.55 to 0.875 seconds, which is 32.5 percent of the maximum displacement. This criterion only applies to the constant-torque design, while the cosine-torque design will only be validated by the reaction torque error (section 3.6.1.1).

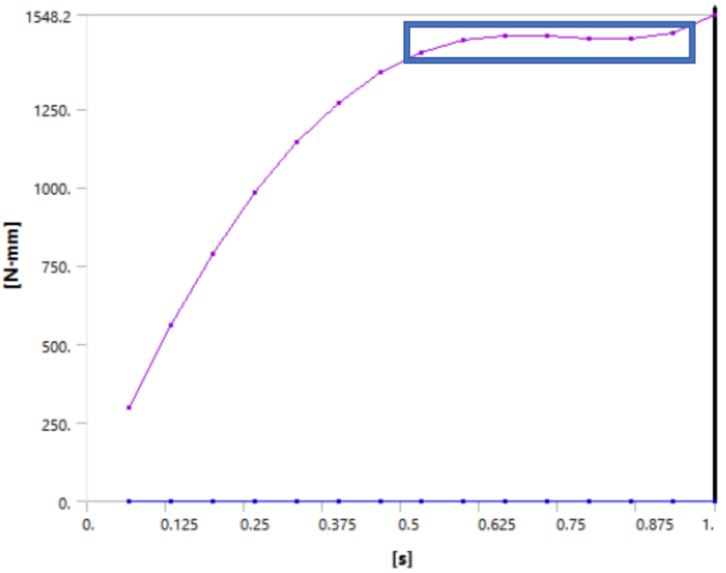


Figure 51: Viable Range Result Example

3.3.4 Reaction torque error limit (Design Flatness %)

The imperfection of the output sometimes occurred in the optimisation results, such as the fluctuation of the generated reaction torque function of the spring. The error limit is the limit of the result fluctuation, which is set as a criterion to ensure the quality of the optimisation result. The fluctuation limit of this study is set to a maximum of 5% of the steady-state value. The design flatness can be calculated as the expression below. On the other hand, the cosine-torque spring reaction torque error limit will be calculated by the "cosine function value", which is the same function used for identifying the suitability of the design during the optimisation process. The CFV is calculated by the difference between the ideal cosine plot and the plot of the design output, which will be explained in the cosine-torque objective function section.

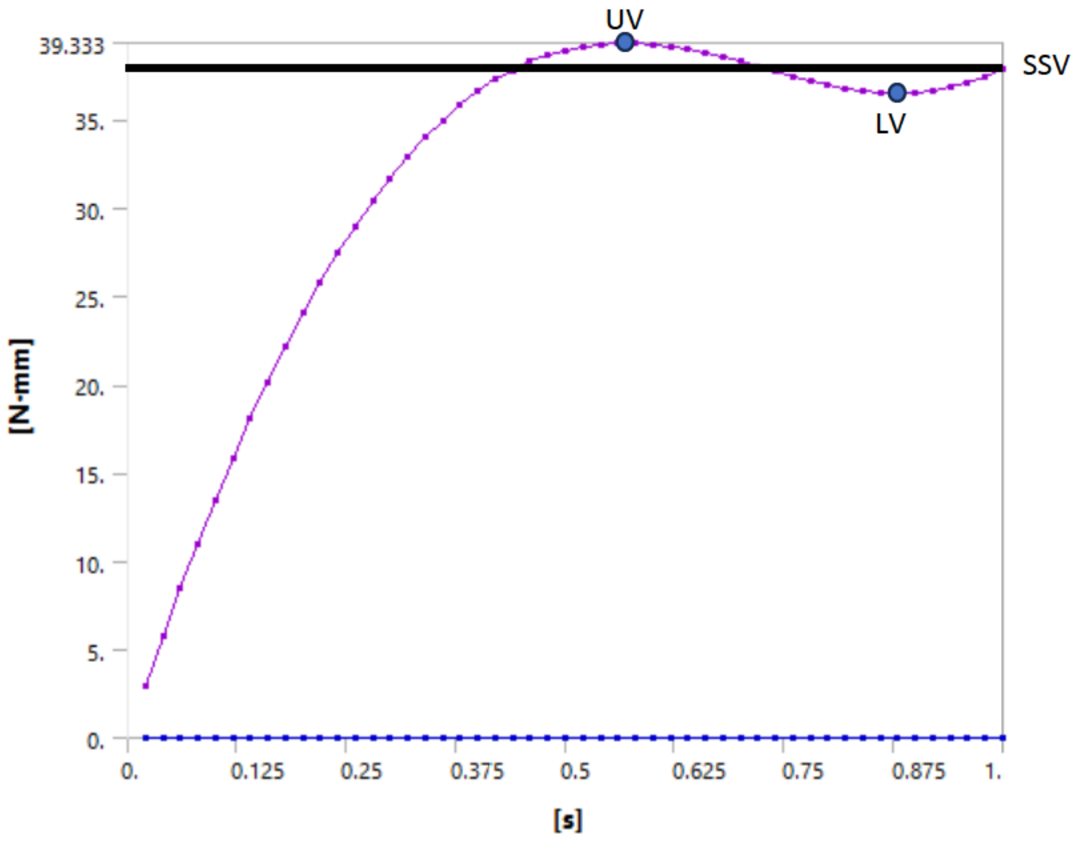


Figure 52: Design flatness result calculation example

$$Steady\ State\ Value\ (SSV) = 2\ (Upper\ Value\ ) - Lower\ Value \tag{3.1}$$

$$Design\ Flatness\ (\%) = \frac{Steady\ State\ Value - |\ maximum\ error\ value|}{Stready\ State\ Value} \tag{3.2}$$

Note: Maximum error value is the largest error value from SSV, which can be either UV or LV.

3.4 Geometric Optimisation

The genetic algorithm was used for the optimisation to find the most optimal geometry of the spring, which can provide a reaction torque as each criterion. The details of the genetic algorithm will be discussed below.

3.4.1 Genetic algorithm (GAs)

A genetic algorithm is one of the optimisation methods created after the natural biological selection and adaptation process [26]. The disadvantage of the GAs is the computational time, which is longer than other methods due to its nature. Not to mention, GAs was built as a non-gradient-based optimisation, which is different from other optimisation methods. However, in this study, such a non-gradient-based optimisation has made GAs a more suitable optimisation method than other methods available. This is because GAs can find the optimisation pattern in their unique way, which is better at avoiding reaching the local minimum than gradient-based optimisation in this case. In detail, GAs starts with multiple random initial designs, which can be called chromosomes. In each optimisation iteration (generation), the set number of chromosomes or population is generated using three processes, which are selection, cross-over and mutation process. The optimisation parameters and other aspects will be discussed in the next section.

3.4.1.1 Chromosomes

In this study, each of the spring designs is represented by a single unique chromosome. A single chromosome includes multiple variables (depending on the design input) that determine the topology and shape of the spring design. For example, the chromosomes are the value of control points and maximum pi in the design model generation function of constant-torque torsion springs. The optimisation algorithm will adjust the value of control points and maximum pi to create an optimal design in each iteration.

3.4.1.2 Initialization

GAs optimisation of this study begins by creating a population of 100 to 500 unique chromosomes. The chromosome represents each unique spring design. The amount of population was set based on the optimisation trials of this study, which were conducted to investigate the most suitable population amount. The aim of the trial setting is to use a genetic algorithm to find the result in a different amount of population from 100-1000 chromosomes in each generation. The population with 100 and 500 chromosomes is a suitable amount for the algorithm to achieve the global minimum while minimising the optimisation time compared to other population values. During the optimisation process, each unique design must be verified for design suitability for the 3D geometry creation process, which will be done by SpaceClaim software before passing on the geometry to ANSYS for optimisation. The design suitability criteria include “Design Overlapping”, “Design Minimum Angle”, and “Design Minimum Distance”, which will be discussed in detail later in this chapter. The designs that fail to satisfy the criterion are ranked in last place and considered as the least optimal design.

3.4.1.3 Elite Selection, Selection, Cross-over & Mutation

After all the designs are evaluated and checked with suitability criteria in each iteration, the designs will be sorted from lowest (best) to highest (worst) score (Objective Function Value) by using the objective function as a criterion for the value. The process is necessary to select the elite design in the population of the current generation to create a new generation. The first step of post-processing is Elite selection – ten of the designs with the lowest score are selected as an elite of the generation, and these designs are passed on to the next generation without any processing. The remaining designs are evaluated through a selection process that selects the fittest group of the design with a low objective function value. Thenceforth, the group of the selected design are selected as the base chromosome for the next generation. New generation chromosomes are created by mixing two of the selected design chromosomes in a ratio that can create a new design; such a process is called Cross-over. This Cross-over process is repeated until the population of the next generation reaches the limit. Finally, the Mutation process can mutate the value of the chromosome to make slight changes to the design. The mutation process is necessary for this optimisation to help avoid local minima and reach the best result possible. This process has a low chance of occurring and is only possible during the Cross-over process.

3.4.1.4 Design parameters

In this study, design parameters are a set of values of the chromosome that represent aspects of the spring design. Each parameter can be changed and optimised within the range called “design constraints”, which will be mentioned in the next section. The design parameter in this study is constructed based on the design model generation method. The design parameters of the first b-spline generation function contain 22 variables. On the other hand, the design parameters of the second b-spline generation function contain seven variables. The details of the design parameters of each spring design will be discussed further in the B-spline Generation section.

3.4.1.5 Design Constraints

The design constraints are necessary for limiting the possibility of inputs, which can help lower the amount of computational time. Without the design constraints, the possibility of inputs is infinite, and as a result, the computational time will be more likely to increase exponentially. The design constraints of this study are different based on the design goal. The first b-spline generation function includes two constraints, which are the size limit and spring design constraints. On the other hand, the second b-spline generation function includes only size constraints. The details will be discussed further in the B-spline Generation section.

3.4.1.6 Objective Function

The objective function in this study is a function that represents the performance of each design. The objective function also affects the computational time to achieve the optimal result. Without a suitable objective function, the result may result in divergent optimisation, unviable results, and excessive computational time. This study has three objective functions for three different design goals, including constant-torque design and cosine-torque design.

3.5 Constant-torque Design Objective Function

The design objective of a constant-torque spring is to create a spring that can provide 90 rotational degrees while being able to generate constant reaction torque over 45 rotational degrees. The maximum stress of the design must be minimised below at least 1.2 times the yield stress of the design material as a safety factor. The constant-torque torsion spring objective function includes three main elements, which are Function Error Value (FEV), Maximum Stress Penalty (MSP) and Design Suitability Criteria (DSC). The constant-torque torsion spring object function can be written as below.

$$Objective\ function\ (CTTS)\ =\ min\ (FEV + MSP + DSC) \tag{3.3}$$

3.5.1 Function Error Value (FEV)

Function Error Value (FEV) is the sum of the difference between each point throughout the output function. FEV is calculated by using the evaluated results from FEA analysis. The result used for this element is the torque-rotational displacement function, while the maximum rotational angle is constant throughout the optimisation.

The function error value is measured from the difference of each point of the torque-rotational displacement function plot. Instead of measuring every point of the function, only five specific points of the function are selected to help reduce computational time. These points are chosen by selecting the suitable optimisation region of the function to ensure the constant-torque result can be obtained. As illustrated in Figure 53, the initial value from each point is measured and shown as L1 to L5 (Local Value). Each point has its own angular displacement value, which ranges from 30% to 100% of the maximum rotational angle. The goal of the function error value is to minimise the difference between L1 to L5, which results in constant-torque output at the optimal solution. L_{os} (Overshoot Value) is a point of the function which represents the maximum overshoot that occurred. This value also reduces the amount of overshoot in the design.

FEV includes three components, which are Local Error (LE), and Overall Error (OE) and, Overshoot Error (OSE). These three factors have a significant role in the optimisation process to help achieve constant-torque results. FEV can be written as an expression as in the equation below.

$$FEV = LE + OE + OSE \tag{3.4}$$

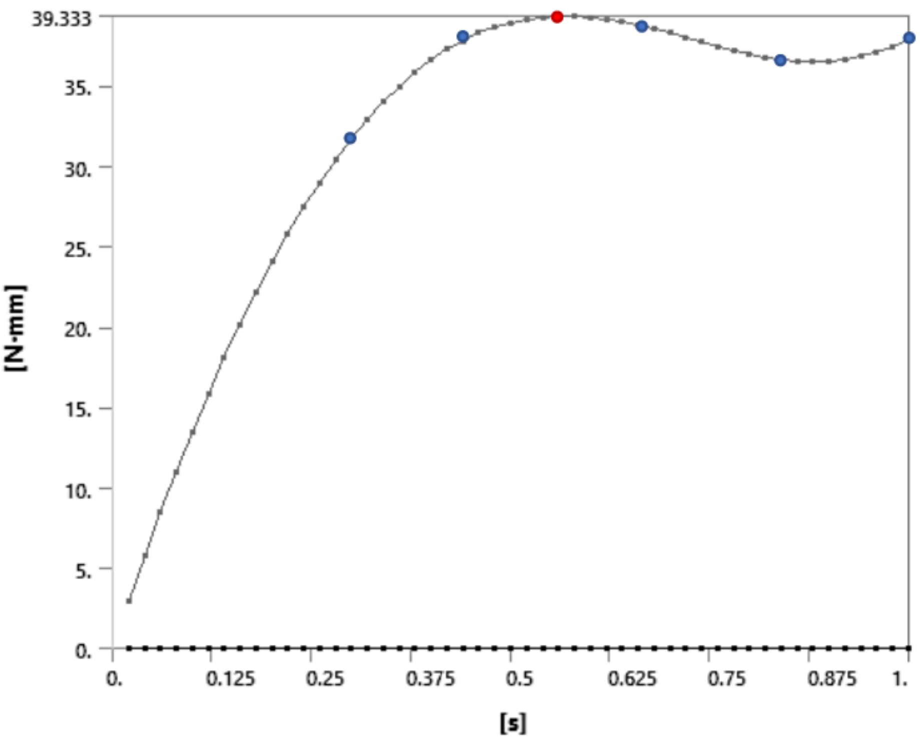


Figure 53: FEA result with L points (L1-L5(blue) L_{os} (Red))

3.5.1.1 Local Error Factor (LE)

Local error factor (LE) represents the sum of the average error value between each two local values such as L1-L2, L2-L3, and so on. LE is created to ensure that error between each L point are minimised by the optimisation, which can be written as in the equation below. However, the optimal result cannot be obtained by using only LE. Nevertheless, at the same time, a fluctuation and an overshoot of the function can still be seen. This occurrence is due to the fact that the overshoot and fluctuation parts are not always taken into account in the objective function criteria.

$$LE = \left(\sqrt{\frac{\sum_{i=1}^{i_{max}=5} \left(L(i) - L(i+1)\right)^2}{i_{max}}} \div \sqrt{\sum_{i=1}^{i_{max}=5} \left(L(i)\right)^2} \right) \tag{3.5}$$

3.5.1.2 Overall Error (OE)

To solve the issue mentioned earlier, Overall Error (OE) is created to represent the error between the overall local values. OE is calculated by using the average mean square value of the function, which can be written as the equation below. OE can ensure that not only the error value between local points but also the overall values are minimised during the optimisation. In other words, a fluctuation can be minimised using OE. However, on the other hand, the torque-angular displacement result of the optimal design still contains some overshoots, which depicts that OE is still lacking in minimising overshoots.

$$OE = \left(\sqrt{\frac{\sum_{i=1}^{i_{max}=5} \left(L(i) - L(5)\right)^2}{i_{max}}} \div \sqrt{\sum_{i=1}^{i_{max}=5} \left(L(i)\right)^2} \right) \tag{3.6}$$

3.5.1.3 Overshoot error (OSE)

Overshoot often occurs in the FEA, resulting in many designs created by the optimisation loop. Even though the overall value is minimised during the optimisation, the overshoot does not always get optimised. The optimisation local value L1 – L5 are not enough to ensure that the optimal result has no overshoot, since the overshoot may not occur at the local value position. Hence, OSE is created to ensure that the value at L1 – L5 and OSE have the most similar value at the end of the optimisation process. OSE is calculated by finding the average mean square between L_{os} and L(i) from L1 to L5, which can be written as in the equation below.

$$OSE = \left(\sqrt{\frac{\sum_{i=1}^{i_{max}=5} (L(i) - L_{os})^2}{i_{max} + 1}} \div \sqrt{\left(\sum_{i=1}^{i_{max}=5} \left(L(i)\right)^2 \right) + L_{os}^2} \right) \tag{3.7}$$

Moreover, combining three criteria can improve overall optimisation performance by providing a correct and higher quality of the objective function value, which can help to reduce computational time. The designs that get approved to pass into the next generation are the only high-performance designs in all perspectives.

3.5.2 Maximum Stress Penalty (MSP)

A maximum stress penalty is added to the objective function to ensure that the optimal design result has the lowest maximum stress possible. The maximum stress penalty is essential for the objective function due to the limited yield strength of the material. This study requires a suitable design with a maximum stress lower than the yield strength to ensure that the design can maintain its shape after deformation, as well as reduce the risk of fatigue failures in the spring. The value of the maximum stress penalty is calculated by using the maximum stress value of each design. The value is used to calculate the suitable penalty for each design, which can be written as an equation as below.

$$MSP = \begin{cases} 0, & \sigma_{max} < \sigma_{yield} \\ \Delta\sigma \times K, & \sigma_{max} \geq \sigma_{yield} \end{cases} \tag{3.8}$$

$$\Delta\sigma = (\sigma_{max} - \sigma_{yield})/\sigma_{max} \tag{3.9}$$

With this maximum stress penalty equation, the optimal design is promised to have its maximum stress value lower than the yield strength of the design. On the other hand, the design with excessive maximum stress will still be included in the population but with a higher objective function value. This is because some of the designs with higher MSP may have better results in other aspects, such as Function Error Value. With further optimisation, it is possible that the maximum stress of those results can be reduced later during the optimisation process, or the result will be used in a cross-over process to produce better results in the next generations.

Having more than a single aspect for calculating the performance of the design can create complexity in approaching the real design goal. Constant K can help sort out the priorities of the objective function components, with the higher amount of value representing the higher level of significance in optimising the component. Constant K for MSP is set to 0.5 compared to FEV, which is set to 1 to ensure that the objective function value represents the result of the design output function more than maximum stress.

3.5.3 Design Suitability Control (DSC)

Design suitability is one of the parameters of the objective function. DSC has a main role in ensuring that the design generated from the optimisation loop is suitable for the ANSYS SpaceClaim program. In detail, the cubic B-Spline curve generated from the optimisation loop from MATLAB requires verification to ensure that it is possible to be used to create a 3D model using ANSYS. In order to pass the B-Spline curve to ANSYS SpaceClaim for the FEA analysis, the curves need pre-processing criteria, including Design Overlapping (Pre-Processing), Design Sharp Conner, Design Minimum Angle and Design Minimum Distance. On the other hand, post-processing criteria include “Output Slope” and “Design Overlapping (Post-Processing)”. MATLAB software is used to verify the design suitability criteria of each design before passing the Cubic B-Spline curve into SpaceClaim, as well as preventing unsuitable designs from getting passed into the next optimisation generation. Details of each design suitability criteria will be mentioned in the below section. The value of design suitable criteria can only be NAN and 0 (satisfied / not satisfied). The logic behind the value set is to ensure that the unsuitable designs from the optimisation loop are fully filtered out during this process.

$$DSC = MAC + SCC + DO + DMD \tag{3.10}$$

3.5.3.1 Minimum Angle Control

Minimum Angle Control is a criterion created to ensure that the minimum value of the angle in each design passing through the pre-processing is not smaller than the minimum limit. An excessively small angle in the design can create an overlapping 3D geometry during 3D geometry creation and result in an error in creating a 3D design using ANSYS space claim. The angle values of the design are determined by using MATLAB software to analyse the B-spline input shape. The angle of the design from each point is measured by using basic angle calculation by using two vectors to calculate the angle between them. This process starts from the origin to the final point of the curve. The calculated angle values are sorted, and the lowest amount of angle value (θ_{min}) is used for minimum angle criteria. The minimum angle control expression can be written as below.

$$MAC = \begin{cases} 0, & \theta_{min} < 10^{o} \\ \infty, & \theta_{min} \geq 10^{o} \end{cases} \tag{3.11}$$

The output minimum angle control value of 0 and NAN can help filter non-usable design, which can prevent the optimisation process from any error possible.

3.5.3.2 Sharp Conner Control

Sharp Conner Control (SCC) is another component of design suitability criteria that is necessary to ensure that the Cubic B-Spline curves generated from the optimisation loop and passed on to the ANSYS SpaceClaim have no sharp corner in the design. Sharp corners of the design can create high stress concentration, which can create an immense amount of stress in that region during the design rotation. Having a high-stress region opposes the optimal design result, and with the SCC, the unsuitable designs are filtered out from the optimisation loop. This pre-processing criterion can reduce the computational time by skipping the FEA analysis of the unsuitable design. The SCC verification method begins by plotting a graph using input control points from the MATLAB optimisation function and then measuring the distance between each input control point. Sharp corner normally occurs by an excessively small distance between points, which is used to create a B-Spline curve. Figure 54 illustrates two b-spline curves with five control points. Two of the control points are different in both b-spline plots. The plots represent the sharp corner in the b-spline. The b-spline curve constructed using five control points illustrates the sharp corner is created at the corner section of the spline, as in the right Figure. Sharp Conner Control expression can be written as in equations 3.12 and 3.13.

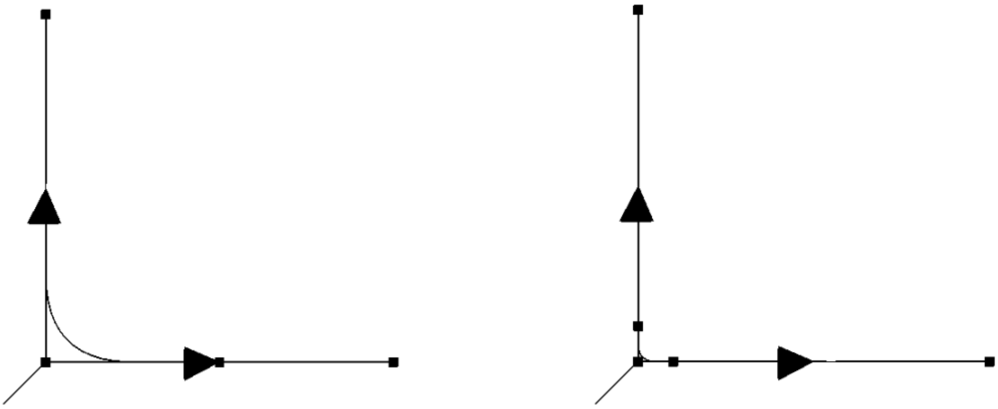


Figure 54: Illustration of Sharp Conner in B-spline plots

$$SCC = \begin{cases} \infty, & D_{min} < 0.05\ mm. \\ 0, & D_{min} \geq 0.05\ mm. \end{cases} \tag{3.12}$$

$$\begin{aligned} D_{min} &= \ \min\,(\mathrm{D(i)} - D(i+1) \quad i = 1,2,3, \dots n \\ \mathrm{n} &= \ \text{Total amount of control points} \end{aligned} \tag{3.13}$$

3.5.3.3 Design Overlapping Control (pre-processing)

The overlapping section in the generated spline from the optimisation algorithm can cause the optimisation error for the optimisation process. Design overlapping control is a criterion created to ensure to indicate the input curve from the optimisation process. The indication of the input curve is whether the spline curve contains overlapping of the design or not. In the case of overlapping design, the design is sent into the coordinate swapping process to adjust the design and validate the viability later in the process. The design overlapping control expression can be written as a function below.

$$DOC = \begin{cases} \infty, & Overlapping\ occours \\ 0, & No\ Overlapping \end{cases} \tag{3.14}$$

As in the expression, The overlapping in the B-spline curve leads to sending the design for the coordinates swapping process. The design that still has overlapping after the coordinates swapping process gets an objective function value of NAN, which is considered an unsatisfactory score. The design overlapping criterion is required to prevent the overlapping issue. It is worth mentioning that the design model generation function of the constant-torque optimisation problem does not require design overlapping control due to the fact that design overlapping cannot occur by using the constant-torque design generation function.

3.5.3.4 Design Minimum Distance Control

Design overlapping control helps reduce the possibility of optimisation errors from the issue in 3D geometry creation in SpaceClaim. However, the error still can occur due to other aspects. In this study, the 3D geometry was created by using a sweeping method, which uses a cross-sectional area and the spline curve as an input. The design overlapping can still occur when creating the 3D geometry in ANSYS due to the area required.

$$DMD = \begin{cases} \infty, & D_{min} < 5\ mm. \\ 0, & D_{min} \geq 5\ mm. \end{cases} \tag{3.15}$$

In order to prevent the error, design minimum distance control is created to ensure that the spline generated from the optimisation algorithm process has enough area for the 3D geometry creation process. The design minimum distance criteria measure the distance in the overall design as well as the distance from both ends of the b-spline. The design, which has a distance of less than 5 mm. is unsatisfactory.

3.6 Cosine-torque Spring Objective Function

The cosine-torque spring is another optimisation goal of this study. A cosine-torque spring is a spring that can provide a reaction torque as a cosine function according to its rotational angle. The cosine-torque spring in this study must be suitable for gravity compensation applications. Gravity compensator commonly requires a reaction torque or force in a cosine function relationship, which can be used to nullify the gravity force. The design methods of the cosine-torque spring are different from those of the constant-torque spring because of the difference in the b-spline generation function, objective function, and design goal. However, some of the design methods are still necessary to ensure the suitability of the design, such as the design suitability criteria.

The springs for the gravity compensator must have a torque-angular displacement function as a cosine relationship to counterbalance the gravity force. This study focuses on three main components of the cosine function plot of the gravity compensator, which are function peak value (PV), rotational angle (theta), and output reaction torque (T). These components are used to construct the objective function in this optimisation goal.

In this study, the viable range of the cosine output function of the spring does not require to begin at 0 degrees or finish at 90 degrees of rotational angle. Only the section of the function which has a cosine relationship is considered. The starting rotational angle can be offset during the assembly to ensure that the spring can provide correct cosine output by the pre-loading process.

3.6.1 Objective Function

Gravity compensator can have a range of motion from 45-180 degrees rotation. The range of motion generally varies from each application used. The goal of this study is to create a cosine-torque spring that can obtain a range of cosine-torque output at a minimum of 90 degrees of rotation. The longer the cosine function reaction torque illustrates, the better the performance of the spring. The objective function of the cosine-torque spring is created to ensure that the optimal design from the optimisation can achieve the highest cosine function range with the highest range of motion possible while minimising the stress throughout the motion. The objective function includes three main components, which are Cosine Function Range (CFR), maximum stress penalty (MSP) and design suitability criteria (DSC).

$$Objective\ function\ (CTS) = CFR + MSP + DSC \tag{3.16}$$

3.6.1.1 Cosine Function Range (CFR)

The objective function of cosine-torque spring optimisation is adapted from the constant-torque objective function. CFR is required to ensure the quality of the design and help reach optimal design easily. CFR is calculated by using 51 points of the function, including 50 points from the overall design called “overall points”. Overall points are created by dividing the result function into 50 sections. Another single point is called the “peak value point”, which is always located at the peak value point of the result function. Both types of points can be illustrated in Figure 55.

The CFR is calculated by using overall points and peak value points to find the suitability of the cosine function of the design output result from FEA. The design output result function can be called an “Output Function”. The CFR begins by finding the maximum torque value of the function or peak value point. Thenceforth, the cosine function is created by using a peak value called “Optimal Cosine Function”. Finally, the optimal cosine function created is used to calculate by using the CFR calculation expression. The CFR calculation begins by plotting Overall Points on both the optimal cosine function and output function, as in Figure 55. The difference in torque value between both functions at each point is measured and calculated by using the equation below. The lower amount of the difference (CFR value) indicates the higher qualify of the spring design.

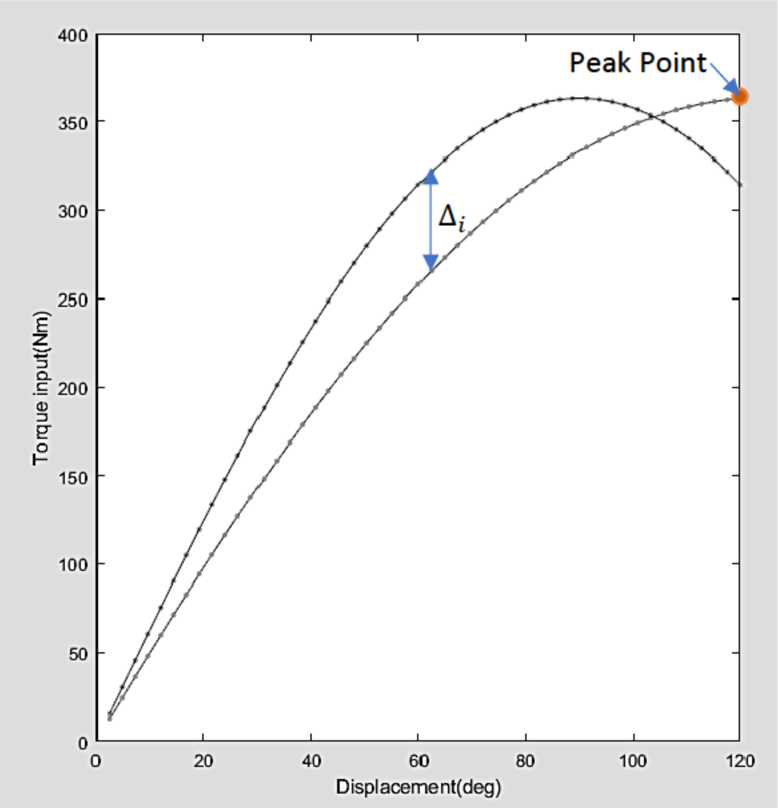


Figure 55: Cosine plot example with 50 overall points and a peak point

$$CFR = \left(\sqrt{\frac{\sum_{i=1}^{i_{max}=50}(\Delta_i)^2}{i_{max}}}\right) \tag{3.17}$$

3.6.2 Maximum Stress Penalty and Design Suitability Penalty

In cosine function spring optimisation, maximum stress penalty and design suitability penalty are still required. The maximum stress penalty is a criterion to ensure that the maximum stress of the optimal design is minimised during the optimisation process. Structural Steel is selected as a material in cosine-torque spring optimisation design as in the constant reaction torque question, which makes the maximum stress penalty unchanged. On the other hand, the design suitability penalty requires some adjustment.

Overall, the suitability penalty is adjusted by removing the slope limit control. Slope limit control is removed due to the changes in the design goal, which does not require slope limit control for this optimisation; the slope of this optimisation must follow the slope of the cosine function instead.

3.7 Scaling Optimisation

This study requires a double optimisation method to achieve the optimal result with the highest performance possible. Fmincon is the 2nd optimisation algorithm used in this study. This optimisation algorithm is a gradient-based optimisation algorithm that can reach the global minimum easily in non-complex optimisation questions. In this study, Fmincon is used for optimising the scaling factor of the model to ensure that, firstly, the maximum stress of the optimal design is lower than the yield strength of the material. Secondly, the size of the design model is at a minimum. Fmincon optimisation setting includes design parameters, design constraints and objective function as general optimisation methods.

3.7.1 Design Parameters

The design parameter of this optimisation only includes Scaling Factor (SF). The scaling factor is a factor used to determine the overall size of the spring. By adjusting the SF, the spring dimensions (width, length) are adjusted by multiplying the initial control point coordinates with the SF to achieve the new design dimensions.

3.7.2 Design Constraints

Design constraints of Fmincon are minimum and maximum values of SF to ensure that the optimisation is converged and has a limited amount of optimisation area.

3.7.3 Objective Function

The objective function of this optimisation includes two parameters, which are Size Penalty (SP) and Maximum Stress Penalty (MSP). A size Penalty is a setting for an objective function that has a goal to minimise the size of the design. The size of the optimal design from the Genetic algorithm can be further reduced while the maximum stress can be increased due to the size reduction. On the other hand, The Maximum Stress Penalty's role is to ensure that the increasing amount of the maximum stress is not beyond the yield strength. The optimal design with excessive stress can be optimised to have an increased size, which can result in a lower amount of maximum stress. Likewise, the optimal design with a low amount of output torque level can be optimised to increase the torque output by reducing the spring size.

3.8 B-Spline Generation & 3D Geometry Creation

In this section, B-spline generation and 3D geometry creation processes will be discussed. The two types of b-spline generation functions will be introduced. Moreover, the 3D geometry creation process requires design preparation to ensure that the spring design is suitable for the process. Pre-processing preparation of the spline curve is required to select the viable spring design for the geometry creation process, which will be discussed in this section.

3.8.1 B-Spline Generation Functions 1

The first b-spline generation function is a function that has the advantage of reduction in optimisation error due to the refined control points generation process. The b-spline generation process begins with generating control points on the polar coordinate system using (r, θ). This process can be called the control points generation process. While r represents the distance from the 0,0 position and theta(θ) represents the angle from 0 to 2π increments. The control points generated from the control points generation process are used for cubic B-spline construction, which generates the final shape of the spring design before passing the design for the geometry creation process using ANSYS SpaceClaim. The control points generated from the control points generation process can be written as a function below. This function is called the "Control point generation function". In this function, KP(i) represents the coordinate of each control point with a total number of 21 control points. The parameters r(i) and K_θ are input parameters that can be obtained from the optimisation algorithm and design constraints, respectively. K_θ represent a constant that determines a maximum angle for the control points.

$$KP(i) = \big(r(i), \theta(i)\big)\, ;\ i = 1,2,\ldots,21 \tag{3.18}$$

$$\theta(i) = \frac{\theta_{max}}{i} \tag{3.19}$$

$$\theta_{max} = 2\pi \times\ K_\theta;\ K_\theta = 0.7\ to\ 1.75 \tag{3.20}$$

The design constraint for this b-spline generation function includes the radius of the design parameter and the maximum angle of the b-spline. The radius of the design parameter has 21 variables, and each variable has a limit from 80 to 250 mm, while the maximum angle of the design is limited from 0.7 pi to 1.75 pi. The optimisation constraints are set based on the ANSYS analysis suitability, which can help reduce the risks of generating unusable designs.

The control points generation function is created based on the general torsion spring shape. The function not only generates control points for a unique spring design but also helps reduce post-processing time. The function is suitable for reducing the amount of time used for checking design suitability for ANSYS and can help avoid un-analysable design. Furthermore, reducing the chance of generating unusable designs can help reduce computational time since the amount of design generated in one generation is tremendous. Moreover, checking or passing unsuitable designs can lead to an exponential increase in computational time.

Another advantage of the control points generation function is that the function can help reduce the average maximum stress of the design generation by creating a design that is suitable for clockwise or anti-clockwise rotation only. Similar to a generic clock spring, which can provide large angular deformation in one direction but not another. Thus, by applying a reverse rotation with the same amount of torque, the maximum stress of the same spring can exceed the yield strength of the material. To summarise, The function can reduce the possibility of passing unusable designs into the optimisation process. Ultimately, this generation function can ensure that only designs that are suitable and have potential are considered in the optimisation.

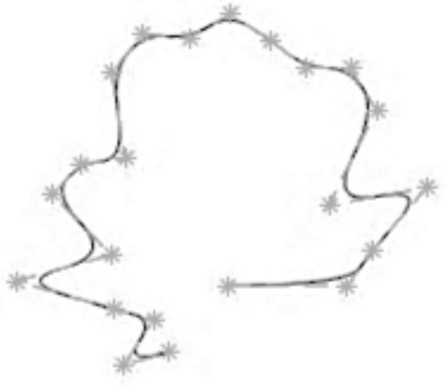


Figure 56: B-spline curve with 22 control points created by design model generation function with pi = 1.25

3.8.2 B-Spline Generation Function 2

The second type of b-spline generation function is created due to the limitation in the optimisation possibility of the first function, which cannot satisfy the cosine-torque spring optimisation goal. This b-spline generation function has unlimited possibilities in terms of shape generated due to the developed control points generation function. However, the unlimited possibility of shape generation also comes with drawbacks, such as an increasing amount of overlapping shapes and unusable b-spline shapes for geometry creation.

Similar to the previous b-spline function, this function begins with control point generation. However, this function begins by generating seven control points in (x, y) coordinates. The control points are generated randomly while having a constraint in the XY axis.

$$KP(i) = \big(x(i), y(i)\big), \qquad \begin{array}{l} i = 1,2,\dots,7 \\ x(i) \leq 250 \\ y(i) \ \leq 250 \end{array} \tag{3.21}$$

This b-spline generation function introduces a simple process to create a b-spline plot. However, due to its simplicity and robustness, the generation process still requires other components to limit the possible errors. Design overlapping control and coordinates swapping process are specifically created to resolve the drawbacks of this function, which will be discussed in the next section.

3.8.3 Pre-Processing Preparation

Design preparation is necessary to minimise the error in software connections and processing, maximise the optimisation performance, and reduce overall optimisation time. The design preparation includes four main aspects, which are Spline Preparation, Coordinates Swapping, and Design Coordinates Rotation. All of the aspects mentioned in this section are the processes required for pre-processing input and output of each software, which is necessary for combining both MATLAB and ANSYS for design optimisation. In addition, the Coordinates swapping and Design Coordinates Rotation processes are created specifically for cosine-torsion spring optimisation, which uses the 2nd B-spline generation function to limit the possible error that occurs due to the robustness of the generation function.

3.8.3.1 Spline Preparation

In general, CAD software generates spline curves using their specific spline creation function. Ansys software uses its own unique spline function type to create a spline that cannot be identified. Using different types of input spline curves can cause an issue in geometry. However, this process is still required since the spline function of ANSYS software is unknown. In order to create a methodology that can be easily replicated and repurposed, overcoming this challenge is required.

A cubic B-spline curve was chosen as an input to create a 3D model. The spline includes 22 control points. The first issue was discovered by sending the input directly, using the control points of the spline as input for the geometry creation process, which results in inaccurate spline output dimension due to the difference in spline type. The issue can be solved by dividing the b-spline curve into small equal sections. Each section is used to create a tiny spline curve. In the end, the tiny spline curves are connected to create an initial b-spline shape that is generated by input control points. This method can resolve the issue and provide a spline output with no error. However, without the right amount of resolution, the result is still possible to have a design dimension error. Furthermore, increasing the resolution of the input spline also increases the computational time. In this study, the b-spline curve divides into 2000 sections (100 sections within each b-spline knot). Each section is converted into control points and passed into ANSYS software. The spline curve created from converted control points has enough resolution to ensure that the spline created from both software are identical.

3.8.3.2 Design Coordinates Preparation

Design coordinates preparation is required to ensure that coordinates of the generated design from MATLAB software are suitable for ANSYS SpaceClaim for creating the 3D geometry. Non-suitable coordinates can cause issues to the overall optimisation result and optimisation process, such as corrupted geometry and optimisation process error. Optimisation process errors can cause the optimisation workflow to be terminated. This requires the user to manually fix the issue and continue the optimisation process, which can increase the computational time. The design coordinates preparations include coordinates swapping and coordinates rotation.

3.8.3.3 Coordinates Swapping

Coordinates swapping is a method used to correct the design generated by an optimisation algorithm that has a design overlapping. The overlapping section of the spline is shown in Figure 57. The design generated by the optimisation algorithm is required to be verified for design overlapping criteria prior to any further process. The coordinates swapping process will be executed for the design that cannot fulfil the criteria. The process begins with swapping each input control point in order until the design overlapping is eliminated. However, eliminating overlapping in the design is not always possible. The impossible designs are rated as unsatisfied designs in design overlapping criteria. On the other hand, the design with resolved overlapping gets passed into the optimisation process with a satisfied rating.

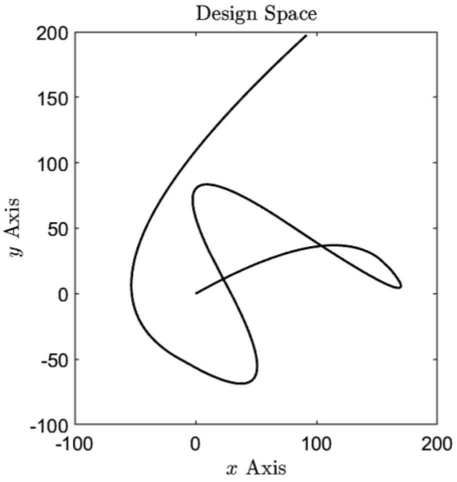


Figure 57: B-spline curve with knots

3.8.3.4 Design Coordinates Rotation

Design Coordinates Rotation is a method used to ensure that the design generated from the optimisation algorithm is suitable for ANSYS SpaceClaim to produce a 3D model of the design. The design generated by the optimisation algorithm, which is suitable for ANSYS SpaceClaim, must have a parallel section from the origin to 2nd control point to ensure that the 3D geometry of the spline is created correctly. In detail, the design coordinates rotation code created a vector of the control points 0 and 1 and calculated the angle between the vector and the x-axis. Finally, the generated spline is rotated by the calculated angle until the vector and x-axis are aligned. Design Coordinates rotation is necessary to ensure the high quality of the design and output. Without this process, the fault geometry with fault result can get passed into the population, which results in the unreliability of the optimal result. For example, Figure 58 illustrates the spring design with design coordinates rotation (vector AB is parallel with the x-axis).

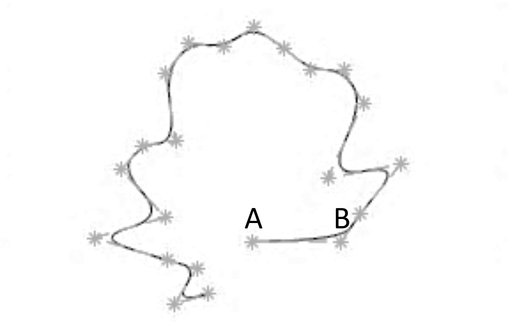
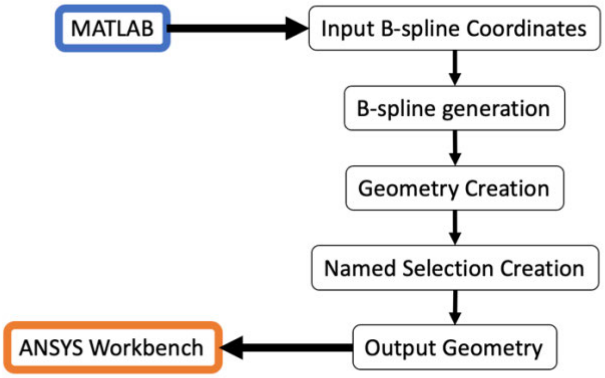


Figure 58: B-spline plot with coordinate rotation

The cross-sectional area of 3D geometry with coordinates rotation has a correct size and angle. On the other hand, the cross-sectional area of the 3D geometry without design coordinates rotation has an incorrect angle, which leads to incorrect width and height of the overall 3D design. This issue can cause an output error in FEA analysis.

3.8.4 3D Geometry Generation

3D geometries used for the optimisation process in this study are generated by using SpaceClaim software. In this study, The spline curve is automatically generated using Python scripting in SpaceClaim software for ease of design optimisation. The scripting process includes three main steps, which are B-spline generation, Geometry creation, and Name selection creation.



The b-spline curve created by MATLAB software with high-resolution coordinates is transferred to ANSYS SpaceClaim for plotting a spline curve. The curve will be used in geometry creation. Then, the 3D geometry of the B-spline curve is created by using a sweeping method based on the input cross-sectional area setting. Afterwards, the name selection for each section of the spline is created for the ease of executing the FEA process automatically.

3.9 Finite Element Analysis

Design verification is essential for the optimisation process. In order to measure the suitability and performance of each spring design, FEA is used as a main tool for design verification due to the repeatability of the task. ANSYS Workbench is selected as software for this process due to various reasons. Firstly, Workbench software can execute the FEA analysis with high-quality output and with multiple settings. Secondly, the software has a Python scripting feature, which can be used for repetitive tasks. Lastly, the ANSYS software license is available and used as the main software for the students at the University of Wollongong. In summary, the software was selected due to its ease of use, familiarity, and the essential feature of the software. In this study, the results expected from FEA are maximum stress and reaction torque function. Both outputs are required for the objective function, which is used as a design performance measurement. This section will introduce three aspects of the FEA, which are the FEA setting, ANSYS scripting, and software limitation.

3.9.1 FEA Settings

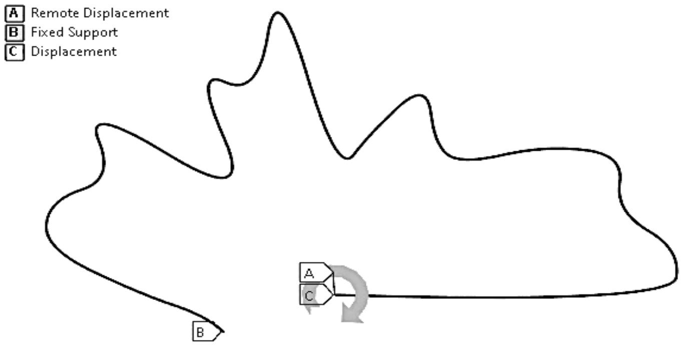
A suitable FEA setting must be used in order to achieve reliable output performance and expected outcomes.

3.9.1.1 Mesh Setting

In terms of mesh setting, a suitable size of meshing is required in order to achieve a converged analysis result. Poor meshing can result in unconverged results as well as in accuracy in spring output. In this study, the mesh size setting was selected to be 0.5 mm. The selected mesh size was measured by design trial to find a suitable mesh size for this study. Another aspect of mesh setting is meshing types. The mesh type selection of this study is nonlinear mechanical analysis with linear element order. This can help reduce the computational time due to the element type selection will prioritise using hexahedral meshes, which requires lower computational time than others.

3.9.1.2 Design Constraints

Design constraints are represented by spring application in practical. The design constraints in this study include a fixed support, an angular displacement, and a pin support. Fix support represents a position of fix support in practical during spring installation. Furthermore, angular displacement and pin support represent the position of input rotation for the applications.



3.9.1.3 Other settings

For ANSYS software, another setting is required for the optimisation. Large deformation analysis is required in this study due to a large amount of movement during the spring rotation. Nonlinear analysis is also required due to the nonlinearity of output types.

3.9.2 ANSYS Scripting

ANSYS ACT is a feature for using Python language to control the setting, workflow, and overall ANSYS software. In this study, the scripting method was used in 4 different aspects, which are exporting output, optimisation setting, FEA process execution and software communication. Figure 59 illustrates a flowchart of the Ansys ACT scripting with inputs and output interaction between MATLAB and ANSYS software.

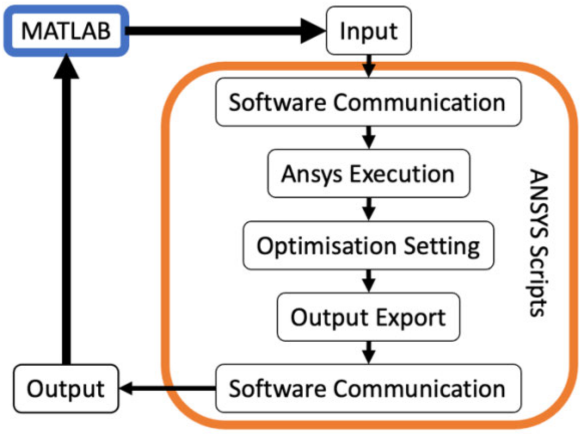


Figure 59: Ansys Scripting Flowchart

3.9.2.1 Software communication Script

Software communication scripting is created for software communication between MATLAB and ANSYS in terms of input and output communication as well as error resolution. The data required for communication includes finite element analysis results output from ANSYS software and B-spline geometry coordinates from MATLAB software as an input.

3.9.2.2 Python Script for Output Export

Output export script is responsible for exporting a finite element analysis result of the spring for future use. In this study, the built-in function of result export was not available, which makes crafting Python script from scratch necessary. The output export includes maximum stress, torsion-output function, and inner-outer node locations, which are used for design overlapping checks mentioned in the methodology section.

3.9.2.3 Python Script for Optimisation Settings

A suitable optimisation setting is required in order to obtain the best performance results while minimising inaccuracy and error during the optimisation process. The Python script was created to ensure that the optimisation setting for every iteration must be identical, which can help ensure the output performance. The optimisation setting includes mesh setting, material setting, design constraints, b-spline geometry's surface dimensions, as well as other ANSYS basics settings such as name selection and analysis setting.

3.9.2.4 Python Script for ANSYS Execution

Python execution code works as an activator for ANSYS software to proceed with the finite element analysis in each iteration. The execution code receives a command from software communication to start the FEA process.

3.9.3 Software Limitation

In Genetic algorithm optimisation, the amount of population is a crucial aspect for achieving an optimal result. Excessive computational time can happen due to the inferior number of populations. In this study, a minimum of four criteria were created to prevent an error caused by the software limitation. The designs generated from the optimisation algorithm must fulfil the criteria in order to proceed into the next optimisation process. The design that cannot fulfil the criteria is rejected without any consideration in the FEA result. For example, generally, for this study, 30% of the first generation of the optimisation gets rejected to prevent the error from software limitation, which can lower the amount of spring design passing on to the next generation. This can result in difficulty in reaching optimal design at the global minimum. The solution for this issue is to add extra amount of population to ensure that the optimisation algorithm receives enough viable design to proceed into the next generation. However, the extra amount of population results in extra computational time, which makes this study may not have the best performance in computational time.

Chapter 4: Design Examples

In this thesis chapter, four nonlinear spring examples are presented to illustrate the outcome and performance of the design methodology introduced in the previous chapter. The nonlinear springs include constant-torque and cosine-torque springs. Two different b-spline generation functions were used for each spring type to investigate the performance of each b-spline generation function.

4.1 Constant-Torque Spring

Constant-torque spring is one of the design goals in this study. The spring must be able to provide a constant-torque function. This section illustrates two constant-torque spring designs from 2 different b-spline generation functions as well as the specification of the spring.

4.1.1 Constant-Torque Spring (1st B-spline Generation Function)

The optimisation setting of the spring includes 100 populations, while each chromosome includes 22 adjustable design parameters, as mentioned in the methodology chapter. The maximum generation of this optimisation was set to 100 generations. Design specification and output results can be illustrated in Table 6.

Table 6: Constant-Torque Spring Result (1st B-spline Generation Function)

Spring Specifications	Design Specification	Output Results
Objective Function Value	N/A	0.5802
Optimisation Time (Hours)	N/A	28
Constant-torque Value (Nmm)	10	7.68
Constant-torque Range (Degrees)	50	40
Design Space (mm)	600 x 600	250 x 450
Out-of-Plane Thickness (mm)	3	0.8
In-Plane Thickness (mm)	1	0.8
Steady-State Error (%)	5	0.7
Maximum Stress (MPa)	150	133

As in Table 6, the spring design has an objective function value of 0.5802, and the spring was able to provide 7.68 Nmm maximum constant-torque output with 40 degrees of constant-torque range, which is slightly lower than the design specification. However, the steady-state error of the constant-torque spring was calculated to be 0.7%, which is drastically lower than the design specification. Furthermore, the constant-torque range of this spring design can be increased by using a 5% steady-state error. Figure 60 illustrates the constant-torque spring design as well as its output function from computational analysis with a constant-torque level. The maximum stress of the spring was 133 MPa, which was below the yield stress of the material. The amount of maximum stress can represent a safety factor of the spring, which results in a 1.87 safety factor value. In terms of spring size, the footprint and spring thickness are lower than spring specifications.

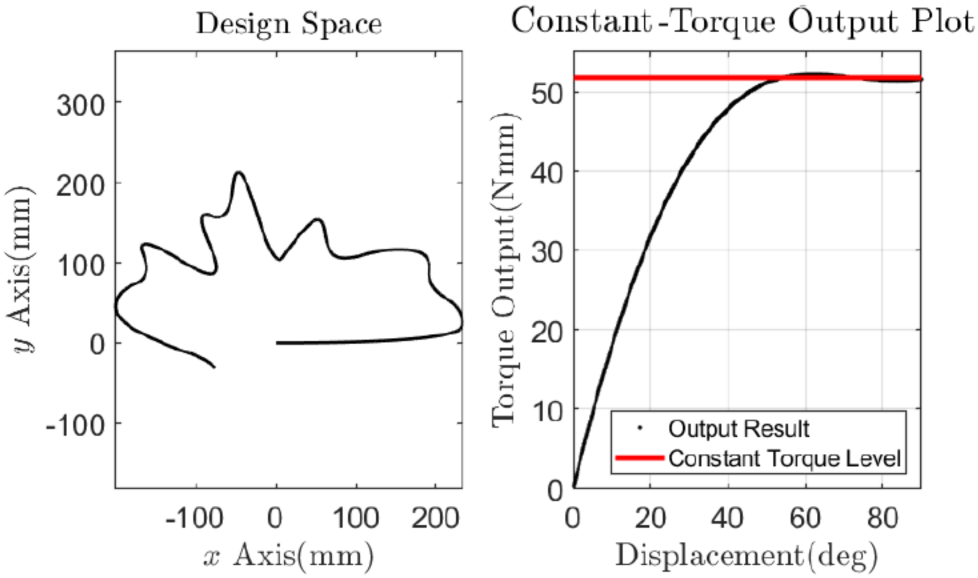


Figure 60: Optimal Design of Constant-Torque Spring (left) and Design Output (right) (1st B-spline Generation Function).

4.1.2 Constant-Torque Spring (2nd B-spline Generation Function)

In this spring type, the optimisation setting is changed due to the disadvantage of the 2nd B-spline generation function, which is mentioned in the previous chapter. The population was set to 500, while the maximum generation of the optimisation was set to 100 generations. The design specification and output result of the spring is illustrated in Table 7.

Table 7: Constant-Torque Spring Result (2nd B-spline Generation Function)

Spring Specifications	Design Specification	Output Results
Objective Function Value	N/A	0.271
Optimisation Time	N/A	40
Constant-torque Value (Nmm)	10	9.916
Constant-torque Range (Degrees)	50	35
Design Space (mm)	600 x 600	400 x 300
Out-of-Plane Thickness (mm)	3	0.8
In-Plane Thickness (mm)	1	0.8
Steady-State Error (%)	5	5
Maximum Stress (MPa)	150	193

The spring has an objective function value of 0.271. Moreover, the results illustrate a constant-torque value that is much lower than the design specification by using the same value of spring thicknesses. It is worth mentioning that the constant toque value can be increased by increasing the out-of-plane thickness of the spring. However, increasing design thickness can result in an increase in weight and size, which can be a disadvantage of this spring type. The constant-torque range of the spring is also lower than the design specification, while the steady-state error of this spring is still under the design specification. The maximum stress of the spring is higher than the design specification. However, the maximum stress of the output results remained lower than the yield strength of the material. The value safety factor of the spring is calculated to be 1.295. Figure 61 illustrates the geometry of the spring as well as the spring design output function.

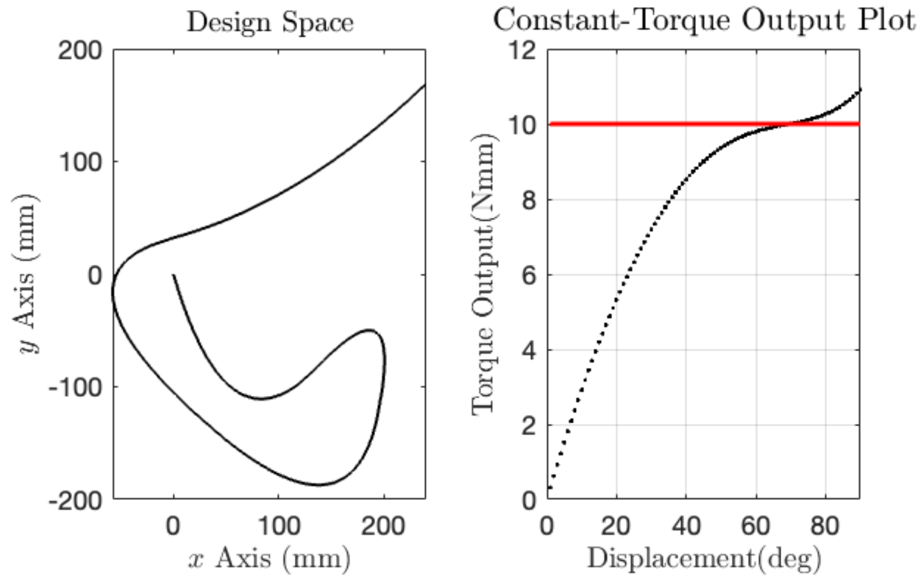


Figure 61: Optimal Design of Constant-Torque Spring (left) and Design Output (right) (2nd B-spline Generation Function).

4.2 Cosine-Torque Spring

Cosine spring was designed to be used in gravity compensation applications. The spring must be able to provide a cosine function torque prescribed to the rotation angle. Due to the design criteria, the design specifications are set in Table 8. It is worth mentioning that steady-state error in this design goal must be minimised as much as possible to provide a reliable cosine-torsion spring due to the fact that the error in the cosine-torque function has a major effect in gravity compensation application.

4.2.1 Cosine-Torque Spring (1st B-spline Generation Function)

The optimisation setting in this design is similar to the constant-torque spring with 1st b-spine generation function in terms of population, generation and design optimisation parameters.

Table 8: Cosine-Torque Spring Result (1st B-spline Generation Function)

Spring Specifications	Design Specification	Output Results
Objective Function Value	N/A	2.544
Optimisation Time	N/A	16
Maximum Torque Value (Nmm)	10	9.66
Cosine-torque Range (Degrees)	90	90
Design Space (mm)	600 x 600	400 x 400
Out-of-Plane Thickness (mm)	3	0.8
In-Plane Thickness (mm)	1	0.8
Maximum Stress (MPa)	150	146

The optimal design has an objective function value of 0.544. The spring can provide a maximum torque of 9.66 Nmm with a cosine-torque range of 90 degrees of rotation. The maximum stress of this spring was 146 MPa, which is slightly lower than the specifications of the design. The maximum stress is lower than the yield strength of the material, which is acceptable. However, the torque-displacement function of the spring provided a high amount of error compared to the cosine function, as in Figure 62.

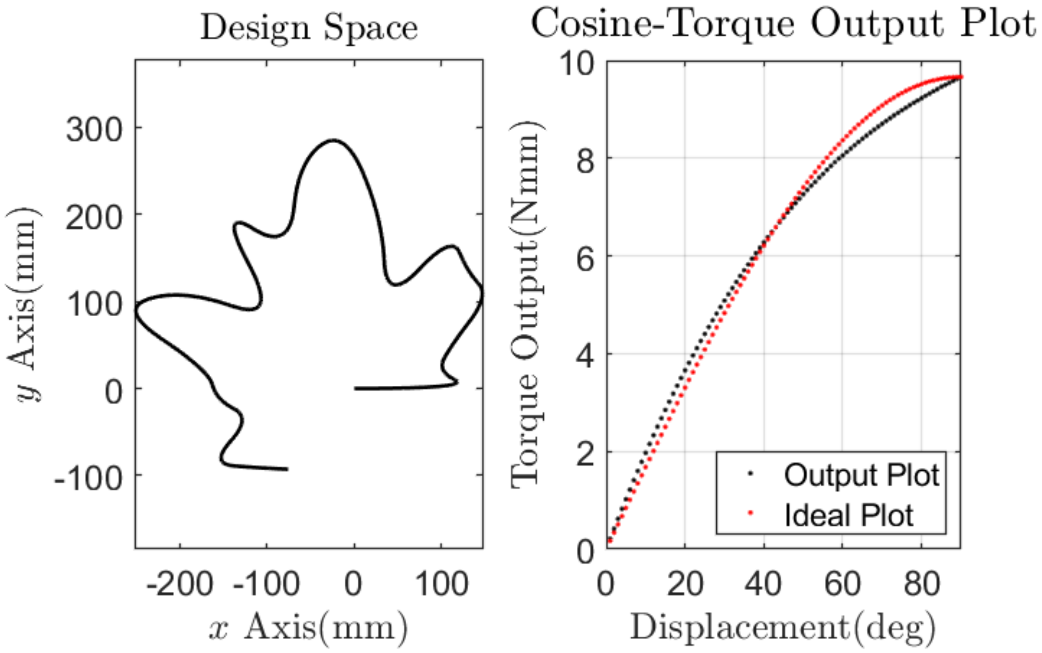


Figure 62: Optimal Design of Cosine-Torque Spring (left) and Design Output (right) (1st B-spline Generation Function).

4.2.2 Cosine-Torque Spring (2^{nd} B-spline Generation Function)

In this cosine-torque spring with the 2^{nd} b-spline generation function, the optimisation setting is identical to the constant-torque spring with the 2^{nd} b-spline generation function in the previous section. Table 9 illustrates the design specification as well as the computational analysis output of the spring.

Table 9: Cosine-Torque Spring Result (2^{nd} B-spline Generation Function)

Spring Specifications	Design Specification	Output Results
Objective Function Value	N/A	1.9
Optimisation Time	N/A	38
Maximum Torque Value (Nmm)	10	4.667
Cosine-torque Range (Degrees)	90	90
Design Space (mm)	600 x 600	300 x 350
Out-of-Plane Thickness (mm)	3	1
In-Plane Thickness (mm)	1	0.8
Maximum Stress (MPa)	150	167.56

The spring has an objective function value of 1.9. The maximum torque value of the spring output was 4.667 Nmm. which is lower than the maximum torque value in the design specification. The spring is able to satisfy the design specification criteria in other aspects, including cosine-torque range, spring footprints, and design thicknesses. However, this spring design illustrates an outstanding performance in terms of minimising function error value. However, the maximum stress of the spring slightly exceeds the design specification with a safety factor value of 1.492. Figure 63 illustrates the design geometry of the cosine-torque spring as well as its output.

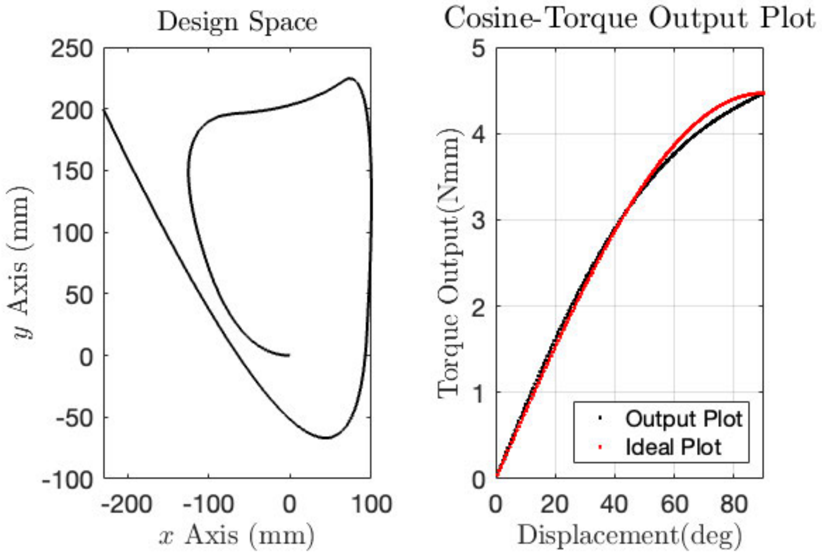


Figure 63:Optimal Design of Cosine-Torque Spring (left) and Design Output (right) (2^{nd} B-spline Generation Function).

The optimal results from the optimisation process are acceptable. Overall, the optimisation results can fulfil most of the design criteria, such as spring size, constant-torque range, and cosine-torque range. On the other hand, some of the criteria cannot be satisfied, such as maximum stress and maximum torque value of the spring. In addition, the specifications can be adjusted by scaling optimisation as well as adjustment of spring’s thickness. The crucial aspect of each spring design is function error, which is represented by the unique shape of the spring. The results illustrate that some of the results illustrate an outstanding performance than others in terms of minimising function error.

Optimisation time is another aspect that can determine the performance of the design methodology. The optimisation time of the design geometric optimisation is 16 to 40 hours. The first b-spline function requires a lower amount of optimisation time, while the second b-spline generation function requires almost twofold of the computational time.

In this chapter, scaling optimisation is not necessarily due to various reasons. The optimal design of the springs generated from the optimisation process can fulfil the design criteria in terms of spring dimensions as well as the maximum stress. In the case of exceeding maximum stress or spring dimensions, scaling optimisation is required to optimise the spring size, which results in adjusting the overall dimension as well as the maximum stress of the spring.

Chapter 5: Practical Experiment

Practical experiment is an essential process for research and design development. In general, the experiment is required in order to validate the feasibility of the introduced methodology and design in reality. In this study, the experiments were used to verify the reliability of the computational results of each spring, including cosine-torque and constant-torque springs. The goal of this experiment is to compare the output function of the springs of computational analysis and practical experiment by using a torque measurement sensor and manual torque test stand. This chapter will illustrate the essential aspects of the experiment conducted in this study, including torsion spring production, experiment setting, and experiment results.

5.1 Experiment Tools

In order to verify the feasibility of the methodology introduced in this study, as well as the performance of the spring result, reliable experimental tools must be used to ensure the quality of the result. In this experiment, torque sensor MR50-100 from MARK-10 company is selected as an experimental tool. The specification of the sensor is illustrated in Table 10.

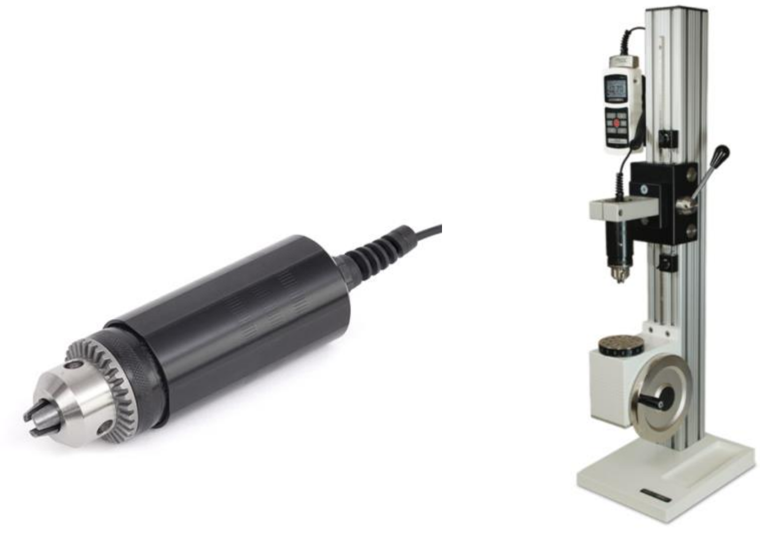


Figure 64: Torque sensor (left) and Manual Torque Test Stand (right)

Table 10: MR50-100 Specification

MR50-100 Specification	
Accuracy	± 0.35% of full scale + indicator
Capacity	1150 Ncm
Resolution	1 Ncm

The selected sensor alone is impractical for the experiment of this study due to the lack of angular rotation indication. The manual torque test stand (TST) from MARK-10 company is also being used for this study to ensure the accuracy of the rotational angle, which increases the output correctness. The experiment setting can be shown in Figure 64. In this study, the experimental tools are selected due to the availability of the equipment at the university and the suitability of the device for this experiment. However, prioritising the availability of the tools leads to minor design changes of the spring as well as additional equipment. Furthermore, due to the adaptivity of the spring design, the design changes can be executed with ease. Design changes and their cause will be discussed in the next section.

5.2 Spring Production

In this experiment, two of the springs produced by the 1st B-spline generation function were selected for the experiment. The springs include constant-torque and cosine-torque springs. The production material of the springs was selected in the initial computational analysis which is structural steel, due to the suitability of the material properties as well as material availability of the university production workshop. However, the springs cannot be directly connected to the experimental devices due to the incompatibility. The spring designs from computational analysis require changes as well as an additional tool for resolving the compatibility issue.

5.3 Design Compatibility Adjustment

Incompatibility between the spring and experimental tool must be solved. 2 major issues were identified, including 1st Torque sensor resolution and 2nd Output measurement position.

5.3.1 Torque Sensor Resolution

Torque sensor resolution is one of the compatibility issues of the original spring design. The resolution of the torque sensor available for the experiment is 1 Ncm, as in the previous section. However, the maximum output torque of the computational analysis was lower than 1 Ncm, and the most suitable sensor resolution is 1 Nmm. Due to the priority in adjusting the design to increase compatibility, the solution for this obstacle is to increase the out-of-plane thickness of the spring, which can increase the amount of maximum output torque.

For the experiment, the spring designs' thickness is adjusted to 4 millimetres to increase the output torque of the springs. The thickness was selected due to the material availability of the university. Figure 65 illustrates adjusted thickness springs produced by a water jet cutter. Furthermore, computational results of the springs for design verification require to be updated according to the adjusted thickness of the spring. In detail, Finite element analysis for the springs is required in order to verify the expected results of the spring reaction torque output as well as the maximum stress of the spring. Table 11 illustrates the FEA output of the springs with adjusted thickness in terms of maximum stress and maximum output torque, as well as a comparison of the design properties in different thicknesses. Interestingly, The output torque of the adjusted thickness is not proportional to the increased thickness ratio as expected. However, the increasing maximum torque value between the two spring types has a different scaling ratio even though the thickness ratio is equal. Furthermore, the maximum stress of both spring after the thickness adjustment are slightly increased.

Table 11: FEA Output Comparisons Between Different Spring Thickness

Spring Types	Out-of-plane Thickness (mm)	Maximum Stress (MPa)	Maximum Output Torque (Nmm)
Constant-Torque	0.8	133	7.68
	4	146	28.916
Cosine-Torque	0.8	146	9.6636
	4	158.25	21.886

5.3.2 Output Measurement Position

During the computational analysis, the position of output measurement was selected to be at the fixed support of the spring. However, the experiment setup requires an output sensor to be at the origin position of the measurement device, which is the first obstacle to proceed further with the experiment. Literally, the position of the input torque and fixed support of the spring are interchangeable. Furthermore, the resultant torque of the spring remains unchanged as long as the rotational trajectory is unchanged. Adopting specified information, specifically designed connectors that connect the spring, and the experimental tool were created.

The connectors must be able to maintain spring motion as in the computational analysis setting to ensure the output performance. The connectors are created using a steel sheet with 3 millimetres thickness to prevent a possible error of deformation in connector geometry during the experiments. The designs of connectors are created from a foundation of spring and experimental tool geometries. The connector design was also verified by using finite element analysis to ensure that the connector satisfy all criteria. The connector geometries produced by the water jet cutter are illustrated in Figure 65.

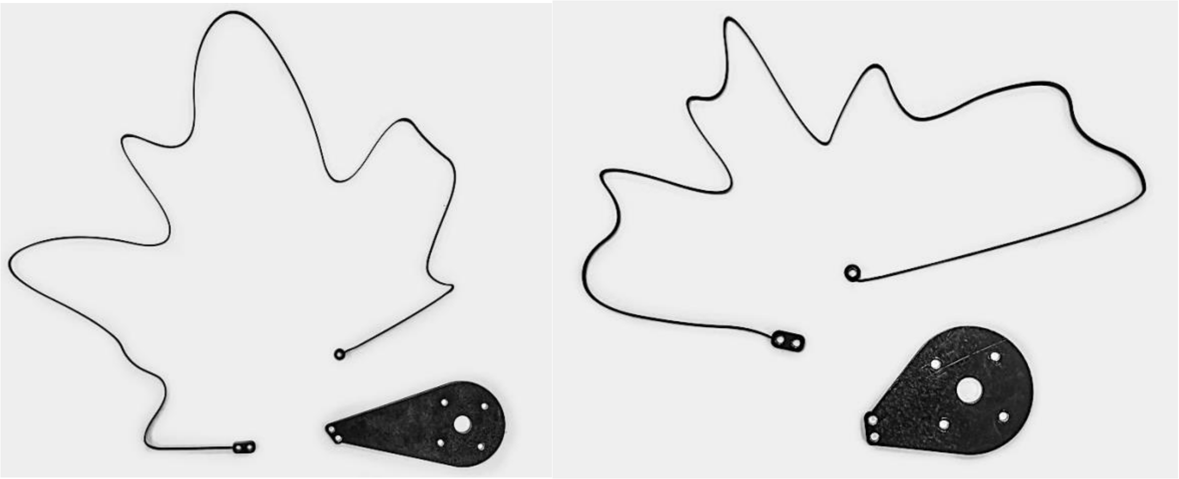
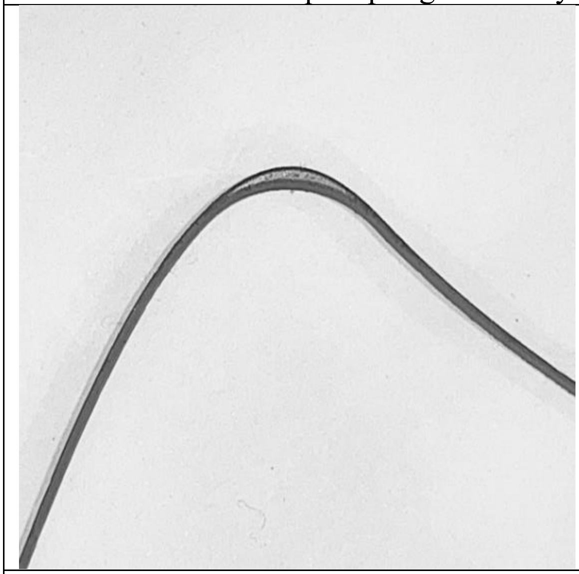
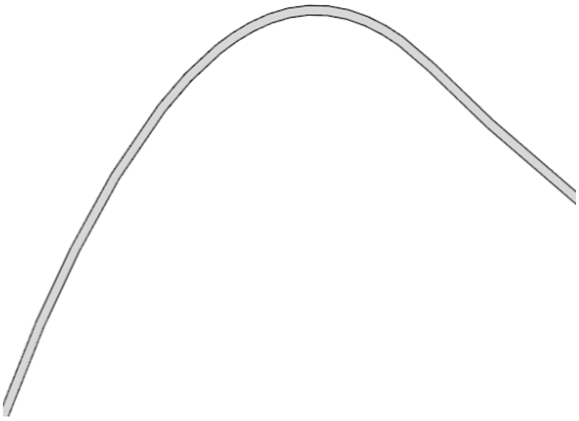
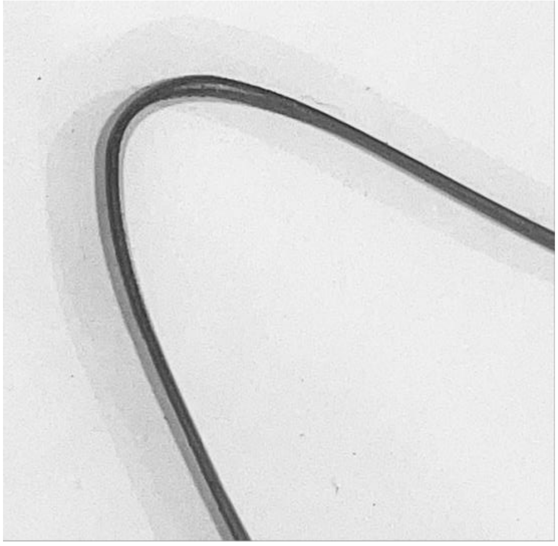
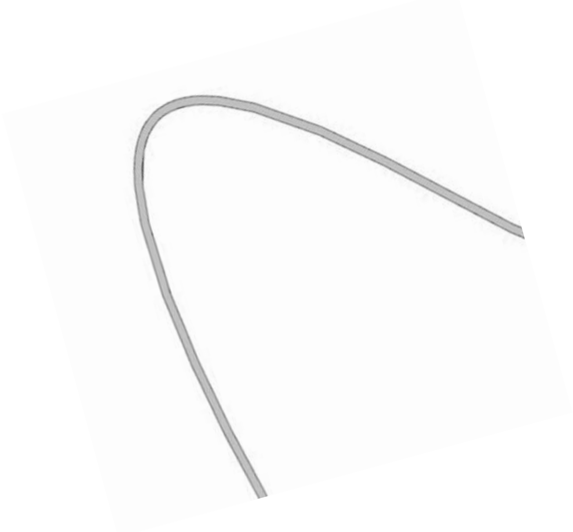
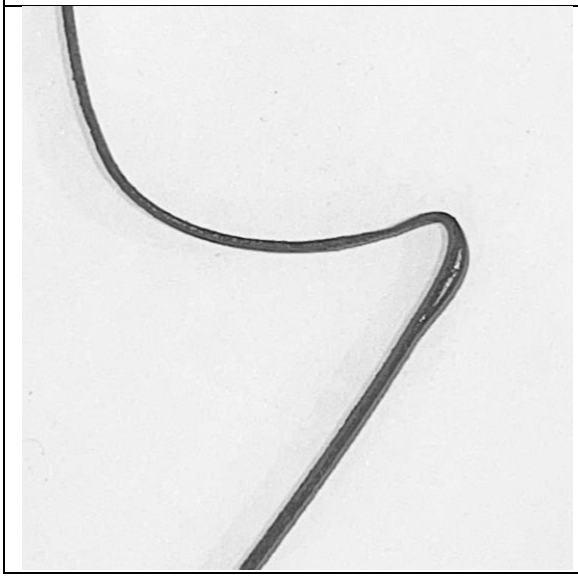
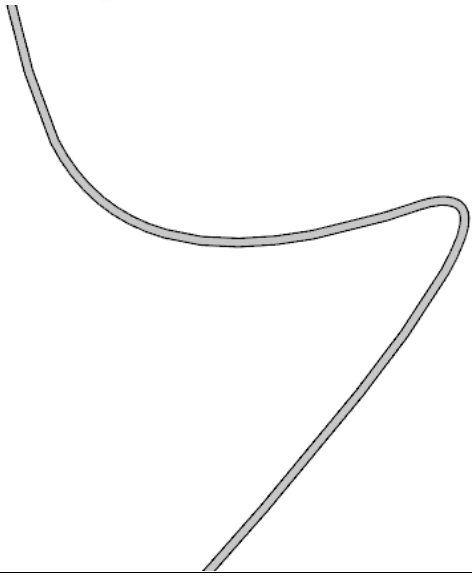


Figure 65: Produced Cosine-torque Spring and Connector (left) and Constant-Torque Spring and Connector (right)

5.3.3 Geometric Error from Production Process

Geometric error can lead to output performance issues in terms of output accuracy and precision. The cosine-torque spring produced by the water jet cutter has a geometric error due to an issue during the production process. The weight and size of the spring geometry create a displacement during the process, which results in some geometric errors. The errors are shown in Table 12. The Table illustrates a comparison between the correct geometry from the original step file and a produced cosine-torque spring. On the other hand, the constant-torque spring has an outstanding geometric accuracy due to the size of the spring, which is almost 50% of the cosine-torque spring size. The spring is expected to provide an accurate result during the experiment. However, the cosine-torque experiment result may contain some errors due to the geometric error.

Table 12: Geometric Error Comparisons

Produced Cosine-Torque Spring Geometry	Expected Cosine-Torque Spring Geometry
	
	
	

5.4 Experiment Setting

The experiments include three main components, which are torque sensor, torsion springs, and connectors. The experiment setup is shown in Figure 66. The experiment's goal is to investigate three components, which are 1st spring output function and 2nd spring deformation. Firstly, the spring output function will be measured by measuring the reaction torque of the spring according to the rotational angle. The spring output function of the springs will be used to compare with computational results to validate the output performance of the springs in practical and computational analysis. On the other hand, spring deformation will be measured after conducting the toque testing to identify the spring geometry for the deformation, which can represent an exceeding amount of the maximum stress of the spring. In this experiment, the measurement of the output torque will be measured multiple times to ensure the accuracy of the output result.

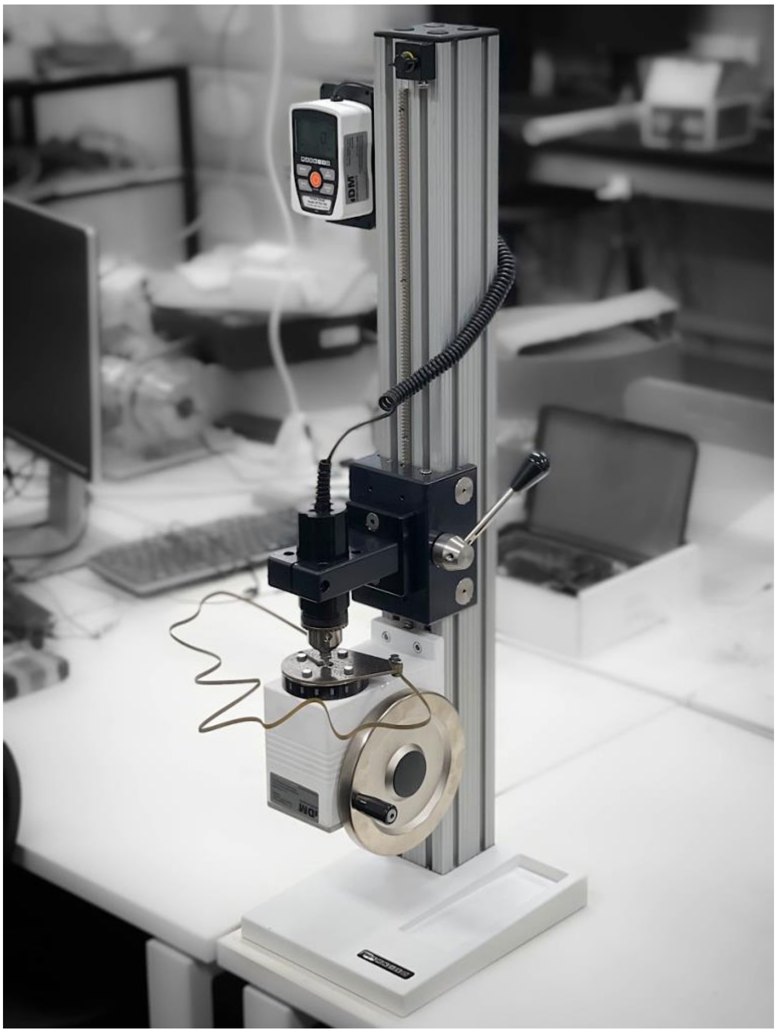


Figure 66: Torsional Test Experiment Setup

5.5 Experimental Results

In this section, the experimental results of the springs, including constant-torque and cosine-torque springs, will be illustrated. The result in this section includes maximum torque, reaction torque in each rotational angle and spring deformation.

5.5.1 Constant-Torque Results

Constant-torque spring design is selected as one of the design goals. The spring is required to provide a constant-torque output throughout the rotational movement. Table 13. Illustrate the constant-torque output at each specific rotational angle. The measurement is taken when the output torque value changes at a specific rotational angle. However, the reaction torque remained constant from 46 to 90 degrees of rotation.

Table 13: Experimental Results of Constant-Torque Spring

Rotational Angle (Degrees)	Output torque (Ncm)
0	0
5	1
20	2
46	3
70	3
90	3

The output Table can be transferred into a reaction torque function plot, as in Figure 67. The results illustrate a constant-torque range from 46 degrees of rotational to 90 degrees, as expected.

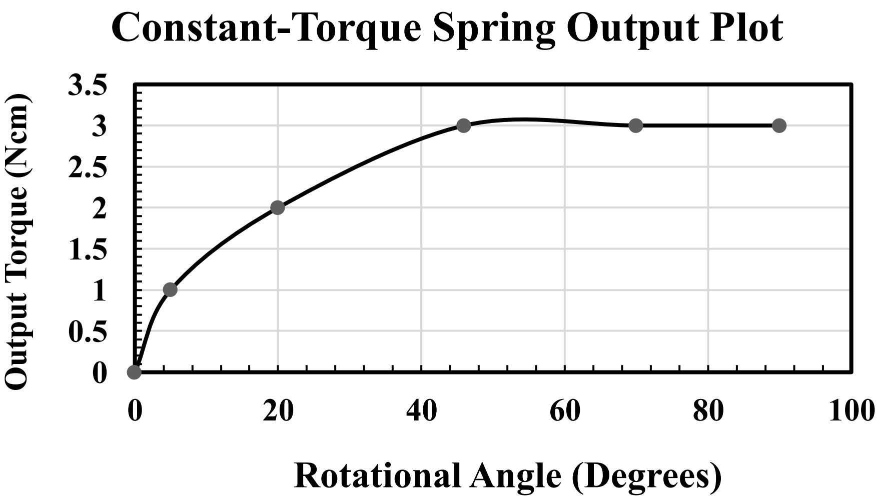


Figure 67: Constant-Torque Spring Experimental Output

5.5.2 Cosine-Torque Results

Cosine-torque is another design goal in this study. The expectation of the result is to be able to provide a cosine relationship according to the rotational angle. Table 14 illustrates the result output torque in each rotational degree as well as the cosine value at each rotational degree. The spring functional range is from 0 to 90 degrees. However, in this experiment, the measurement is taken as the output torque value changes. Moreover, the resolution of the torque sensor cannot measure any changes between 67 to 90 rotational angles, which makes this experiment require finding the following measurement point at 103 degrees.

Table 14: Experimental Results of Cosine-Torque Spring and Optimal Cosine Value

Rotational Angle (Degrees)	Output Torque (Ncm)	Cosine Value
0	0	0.000
5	1	0.523
18	2	1.854
30	3	3.000
46	4	4.316
67	5	5.523
103	6	5.846

The output can be transferred into an output function plot, as shown in Figure 68. The plot shows the cosine function as well as the output torque function.

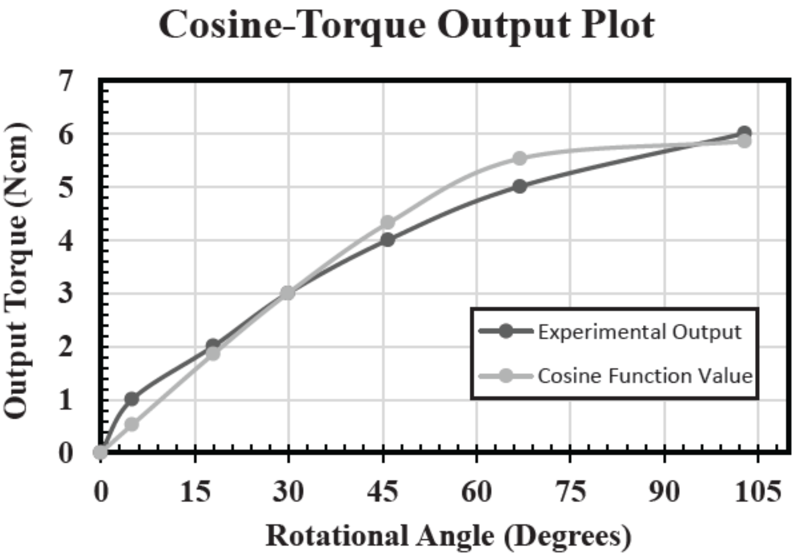


Figure 68: Cosine-Torque Spring Experimental Output

The performance of overall results is as expected. A constant-torque spring was able to provide a constant-torque range up to 44 degrees with a maximum torque of 3 Ncm. On the other hand, the output of the cosine-torque spring has a 7.6% overall error. Furthermore, the output plot is required for the verification of the performance of the spring by the comparisons between computational and practical results. In addition, the comparisons can provide reliability and predictability of the design, which will be discussed in the next chapter.

The experiment results for both springs are acceptable to verify a spring performance in the next chapter. However, the results' quality can be improved further in some aspects. As in the experiment output function, a noticeable error of output torque from 0 to 1 Ncm occurred in both cosine-torque and constant-torque springs, which generally happens while using a sensor with low resolution compared to the output level. This issue also creates a difficulty in measuring the exact value of the function in each rotational degree, which may cause some errors in the experiment result. However, the results will be used for a comparison in the next chapter, which can be verified by using the experimental data and computational analysis data.

Chapter 6: Discussion

This chapter will discuss the performance of each spring design in terms of maximum torque, steady-state error, range of motion, maximum stress level, computational time, and springs' footprints. Furthermore, this chapter will discuss the experimental results performance compared to computational analysis results to verify the reliability of the methodology introduced in this study. Moreover, the aim of this chapter is to verify the advantages, disadvantages and possible improvements of the springs and design methodology. Lastly, the performance of springs created in this study will be compared to the existing spring designs in other studies in terms of performance comparisons in similar aspects as mentioned above but also includes manufacturing difficulty, installation difficulty, and application flexibility. Multiple studies will be used in this comparison by selecting the best performance spring design in each aspect from the literature review chapter.

6.1 Constant-Torque Spring

Constant-torque spring designs in this study include two design geometries from two different b-spline generation functions. The performance of both spring designs will be discussed in this section. In 3 different aspects, which are design performance comparisons, Result validation and design comparisons with other studies.

6.1.1 Design Performance

In terms of design performance, the springs will be validated and compared in three different aspects, which are 1st output performance, 2nd physical performance and 3rd computational time.

6.1.1.1 Output Performance

Table 15 illustrates the output performance comparisons between springs with 1st b-spline generation function and 2nd b-spline generation function.

Table 15: Output Performance Comparison of Constant-Torque Springs

Spring Specifications	Design Specification	1st Output Results	2nd Output Results
Objective Function Value	N/A	0.5802	0.2609
Optimisation Time (Hours)	N/A	28	38
Constant-torque Value (Nmm)	10	7.68	9.916
Constant-torque Range (Degrees)	50	40	35
Design Space (mm)	600 x 600	250 x 450	400 x 300
Out-of-Plane Thickness (mm)	3	0.8	1
In-Plane Thickness (mm)	1	0.8	0.8
Steady-State Error (%)	5	0.7	5
Maximum Stress (MPa)	150	133	193

The 1st spring is able to achieve a higher constant-torque value as well as higher in constant-torque range. Furthermore, the steady-state error and maximum stress of the 1st spring is also lower than the 2nd spring. 1st spring achieved 0.7% steady-state error, which can provide exceptionally higher constant-torque performance than 2nd spring. In terms of output performance, 1st spring can provide better performance in all aspects compared to 2nd spring. Interestingly, the objective function value of the 2nd spring is lower than the 1st spring. The lower objective function is obtained by 2nd spring due to the fact that the objective function value was calculated by a root-mean-square average method as in the methodology chapter. In detail, a higher maximum torque value of the spring leads to a higher root-mean-square value, which results in a higher objective function value, even though the performance of the constant-torque spring produced by the first b-spline generation function is higher. This can be improved by using root mean square average percentage to ensure that the different amounts of maximum torque will not affect the quality of the result.

6.1.1.2 Physical Performance

The physical performance of both springs is quite similar in terms of weight and size of the spring. The design footprint of the first constant-torque is 6.25% smaller than 2nd spring. The design footprint of 1st constant-torque spring can be reduced further by scaling down the overall size, which results in an increasing amount of maximum stress level. However, due to the low amount of maximum stress of the spring, this process can be done effectively.

6.1.1.3 Computational time

As in Table 14, the computational time of the 1st constant-torque spring is lower than the 2nd constant-torque spring. Due to the nature of the b-spline generation function as well as the difference in the optimisation process in each function. 2nd b-spline generation function requires a higher amount of population in order to perform the task correctly compared to 1st function. This results in an increasing amount of computational time proportionally to the increasing amount of population required.

6.1.2 Results Validation

In this section, the performance of the constant-torque spring design will be verified by using a comparison between experimental results and computational analysis results. Figure 69 illustrates the results of computational analysis and experimental results. Overall, the output function of both results can provide a constant-torque output with a constant-torque range from 40 to 50 rotational degrees. Both constant-torque function illustrates a similar relationship. However, the maximum torque of both results is different. The computational analysis results and experimental results have maximum torque values of 27.245 Nmm and 30 Nmm, respectively. In addition, the maximum torque level error can be calculated as 9.18%.

The error for this comparison occurs because of the material properties used in computational analysis. Steel was selected as a material for this study, and default steel material properties were used for computational analysis. However, the tensile test was not performed to verify the information required for the accurate computational result, which are tensile strength, yield strength, ductility, and Poisson's ratio. This leads to the inaccuracy of the result.

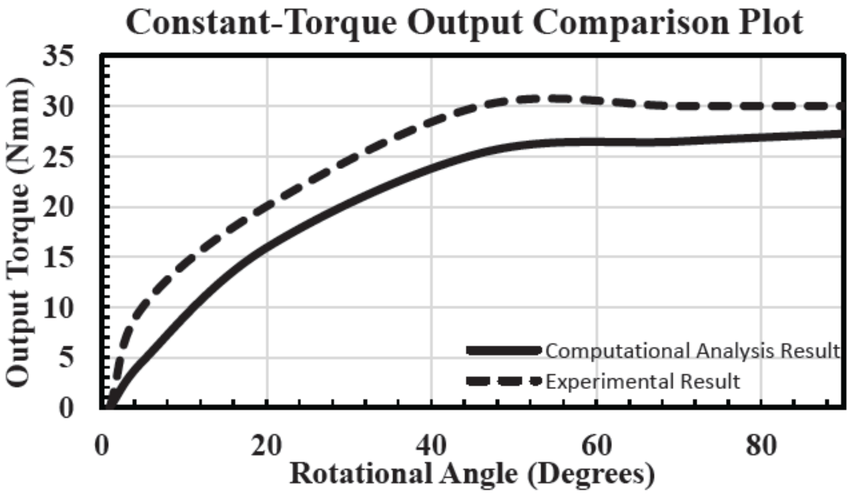


Figure 69: Constant-Torque Spring Output Comparison Plot

6.1.3 Design Performance Comparisons with Other Studies

The performance comparison of constant-torque springs includes a steady-state error, range of motion, and maximum stress level. Maximum torque level is not included due to the difference in design method and material used. The constant-torque spring of study [21] was selected as an example of spring for performance comparisons with a constant-torque spring created by using the 1st b-spline curve generation function is also selected due to the fact that the spring design from [21] illustrates an outstanding performance in every aspect in constant-torque springs studies.

6.1.3.1 Maximum Torque and Design Flatness Value

The example spring from the study [21] was able to achieve a maximum range of motion up to 70 degrees with the design flatness value at 89.1%. Moreover, the spring was able to achieve 50 degrees of rotation of constant-torque range. On the other hand, the constant-torque spring in this study is able to achieve 90 degrees range of motion with 40 degrees constant-torque range. The output flatness value of the spring was lower than the spring example. The constant-torque spring of this study has a flatness value of 99.3 %. Furthermore, the comparison standard can be made by matching the design flatness value of the spring in this study. After the calculation adjustment, by using the exact amount of design flatness value, the constant-torque spring in this study can achieve 60 degrees of constant-torque range from 40 to 100 degrees of rotational angle.

6.1.3.2 Other Aspects

In terms of maximum stress performance, the constant-torque spring in [21] achieved 50 degrees of constant-torque range with a maximum stress value of 79.606 MPa, while the yield strength of the material selected was 233MPa, which illustrates that the spring can achieve 50 degrees of constant-torque with 34.1% of yield strength of the material. On the other hand, the optimal spring design of this study can achieve 60 degrees of constant-torque range with 59% of the yield strength by using a similar amount of design flatness value. However, the maximum stress performance of both designs can be adjusted by an adjustment of the design thickness of the spring. By default, the spring design of this study has multiple thicknesses for the adjustment, which can be done with ease. An adjustment of in-plane-thickness can drastically reduce the maximum stress of the spring. Similarly, the out-of-plane thickness can be adjusted to reduce the overall maximum stress with a minor effect compared to in-plane thickness.

The main advantage of this spring compared to the spring in [21] is due to the single-spline compliant design of the spring. The spring in this study has a higher advantage in terms of manufacturing feasibility and cost of manufacturing and assembly. The generic constant-torque spring in [21] requires multiple assembly components, including bearing and carrier. On the other hand, the spring introduced in this study can be used without other components required due to its single-spline compliant spring design.

6.2 Cosine-Torque Spring

This section will discuss the performance of the optimal designs from the introduced methodology with three different methods. Similar to the previous section, the methods include design performance comparisons, design validation and comparison with other work. The details of each method will be discussed in its section.

6.2.1 Design Performance Comparisons

Similar to constant-torque spring, the performance of 2 spring designs created in this study will be compared in 3 different aspects, which are 1st output performance comparison, 2nd physical performance and 3rd computational time. The higher feasibility spring will be selected for using in design comparison with other studies section.

6.2.1.1 Output Performance Comparison

Table 16 was created by using the results output of the computational analysis of cosine-torque springs created by using two different b-spline generation functions. The Table Illustrates the output performance comparisons between springs with 1st b-spline generation function and 2nd b-spline generation function.

Table 16: Output Performance Comparison of Constant-Torque Springs

Spring Specifications	Design Specification	1st Output Results	2nd Output Results
Objective Function Value	N/A	2.544	0.271
Optimisation Time (Hours)	N/A	16	40
Maximum Torque Value (Nmm)	10	9.66	4.667
Constant-torque Range (Degrees)	50	90	90
Design Space (mm)	600 x 600	400 x 400	300 x 350
Out-of-Plane Thickness (mm)	3	0.8	1
In-Plane Thickness (mm)	1	0.8	0.8
Maximum Stress (MPa)	150	146	167.56

As in Table 15, The objective function value of the spring created by using the 2nd b-spline generation function can achieve a greater amount of performance in terms of the objective function value. In this case, the objective function value can be used as a performance indicator for the output error due to the fact that the maximum torque values for spring designs are similar. The 2nd spring can provide 4.66 Nmm of maximum torque output, which is lower than the torque output of the 1st spring, which can provide 9.66 Nmm torque output. However, the spring created by using the 1st b-spline generation function can achieve lower maximum stress compared to the spring generated from the 2nd b-spline generation function. The safety factors of both springs are 1.712 and 1.492 for the 1st and 2nd spring, respectively. On the other hand, the cosine-torque ranges of both springs are identical at 90 degrees rotation.

6.2.1.2 Physical Performance

The physical performance of both springs is quite similar in terms of weight and size of the spring. In terms of spring size, the 2nd spring requires a 34% lower footprint compared to the 1st. On the other hand, the out-of-plane thickness of the 1st spring is slightly lower than the 2nd spring. Due to the low maximum stress level in both springs, the springs' size can be reduced further by scaling down the overall spring size. This process can result in a slightly increasing amount of maximum stress as well as maximum torque level. However, due to the low amount of maximum stress of the spring, this process can be done effectively for size reduction and maximum torque improvement.

6.2.1.3 Computational Time

Similar to the constant-torque comparison, the computational time of 1st spring is lower than 2nd spring. As mentioned earlier, 1st spring was created based on the 1st b-spline generation function, which has higher performance in computational time due to the nature of the b-spline generation function.

6.2.2 Results Validation

In this section, the design of the cosine-torque output spring will be validated by using computational analysis and experimental results to compare the results' similarities. Figure 70 illustrates the comparison plot between both results. According to the plot, the maximum torque values of both results are noticeably different. The experimental result provided a maximum output torque of 58 Nmm at the maximum rotational angle. On the other hand, the computational analysis results have a maximum torque output value of 31 Nmm. The extensive amount of error occurred by issues occurred during the production process of the spring.

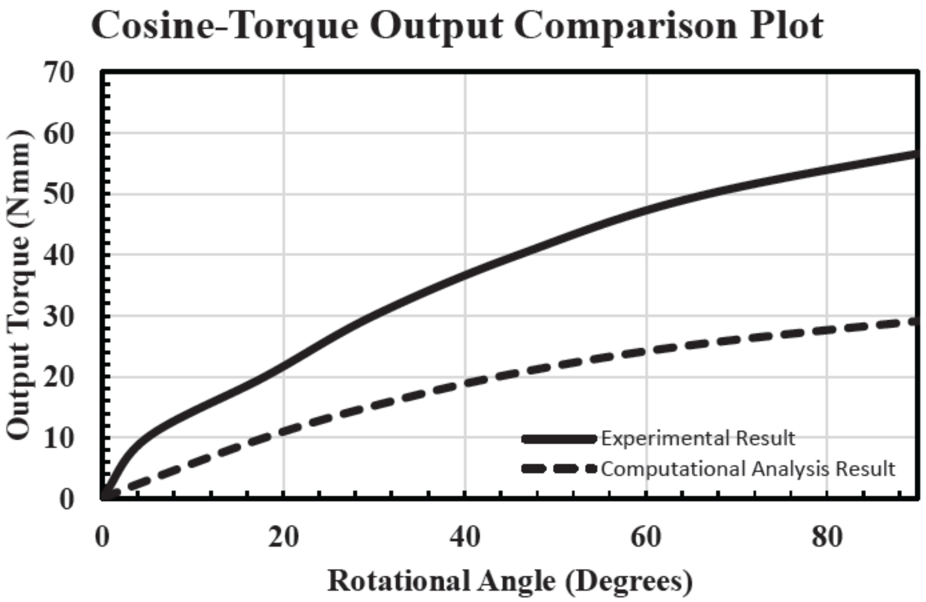


Figure 70: Cosine-Torque Spring Output Comparison Plot

6.2.3 Design Performance Comparisons with Other Studies

In terms of performance comparison in torsion spring for gravity compensation, only range of motion, balancing accuracy, compactness, assembly difficulty, and manufacturing difficulties will be compared due to the lack of study in this area, which makes this comparison requires other types of gravity compensation for the comparison. The cosine-torque spring generated by using the 2nd b-spline generation function is selected for the comparison due to the higher output performance of the spring.

In order to create a precise comparison in terms of the design performance of the spring, an additional analysis is required. A gravity-compensated output plot is required to compare the accuracy of the cosine-torque spring for gravity compensation. Figure 71 illustrates a performance plot of the cosine-torque spring for gravity compensation application.

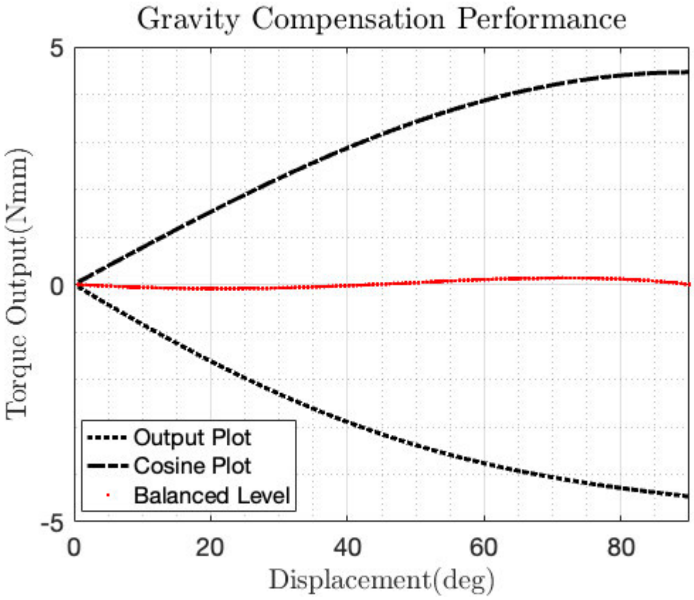


Figure 71: Cosine-Spring Performance in Gravity Compensation Application

The Figure includes three different plots, which are 1st cosine-torque spring output (Output Plot), 2nd cosine plot, which represents the torque required for perfect balancing compensation and the balanced level plot, which illustrates the balancing compensation performance by using an addition of output plot and cosine plot. As represented in the Figure, the cosine-torque spring can provide up to 90 degrees of rotation for gravity compensation, while the balanced level of the spring has a 0.46% RMS error compensation error throughout 90 degrees of rotation.

6.2.3.1 Output Performances

Cosine-torque spring can provide up to 90 degrees of cosine function. However, other types of mechanisms for gravity compensation can provide a higher range of motion. A magnetic gravity compensator from [27] can provide up to 180 degrees of gravity compensation. Similarly, a gravity compensator created by using gear spring modules in the study [28] was also able to achieve a 180 compensation range. Furthermore, multiple studies show a result of multi-degree-of-freedom gravity compensation. In this study, the torsion spring was designed for one degree of freedom movement, which has a disadvantage in this perspective. However, the compensation accuracy of the cosine-spring is outstanding. The spring is able to achieve 0.46% of RMS error. Compared to other studies, the spring can achieve higher performance in terms of compensation accuracy.

6.2.3.2 Other Aspects

The drawback of the generic gravity compensation method is the complexity of the mechanism. A general approach for gravity compensation requires multiple components such as gears, bars, weight, and elastic modules to create a gravity compensator [29]. Multiple amounts of components lead to higher design complexity as well as higher weight, which often leads to higher difficulty in manufacturing and installation. Some of the gravity compensators rely on using magnetic force to create a compensator that has high complexity and difficulty in control.

The cosine-torsion spring can be manufactured with ease due to the design of the spring and the low number of parts. Moreover, it also reduces the requirement of assembly, which can help reduce assembly costs and time. The generic gravity compensation design by static balancing is also lacking in compactness and lightweight design. In contrast, a cosine-torque spring can be provided 1 degree of freedom 90 degrees of the cosine-torque range.

Furthermore, the cosine-torque spring design is based on a compliant mechanism. The compliant mechanisms offer advantages for the spring, including a reduction in higher precision control and lower maintenance cost. The mechanism precision control advantage in compliant mechanism spring is obtained by the unessential of connector and hinged, which often requires a slot between different parts. Moreover, the hinged and connector in the generic mechanism require lubricant over time to reduce friction between parts as well as to prevent wear and tear of the component. In contrast, the spring created in this study can be used as a one-piece mechanism that does not require any unnecessary parts, which can result in overall cost reduction.

Even though cosine-torque spring in this study may not be able to outperform other mainstream gravity compensators in terms of performance in some perspectives, such as the range of motion. However, the spring can provide an outstanding cosine output function and can be further developed for improvement in performance.

6.3 Computational Time

Compared to other studies, the computational time of this study exceeded the average time consumption. The causes of low performance are CPU cores limitation and software capability. In this study, ANSYS software was selected for Finite Element Analysis and geometry generation. Most of the computational time in each optimisation iteration was spent on the geometric generation of SpaceClaim software as well as Finite Element Analysis. On the other hand, the computational time consumed by MATLAB was only 5% of the total computational time. Moreover, the number of CPU cores used for the simulation is limited to 2 cores.

6.4 Error in Experimental Result

The cosine-spring geometry with large footprints was produced by using a water jet cutting machine. As mentioned in the previous section, the issue occurred during the spring production process. The spring cannot support its weight during the process, which results in spring deformation during the process, which creates errors in spring geometry. The errors include a thickness difference in some parts throughout the geometry as well as a plastic deformation of the spring, which results in changes in material properties. Moreover, the experimental result is also influenced by the difference in material properties. Similar to the constant-torque spring, the material used for computational analysis is set as a default steel material property, which is different from the steel used for spring production.

The output error from the production process only occurred to the cosine-torque output spring, which has an error in geometry. This error can be overcome by reducing the spring's size to reduce the weight of the spring. Furthermore, the in-plane thickness of the spring can be increased to improve the stiffness of the spring. This can result in a lower amount of error production. However, the maximum stress of the spring will be increased by using both methods. Furthermore, conducting a bulking test for accurate material properties can help improve the output result.

Chapter 6: Conclusion and Future Work

This thesis presents a nonlinear compliant torsion spring design methodology using a combination of design optimisation, finite element analysis and compliant design. The springs were created based on a single b-spline curve. The design goal of this study includes two of the spring types, which are constant-torque and cosine-torque springs. The optimal spring designs illustrate high performance in output accuracy and design compactness. The spring performance was proven to be feasible for constant-torque and cosine-torque designs, which can be used in multiple applications such as gravity compensation and human-robot interaction. Above all, the performance of the springs illustrates the reliability and the power of the introduced methodology for creating a single b-spline curve spring.

This thesis provides the first attempt at developing a nonlinear torsional compliant spring with a single b-spline curve for 2 of the applications, which are constant-torque and cosine-torsion springs. The previous research in nonlinear torsion spring design methodology is based on generic complaint torsion spring design, and none of the studies attempted to create cosine-torsion springs before. However, similar research has been done for constant force design with a single spline curve. The spring design can reduce the parts required in torsion springs, such as shaft and bearing, which can further reduce the maintenance cost and manufacturing difficulty as well as assembly cost. The new approach of creating nonlinear torsion springs and springs for gravity compensation can provide awareness of this possible design solution, which can increase the amount of research in this field.

6.5 Limitations of the Research and Methodology

Limitations of this research and methodology can be divided into two aspects, which are Limitations in optimisation parameters and Limitations in result verification. Optimisation parameters in this study only included some of the adjustable parameters of the spring. The performance of the methodology may improve by increasing the adjustable optimisation parameter. On the other hand, the result of this study was only verified by using finite element analysis as well as practical experiments. In practical, the springs still requires other experiments for performance verification fatigue test to verify the lifetime usage of the spring.

6.6 Future work

6.6.1 Improvement of Optimisation Algorithms and its Parameter

In this research, optimisation parameters in each optimisation question are only limited to spline curve geometry. The scaling optimisation process is avoidable by including other aspects of the spring design as an optimisation parameter, including scaling factors, out-of-plane thickness and in-plane thickness. An optimisation algorithm with a specific additional amount of optimisation parameters can help improve searching performance for optimal solutions without double optimisation, which can help reduce the overall computational time. Furthermore, a combination of optimisation algorithms can be created to help achieve the optimal result. A combination of a Genetic algorithm and a gradient-based algorithm can help reduce the optimisation time. In detail, the Genetic algorithm has an advantage in early optimisation iteration in finding the optimal design. On the other hand, the Gradient-based algorithm is suitable for searching for optimal design by using the best result from GAs for searching the optimal result. Gradient-based algorithms can approach the optimal results efficiently in later generations compared to GAs.

6.6.2 Improvement of Spring Synthesis Approach

In terms of computational time of this research, 95% of the computational time was spent on generating 3D models of the springs and Finite Element Analysis. In this study, ANSYS is an application used for both processes. The computational time of design questions exceeded the average computational time of other studies that use other commercial software such as COMSOL and ABAQUS. Software change is considered in future research to reduce the computational time. However, implementing the FEA algorithm directly into MATLAB program is the best option to reduce the computational time further. The FEA algorithm can help eliminate the use and limitation of FEA software and reduce computational resources for multiple software uses.

6.6.3 Compliant Torsion Spring Design Improvement

Compliant torsion spring design requires further research and improvement. As a result of this study, the cosine-torque and constant-torque spring designs are able to satisfy the criteria by using the design methodology introduced in this study. Furthermore, as mentioned previously, gravity compensation design at present requires a bulky design with multiple components, which leads to complexity and difficulty in manufacturing. The literature review chapter illustrates the research gap in torsion springs for gravity compensators. Further research on combining generic compliant torsion springs, as well as the design methodology, can be used for creating gravity compensation springs. A combination of the methodology and compliant torsion spring design can help achieve higher performance of cosine-torsion spring compared to the spring in this study. The generic compliant torsion spring, which is constructed by a combination of shaft, ring and compliant mechanism modules, can generate a higher amount of output torque with a smaller weight and size than the spring in this study. Above all, The improvement may include increasing the cosine-torque range, which is the only lacking aspect of the spring introduced in this study. A combination of spring designs can help increase the flexibility and complexity of the design and optimisation, which can help achieve viable results and further improvement, such as bidirectional compliant torsion spring design for gravity compensation application.

6.6.4 Practical Experiments

Practical experiments are required to verify the performance of the spring in practical use. This study conducted a torsion test for both spring types, which can help validate the spring performance in practice. However, this study is still considered as lacking in other aspects of practical experiments. Experiments such as loading-unloading tests and fatigue tests are necessary. Loading-unloading test can help determine the performance of the spring compared to FEA results in both directions. Moreover, in some applications, an equal amount of loading-unloading output is required, such as gravity compensation. On the other hand, a fatigue test is necessary for compliant mechanism design. Compliant mechanisms use geometric nonlinearity, which results in repetitive bending in the same area. This can lead to fatigue failure. The fatigue test is required in compliant mechanism studies to verify mechanism lifetime.

Chapter 7: References

[1] L. L. Howell, S. P. Magleby, and B. M. Olsen, *Handbook of Compliant Mechanisms* (Handbook of Compliant Mechanisms). 2013.

[2] G. K. Ananthasuresh and S. Kota, "Designing compliant mechanisms," *Mechanical Engineering,* Article vol. 117, no. 11, pp. 93-96, 1995. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-0029406305&partnerID=40&md5=066aa02d04414ee2f8d4f3c813d43a0a.

[3] O. Altun, P. Wolniak, I. Mozgova, and R. Lachmayer, "AN ANALYSIS OF SCALING METHODS FOR STRUCTURAL COMPONENTS IN THE CONTEXT OF SIZE EFFECTS AND NONLINEAR PHENOMENA," *Proceedings of the Design Society: DESIGN Conference,* vol. 1, pp. 797-806, 05/01 2020, doi: 10.1017/dsd.2020.320.

[4] P. Soroushian, H. Chowdhury, and A. Nossoni, "Design and experimental verification of pseudoelastic-based constant-force springs," *Journal of Intelligent Material Systems and Structures,* Article vol. 14, no. 8, pp. 475-481, 2003, doi: 10.1177/104538903038018.

[5] V. Potkonjak, B. Svetozarevic, K. Jovanovic, and O. Holland, "Anthropomimetic robot with passive compliance - Contact dynamics and control," in *2011 19th Mediterranean Conference on Control and Automation, MED 2011*, 2011, pp. 1059-1064, doi: 10.1109/MED.2011.5983000. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-80052367590&doi=10.1109%2fMED.2011.5983000&partnerID=40&md5=849ad66b8f9b2e8f12fe3954b5ec57a2

[6] O. Zienkiewicz, R. Taylor, and J. Z. Zhu, *The Finite Element Method: its Basis and Fundamentals: Seventh Edition* (The Finite Element Method: its Basis and Fundamentals: Seventh Edition). 2013, pp. 1-714.

[7] G. Carpino, D. Accoto, F. Sergi, N. L. Tagliamonte, and E. Guglielmelli, "A novel compact torsional spring for series elastic actuators for assistive wearable robots," *Journal of Mechanical Design, Transactions of the ASME,* Article vol. 134, no. 12, 2012, Art no. 121002, doi: 10.1115/1.4007695.

[8] D. Accoto, G. Carpino, F. Sergi, N. L. Tagliamonte, L. Zollo, and E. Guglielmelli, "Design and characterization of a novel high-power series elastic actuator for a lower limb

robotic orthosis," *International Journal of Advanced Robotic Systems,* Article vol. 10, 2013, doi: 10.5772/56927.

[9] F. Negrello *et al.*, "Design and characterization of a novel high-compliance spring for robots with soft joints," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, 2017, pp. 271-278, doi: 10.1109/AIM.2017.8014029. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85028755991&doi=10.1109%2fAIM.2017.8014029&partnerID=40&md5=d947ab642ca06aa99eb284cf4cb8420d

[10] J. C. Meaders and C. A. Mattson, "Optimization of near-constant force springs subject to mating uncertainty," *Structural and Multidisciplinary Optimization,* Article vol. 41, no. 1, pp. 1-15, 2010, doi: 10.1007/s00158-009-0410-4.

[11] C. B. W. Pedersen, N. A. Fleck, and G. K. Ananthasuresh, "Design of a compliant mechanism to modify an actuator characteristic to deliver a constant output force," *Journal of Mechanical Design, Transactions of the ASME,* Article vol. 128, no. 5, pp. 1101-1112, 2006, doi: 10.1115/1.2218883.

[12] C. H. Liu, M. C. Hsu, T. L. Chen, and Y. Chen, "Optimal Design of a Compliant Constant-Force Mechanism to Deliver a Nearly Constant Output Force over a Range of Input Displacements," *Soft Robotics,* Article vol. 7, no. 6, pp. 758-769, 2020, doi: 10.1089/soro.2019.0122.

[13] Y. Wei and Q. Xu, "Design of a new passive end-effector based on constant-force mechanism for robotic polishing," *Robotics and Computer-Integrated Manufacturing,* Article vol. 74, 2022, Art no. 102278, doi: 10.1016/j.rcim.2021.102278.

[14] P. Bilancia and G. Berselli, "Design and testing of a monolithic compliant constant force mechanism," *Smart Materials and Structures,* Article vol. 29, no. 4, 2020, Art no. 044001, doi: 10.1088/1361-665X/ab6884.

[15] C. Jutte and S. Kota, "Design of Nonlinear Springs for Prescribed Load-Displacement Functions," *Journal of Mechanical Design - J MECH DESIGN,* vol. 130, 08/01 2008, doi: 10.1115/1.2936928.

[16] N. Fawazi, I. H. Yang, J. S. Kim, J. Y. Lee, H. S. Kim, and J. E. Oh, "An inverse algorithm of nonlinear load-displacement for a slotted disc spring geometric design," *International Journal of Precision Engineering and Manufacturing,* Article vol. 14, no. 1, pp. 137-145, 2013, doi: 10.1007/s12541-013-0019-9.

[17] M. Paredes and E. Rodriguez, "Optimal design of conical springs," *Engineering with Computers,* Article vol. 25, no. 2, pp. 147-154, 2009, doi: 10.1007/s00366-008-0112-3.

[18] S. Liu, G. Peng, Z. Li, W. Li, K. Jin, and H. Lin, "Design and experimental study of an origami-inspired constant-force mechanism," *Mechanism and Machine Theory,* Article vol. 179, 2023, Art no. 105117, doi: 10.1016/j.mechmachtheory.2022.105117.

[19] F. Mei, S. Bi, L. Chen, and H. Gao, "A novel design of planar high-compliance joint in variable stiffness module with multiple uniform stress leaf branches on rigid-flexible integral linkage," *Mechanism and Machine Theory,* Article vol. 174, 2022, Art no. 104889, doi: 10.1016/j.mechmachtheory.2022.104889.

[20] P. H. Kuo and A. D. Deshpande, "A novel joint design for robotic hands with humanlike nonlinear compliance," *Journal of Mechanisms and Robotics,* Article vol. 8, no. 2, 2016, Art no. 021004, doi: 10.1115/1.4031300.

[21] T. V. Phan, H. T. Pham, and C. N. Truong. *Design and Analysis of a Compliant Constant-Torque Mechanism for Rehabilitation Devices, Springer Proceedings in Materials*, vol. 6, pp. 541-549, 2020.

[22] P. Wang, S. Yang, and Q. Xu, "Design and Optimization of a New Compliant Rotary Positioning Stage with Constant Output Torque," *International Journal of Precision Engineering and Manufacturing,* Article vol. 19, no. 12, pp. 1843-1850, 2018, doi: 10.1007/s12541-018-0213-x.

[23] C. W. Hou and C. C. Lan, "Functional joint mechanisms with constant-torque outputs," *Mechanism and Machine Theory,* Article vol. 62, pp. 166-181, 2013, doi: 10.1016/j.mechmachtheory.2012.12.002.

[24] H. N. Prakashah and H. Zhou, "Synthesis of constant torque compliant mechanisms," *Journal of Mechanisms and Robotics,* Article vol. 8, no. 6, 2016, Art no. 064503, doi: 10.1115/1.4034885.

[25] B. P. Reddy and H. Zhou, "Synthesizing bidirectional constant torque compliant mechanisms," in *ASME International Mechanical Engineering Congress and Exposition, Proceedings (IMECE)*, 2017, vol. 4A-2017, doi: 10.1115/IMECE201770333. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85041007784&doi=10.1115%2fIMECE201770333&partnerID=40&md5=b84a21bc16e4f6da736b454affe20e9b

[26] J. R. R. A. Martins and A. Ning, *Engineering Design Optimization*. Cambridge University Press, 2021.

[27] B. v. Ninhuijs, J. W. Jansen, B. L. J. Gysen, and E. A. Lomonova, "Multi-Degree-of-Freedom Spherical Permanent-Magnet Gravity Compensator for Mobile Arm Support Systems," *IEEE Transactions on Industry Applications,* vol. 50, no. 6, pp. 3628-3636, 2014, doi: 10.1109/TIA.2014.2313066.

[28] V. L. Nguyen, C.-Y. Lin, and C.-H. Kuo, "Gravity compensation design of Delta parallel robots using gear-spring modules," *Mechanism and Machine Theory,* vol. 154, p. 104046, 2020/12/01/ 2020, doi: https://doi.org/10.1016/j.mechmachtheory.2020.104046.

[29] V. Arakelian, "Gravity compensation in robotics," *Advanced Robotics,* vol. 30, no. 2, pp. 79-96, 2016/01/17 2016, doi: 10.1080/01691864.2015.1090334.