



# RAM

● ROBOTICS  
AND  
MECHATRONICS

## CONTROL OF A SYSTEM BASED ON A 3D PRINTED TORQUE SENSOR

R.A. (Rogier) Heeg

MSC ASSIGNMENT

**Committee:**

prof. dr. ir. G.J.M. Krijnen  
ir. D. Kosmas  
Y.P. Wotte, MSc  
dr. ir. R.G.K.M. Aarts

February, 2023

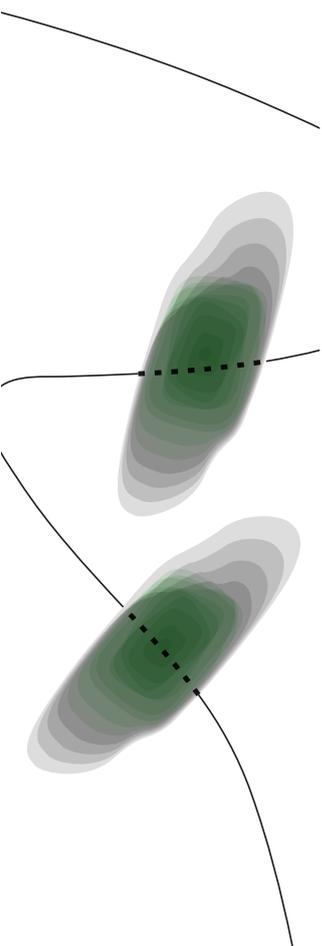
006RaM2023  
Robotics and Mechatronics  
EEMCS  
University of Twente  
P.O. Box 217  
7500 AE Enschede  
The Netherlands

UNIVERSITY  
OF TWENTE.

TECHMED  
CENTRE

UNIVERSITY  
OF TWENTE.

DIGITAL SOCIETY  
INSTITUTE





## Summary

The technology of Additive Manufacturing started in the early 1980's but only recently has risen in popularity that resulted in new and optimized processes and introductions to methods like multi-material additive manufacturing. This method paved way for a new field of research on 3D printed sensors using conductive filaments. The main problem with 3D printed sensors is the non-linear behavior in their response which are mainly contributed to hysteresis, drift and creep. This work present the process of design, application and validation of an 3D printed torque sensor using strain gauges. Inspiration was drawn from commercially available mechanical torque sensors, such that a circular sensor design was made with identical and opposing strain gauges that make differential measurements possible. The characterization of the sensor yielded satisfying results, showing minimal non-linear behavior when excited with an low torque, low frequent harmonic input in a controlled setting. For the application a pendulum setup was made that was integrated with the sensor design. The validation of the sensor showed a response of the same magnitude as the input but with larger error. It was found that this error is mainly contributed by the phase lag, due to the real time low pass filtering of the data, and hysteresis. It is thus recommended that, for real time use of the sensor output, research is done to properly filtering real time sensor data and compensation of hysteresis.



---

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Context . . . . .	1
1.2	Motivation & Current research . . . . .	1
1.3	Problem . . . . .	1
1.4	Research questions . . . . .	1
1.5	Report structure . . . . .	2
<b>2</b>	<b>Literature</b>	<b>3</b>
2.1	Introduction . . . . .	3
2.2	Sensing Principle . . . . .	3
2.3	Fabrication . . . . .	7
2.4	Control . . . . .	10
2.5	Conclusion . . . . .	13
<b>3</b>	<b>Design</b>	<b>15</b>
3.1	Introduction . . . . .	15
3.2	Application . . . . .	15
3.3	Structural Analysis . . . . .	17
3.4	Sensor-Cable Interface . . . . .	21
3.5	CAD Model . . . . .	22
3.6	Fabrication . . . . .	24
3.7	Conclusion . . . . .	26
<b>4</b>	<b>Characterisation</b>	<b>28</b>
4.1	Introduction . . . . .	28
4.2	Components . . . . .	28
4.3	Measurements . . . . .	31
4.4	Results . . . . .	31
4.5	Conclusion . . . . .	37
<b>5</b>	<b>Control</b>	<b>38</b>
5.1	Introduction . . . . .	38
5.2	Dynamics . . . . .	38
5.3	Analysis . . . . .	50
5.4	Realisation . . . . .	52
5.5	Conclusion . . . . .	58
<b>6</b>	<b>Discussion &amp; Future Work</b>	<b>60</b>

---

6.1 Design . . . . .	60
6.2 Characterisation . . . . .	60
6.3 Control . . . . .	60
<b>7 Conclusion</b>	<b>62</b>
<b>A Section Modulus</b>	<b>63</b>
<b>B Pairewise Weighing</b>	<b>64</b>
<b>C SolidWorks Drawings</b>	<b>65</b>
<b>D DEWE Code</b>	<b>67</b>
<b>E MATLAB Code</b>	<b>69</b>
<b>F Measurement Results</b>	<b>72</b>
<b>G Design Iterations</b>	<b>87</b>
<b>H NEMA17 Datasheet</b>	<b>90</b>
<b>I SMAC Datasheet</b>	<b>91</b>
<b>Bibliography</b>	<b>92</b>

## List of Figures

2.1	Schematic of parallel-plate capacitor with dielectric material [1]. On the right its schematic symbol. . . . .	4
2.2	Schematic of capacitor with with changing overlapping area [2]. . . . .	5
2.3	Typical strain gauge where electrical resistance changes in proportion to the applied strain on the specimen [3]. . . . .	6
2.4	Schematic overview of FFF [4]. A nozzle deposits molted material on a surface creating a layer. In this image the nozzle is stationary and the bed position is controllable. This can also be the other way around. . . . .	8
2.5	The Diabase Engineering H-Series Multi-Material 3D Printer. . . . .	9
2.6	Left: Direct-drive extrusion method [5]. Right: Rotary multi-toolhead system for filament switching [5] . . . . .	9
2.7	Commonly used friction model from [6]. . . . .	13
3.1	Early sensor design from Lou [7]. With outer and inner ring diameter respectively being 78 mm and 10 mm. Strain gauges are placed on the beams near the inner ring (both sides). . . . .	15
3.2	(a) Well aligned sensor to the motor shaft. (b) Misaligned sensor to motor shaft, induces a bending motion on the sensor which will influence the sensor response. . . . .	16
3.3	Schematic overview of a rigid body pendulum driven by a stepper motor around $o$ . . . . .	16
3.4	Left: FBD of a single free end cantilever with end moment [8]. Middle: Moment diagram of this equivalent model. Right: Bending diagram with largest deformation at the end point. . . . .	18
3.5	Finite element method (FEM) results from [7]. Largest stress is concentrated near the inner ring (red). . . . .	19
3.6	Left: FBD of a single pinned support cantilever [8]. Middle: Moment diagram of this equivalent model. Right: Bending diagram that represent an s-shape. . . . .	19
3.7	Stress graph depicted in red with $\sigma_{\max}$ at length $L$ and bending diagram from $0 < x < L$ depicted in black (exaggerated). With positive stress at the right side where the moment is applied and negative stress at the fixed side. . . . .	20
3.8	Beams with slots. gray area is the material, white area the removed geometry to decrease section modulus $Z$ . Top: Model used by Lou et al. [7] to increase strain distribution. Bottom: Simplified model for analysis where $Z$ has three constant values. . . . .	20
3.9	$Z, M$ graph of the simplified model with three different values of $Z$ . . . . .	21
3.10	Silver epoxy connection for the attachment of wires to the strain gauges of an early iteration of the sensor. . . . .	21
3.11	A screw and wire connection on the sensor. . . . .	22
3.12	(a) Final design of the sensor with two sets of strain gauges on opposing sides of a spoke (left & right spokes). (b) Inner connection of the screw holes to the gauges on the spokes. Green material is PLA and black the conductive ProtoPasta. . . . .	22

3.13	A detailed look at the strain gauges one one side of the sensor, with screws placed. (a) Shows the the solid structure. (b) The structure is hidden such that the connection is shown. The gauges loop from one side of the sensor to the other. . . .	23
3.14	Pendulum with integrated sensor design CAD. . . . .	24
3.15	(a) The CURA orientation showing layering direction. (b) Resulting print of the model. . . . .	25
3.16	Pendulum with integrated sensor design print. . . . .	26
4.1	DEWE-43-A DAQ from [9]. . . . .	29
4.2	Half-bridge configuration of the connector cables with the sensor and a 10 K potentiometer [9]. . . . .	29
4.3	Annotated overview of the measurements setup. . . . .	30
4.4	(a) Sensor mounted on the aluminum mounting plate. (b) Sensor with extension (dark green) such that applied torque is amplified. SMAC attachment mount (red) for in-plane attachment of SMAC to avoid parasitic movements. . . . .	31
4.5	Single-Sided Amplitude Spectrum of gauge 1 (a) and gauge 2 (b) for an input of $f_{\sin} = 0.2\text{Hz}$ . . . . .	32
4.6	Sensor response of both individual gauges. In blue the unfiltered data, in orange the filtered data. Sensor is excited with a 2 N, 0.2 Hz sine wave. . . . .	32
4.7	Sensor response of both individual gauges. In blue the unfiltered data, in orange the filtered data. Sensor is excited with a 12N 8Hz sine wave. . . . .	33
4.8	Load-cell response, converted to force, plotted against time for both data sets. . .	33
4.9	Unfiltered sensor response of individual gauges plotted against torque. (a) & (b) respectively correspond to the first and second gauge of the 2N, 0.2Hz set. (c) & (d) respectively correspond to the first and second gauge of the 12N, 8Hz set. . .	34
4.10	(a) & (c) present the unfiltered differential of both data sets & (b) & (d) present the filtered differential with a linear fit plotted against the applied torque. . . . .	35
4.11	Difference plot between filtered data and fit for both measurements against time. (a) Data set 182, (b) data set 262. . . . .	36
4.12	(a) & (b) present the filtered differential relative change in resistance with a linear fit plotted against the applied torque, respectively for both data sets. . . . .	37
5.1	Schematic overview of a rigid body pendulum driven by a stepper motor around $o$ . . . . .	38
5.2	(a) Simulink model with the EOM implemented. (b) Response of the plant to zero input torque. . . . .	41
5.3	Block scheme of the simulated system. . . . .	41
5.4	Simulink model based on the block scheme from Figure 5.3. Both controller C1 and C2 are bundled in the Position Control Loop block and the disturbance is added before the plant. . . . .	42
5.5	Step response of the simulated system. A step of $\frac{1}{2}\pi$ is made. . . . .	42
5.6	Resulting plot for the controlled system with time on the x-axis and the angle $\theta$ on the y-axis. Blue presents the measured angle $\theta_m$ and dashed red the desired angle $\theta_d$ . . . . .	43
5.7	Simulink model of the system with interaction control. . . . .	43

5.8	Resulting plots for the controlled system with the interaction control layer. (a) With time on the x-axis and the angle $\theta$ on the y-axis. Blue presents the measured angle $\theta_m$ and dashed red the desired angle $\theta_d$ . (b) Shows the corresponding controller output. . . . .	44
5.9	In blue a continuous-time signal, defined for all values of time. In red the same signal discretized, defined only at set time steps and considered constant until the next. . . . .	45
5.10	(a) Comparison between $\sin(\theta)$ and $\theta$ . (b) Approximation error. Both axis are in radians. . . . .	46
5.11	The system at three positions $(0, 0)$ , $(\frac{1}{4}\pi, 0)$ and $(-\frac{1}{4}\pi, 0)$ . The blue dotted line represents the position of the pendulum using the non-linear sine function of the angle $\theta$ . The red lines represent three linear approximation around these positions. . . . .	46
5.12	(a) Simulink model applying a zero torque input to the continuous-time system, the continuous-time s-domain transfer function and the discrete z-domain transfer function. (b) System response where the angle is shown in time. The initial position is $(0.1, 0)$ . . . . .	49
5.13	Pole zero maps and their stability regions for both continuous-time and discrete-time LTI transfer functions. . . . .	50
5.14	Response of a continuous-time LTI system to pole locations [10]. . . . .	50
5.15	Poles of the continuous-time LTI transfer function. . . . .	51
5.16	Step response of the continuous-time LTI transfer function. . . . .	51
5.17	Pole (red x) zero (red o) map of the discrete LTI transfer function. . . . .	52
5.18	Step response of the discrete LTI transfer function. . . . .	52
5.19	Left: Overview of the application setup. Right: Zooming in shot of the motor, strain gauges, the cable connection and the Mechaduino (red). . . . .	53
5.20	Sensor output plotted against the calculated torque $\tau_c$ . (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz. . . . .	56
5.21	Comparison between $\tau_c$ (blue) and $\tau_m$ (red) plotted against time. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz. . . . .	56
5.22	Error plot corresponding to Figure 5.21. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz. . . . .	57
5.23	Comparison between unfiltered $\tau_m$ (blue) and filtered $\tau_m$ (red) plotted against time. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz. . . . .	57
5.24	Unfiltered (in post processing) comparison between $\tau_c$ (blue) and $\tau_m$ (red) plotted against time. A phase lag is present in the response. . . . .	58
5.25	Error comparison between the phase lagged $\tau_m$ (blue) and compensated $\tau_m$ (red). . . . .	58
A.1	Section modulus of various geometries. [11] . . . . .	63
C.1	Detailed drawing of sensor design showing internal structures aswell dimensions. . . . .	65
C.2	Detailed drawing of sensor design showing internal structures aswell dimensions. . . . .	66
E.1	Measurement results 2N sine wave, 0.2 & 0.5 Hz. . . . .	72
E.2	Measurement results 2N sine wave, 1 & 2 Hz. . . . .	73
E.3	Measurement results 2N sine wave, 4 & 8 Hz. . . . .	74

---

F4	Measurement results 4N sine wave, 0.2 & 0.5 Hz. . . . .	75
F5	Measurement results 4N sine wave, 1 & 2 Hz. . . . .	76
F6	Measurement results 4N sine wave, 4 & 8 Hz. . . . .	77
F7	Measurement results 8N sine wave, 0.2 & 0.5 Hz. . . . .	78
F8	Measurement results 8N sine wave, 1 & 2 Hz. . . . .	79
F9	Measurement results 8N sine wave, 4 & 8 Hz. . . . .	80
F10	Measurement results 10N sine wave, 0.2 & 0.5 Hz. . . . .	81
F11	Measurement results 10N sine wave, 1 & 2 Hz. . . . .	82
F12	Measurement results 10N sine wave, 4 & 8 Hz. . . . .	83
F13	Measurement results 12N sine wave, 0.2 & 0.5 Hz. . . . .	84
F14	Measurement results 12N sine wave, 1 & 2 Hz. . . . .	85
F15	Measurement results 12N sine wave, 4 & 8 Hz. . . . .	86
G.1	Design iterations of the sensor V0.1 to V0.4. . . . .	87
G.2	Design iterations of the sensor V0.5 to V0.8. . . . .	88
G.3	Design iterations of the sensor V0.9 to V0.12. . . . .	89
H.1	Datasheet of the NEMA17. This motor is used in the setup. . . . .	90

---

## List of Tables

2.1	Properties of ABS and PLA [12] . . . . .	10
2.2	5-point scaling. . . . .	13
2.3	Hierarchical weighing of aspects of sensing principles. . . . .	14
3.1	Table of design requirements for sensor design. . . . .	17
3.2	Table of design iterations and their main changes. . . . .	24
3.3	Printing parameters . . . . .	25
5.1	Variables of the pendulum system. . . . .	40
B.1	Pairewise weighing of aspects of the sensing principles . . . . .	64
B.2	Hierarchical weighing of factors of sensing principle. Final weight is based on the rank over a 100% total weight. . . . .	64
I.1	Datasheet of the SMAC LCA25-050-15-6 [13]. . . . .	91

## Table of Abbreviations

<b>AM</b>	Additive Manufacturing
<b>SLA</b>	Sterolithography
<b>FFF</b>	Fused Filament Fabrication
<b>MMAM</b>	Multi-Material Additive Manufacturing
<b>NIFTy</b>	Nature Inspired Fabrication and Transduction
<b>MEMS</b>	Micro Electrical Mechanical Systems
<b>EMI</b>	Electromagnetic Interference
<b>CDC</b>	Capacitance to Digital Converter
<b>ADC</b>	Analog to Digital Converter
<b>SLS</b>	Selective Lase Sintering
<b>PLA</b>	Polylactic Acid
<b>ABS</b>	Acrylonitrile Butadiene Styrene
<b>TPU</b>	Thermoplastic Polyurethane
<b>DOF</b>	Degrees of Freedom
<b>FBD</b>	Free Body Diagram
<b>FEM</b>	Finite Element Method
<b>CAD</b>	Computer Aided Design
<b>COM</b>	Centre of Mass
<b>DAQ</b>	Data Acquisition
<b>EMCL</b>	Embedded Motion Control Library
<b>GUI</b>	Graphical User Interface
<b>Hz</b>	Hertz
<b>N</b>	Newton
<b>EOM</b>	Equations of Motion
<b>NTI</b>	Non-linear Time Invariant
<b>LTI</b>	Linear Time Invariant
<b>ZOH</b>	Zero-Order Hold
<b>PCB</b>	Programmable Circuit Board

# 1 Introduction

## 1.1 Context

The technology of Additive Manufacturing (AM) started in the 1980's with the first patented method of layer by layer rapid prototyping being Stereolithography (SLA) [14]. In SLA, photo-sensitive 3D shapes were made by placing layers of photosensitive resin and curing them using a laser [15]. It was not until 1989 that Scott Crump patented the first Fused Filament Fabrication (FFF) method where a filament is melted within a heated nozzle and directly extruded on a surface [16]. It was after this patent became public domain in 2006 that its popularity increased rapidly [17]. This popularity resulted in new and optimized processes of FFF such as the method of multi-material additive manufacturing (MMAM), which adds more complexity and functionality [18]. This paved way for a new area of research where conductive filaments could be printed within conventional 3D printed designs, leading to a novel research field of 3D printed electronics and sensors [19].

## 1.2 Motivation & Current research

This master thesis research is done at the NIFTy (Nature Inspired Fabrication and Transduction) group at the University of Twente. NIFTy focuses on the use of AM to develop systems with integrated sensory functionalities. Current research at NIFTy feature embedded sensors in soft and rigid body robotics using the multi material 3D printing method. Such as flow sensors [20], tactile sensors [21] [22], shear and normal force sensors [23] [24] to name a few.

Adding to this research field, this work focuses on the use of FFF to design and fabricate a torque sensor and to integrate this sensor in the application to be controlled, showcasing the feasibility.

## 1.3 Problem

The main problem with 3D printed sensors is the non-linear behavior of their responses. The main contributors to this non-linearity are drift, creep and hysteresis [22]. FFF has limited materials that can be printed and it produces non isotropic structures (meaning not identical in all directions). Conventional sensors are fabricated specifically to minimize these non-linearity contributors using compound materials with highly predictable behavior, materials that are unaffected by changes in environment and isotropic structures. The aim of this research is to design a torque sensor in such a way that this non-linear behavior is minimized and the sensor is able to be used without the need for a complicated integration process.

## 1.4 Research questions

The main objective of this thesis is to design & fabricate a 3D printed torque sensor and use it in a control application as a proof of concept. The objective is divided in three sub-objectives:

1. Which sensing technique is best suited for a 3D printed torque sensor?
2. How should a 3D printed torque sensor be designed such that the non-linearity of the sensor response is minimised?
3. What application is suitable for demonstrating, characterizing and validating the sensing performance of the integrated 3D printed torque sensors?
4. What does the control, needed to demonstrate such application, look like?

## 1.5 Report structure

The challenges that are faced during this project are systematically organized and documented. Each chapter is introduced and concluded. In this Chapter 1 the work is introduced and the objectives are laid out. Chapter 2 follows up on the objectives with literature on sensing principles, design & fabrication methods and control strategies. Within this chapter, the design choices for the sensor are made. Chapter 3 introduces the application that will be used to showcase the feasibility of the torque sensor. Chapter 4 provides a detailed overview of the design process, including a structural analysis of the intended behavior, and the fabrication. In Chapter 5 the characterization of the sensor is described. An overview of the measurement setup and resulting data is given. In chapter 6 the sensor and application are simulated and the realisation of the application is described, it provides an overview of the system in which the sensor is integrated and shows the resulting validation process. Chapter 7 discusses the work that is done in this project and provides suggestions towards future work. Lastly, in Chapter 8 the project is concluded and the research questions are reflected upon.

---

## 2 Literature

### 2.1 Introduction

In this chapter literature is reviewed on the topics of sensing, fabrication and control as to provide the necessary background required to support the choices in this work. First and foremost is the sensing principle that will be used in the sensor. Next, the fabrication method is described, together with the materials that might be used for the production of the sensor. Lastly, the control section of the work is elaborated upon.

### 2.2 Sensing Principle

In the field of 3D printed sensors there are several potential sensing principles. Schouten et al. [5] describes two of the most commonly used principles, that might be applicable in this work, namely *capacitive* and *piezoresistive* sensing. While both might be viable, a quantitative method is used to choose the most appropriate one for a set application. In order to do so, first, a pairwise comparison matrix [25] is formed, as to determine the weight of several aspects of their functionality. Second, a hierarchical weighing over the determined weights is done for each sensing principle. Resulting in a weighted value that determines the best suited option. This results in a quantified process of choice. Furthermore, reasoning and explanations regarding the fabrication process are described and the results are concluded at the end of this chapter.

For a pairwise comparison to be made, four aspects of each sensing principle are described and later weighed at the end of this section. The results of this process are viewed in the conclusion of this chapter.

The four aspects of each principle that will be discussed and weighted are; *performance*, *design*, *usage* and *mechanical dependability*. Each of these aspects is described such that it is clear what it entails. Several aspects are divided in sub-aspects and weighted individually.

**Performance** This performance aspect of the sensing principle focuses on the repeatability of its responses. It considers the linearity, drift and hysteresis of the method based on previous research.

**Design** Within this design aspect two sub-aspects are weighted. First, the *methodology* of the design is considered. Based on literature, the fabrication process, the implementation and the reproducibility of the models is evaluated. Secondly, the production *cost* is gauged, this includes material and fabrication costs, but excludes man hours and implementation.

**Usage** The aspect of usage is divided in three sub-aspects. First, it considers the *readout* properties and methods that come with the sensing principle. This includes equipment and interfacing of the data. Within the second sub-aspect the *durability* is assessed. This includes the expected design life cycle of the design and the repeatability of the required motion. Lastly, *implementation* is evaluated. The complexity of using the design in an application outside of testing, the equipment needed for usage and setup time.

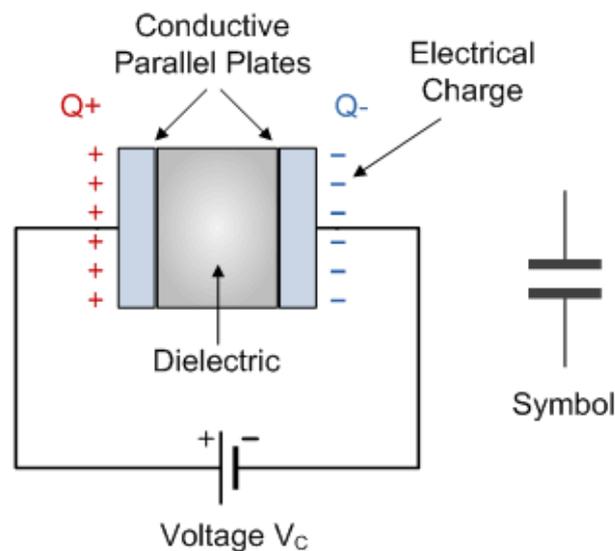
**Mechanical Dependability** Lastly, the mechanical dependability of the sensing principle considered. First, *deformation*, this includes the range of motion and the amount of moving parts. Also, *external factors* are considered. The influence of temperature, humidity, external interference and vibrations.

### 2.2.1 Capacitive sensing

In capacitive sensing, a capacitance is measured between two electrodes. The electrode geometry varies based on application. A common example is the parallel-plate capacitor seen in Figure 2.1. The capacitance is expressed, in approximation, by Equation 2.1. It states that capacitance is equal to the overlapping area  $A$  divided by the distance between the plates  $d$  times a permittivity constant of the dielectric  $\epsilon_c$ .

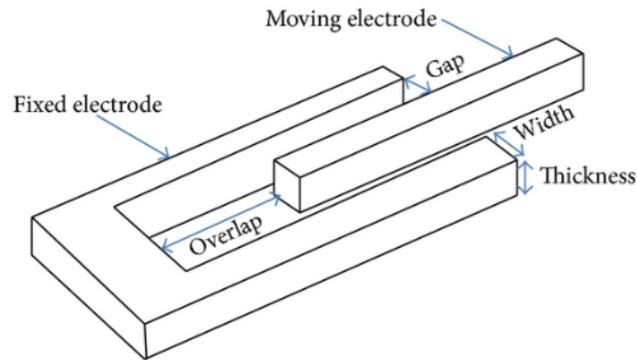
$$C = \frac{\epsilon_c A}{d} \quad (2.1)$$

The ability to 3D print conductive polymers allows for the development of parallel-plate capacitors with a soft dielectric layer [26, 27]. Applying a normal force [23, 26], or increasing a pressure [27] would compress the dielectric material thus reducing the distance  $d$  between the plates, resulting in a change in capacitance.



**Figure 2.1:** Schematic of parallel-plate capacitor with dielectric material [1]. On the right its schematic symbol.

Another example from Opvel [24] utilizes a change in capacitance due to a change in overlapping area of the plates as a result from shear forces (as opposed to changing distance between them). Fig 2.2 shows the principle of an overlapping area between two electrodes. This is supported by Equation 2.1 where a change in overlapping area  $A$  results in a change in capacitance. This technique is widely used in micro-electro-mechanical systems (MEMS) [28, 29].



**Figure 2.2:** Schematic of capacitor with with changing overlapping area [2].

Next, the four aspects with respect to capacitive sensing are described.

**Performance** Linearity in 3D printed capacitance sensors is shown to be high [30] despite the non linear behavior in conductive filaments for 3D printing. Non-linearity due to hysteresis and drift from the materials have limited influence on the measurements [5], this takes into account the assumption of a compressible, isotropic, homogeneous dielectric and deformations that are much smaller than the electrode dimensions [23].

**Design** The main characteristics of a design using capacitance sensing are the conductive structures that have to move relative to each other for the capacitance to change. This means one part of the conductive structure has to remain unaffected by the excitation while the other should be free to move to a certain degree (e.g. limited range of motion) [31]. While designs with both conductive structures moving is possible, it makes interpretation of the system harder. Furthermore, 3D printing increases the complexity of designs with parallel-plates that are horizontal to the printing bed as it will require a bridge between the conductive layers. The restriction being the limited motion of the dielectric. Another consideration of the design would be the requirement of shielding to minimize parasitic capacitances and crosstalk [32]. Also, it is vital for the conductive structures to not be able to touch each other as that will short the circuit and invalidate measurement results, while keeping the distance small as to keep the capacitance high (as seen from Equation 2.1).

As for the cost, larger area's of conductive filament like ProtoPasta [33] are more expensive in material cost [34]. But material cost is minimal compared to the cost of powering the equipment such as the printer, computers, oscilloscopes, work-hours, etc.

**Usage** Capacitive sensing readout requires a capacitance to digital (CDC) converter to output a change in voltage. With the equipment that is available at the time of the assignment, interfacing an application setup may significantly increase development time. Work done in the NIFTy group by [23], using FDC1004EVM CDC and [24] AD7747 CDC [35], respectively achieved sampling rates of 25 Hz (per channel) and 20 Hz during measurements.

The durability of designs using capacitive sensing increases as the range of motion is generally low (this is of course dependent on the structure). In this case the plastic deformation is less likely to occur (plastic deformation changes the characteristics of the sensor as the resting position changes).

**Mechanical dependency** First, deformation of a sensor using the capacitance principle differs by design. As mentioned before, 3D printing parallel plates horizontal to the bed requires a bridge between the layers. When this bridge is chosen to be a dielectric its deformation is re-

stricted to its elastic deformation. In the case of overlapping area, the possibilities are greater. Different design strategies need to be used to increase the deformation of the sensor.

Secondly, external factors such as temperature and humidity changes have an effect on capacitive sensors. A study from Analog Devices [36] shows that a change in capacitance is observed when either and both temperature and humidity vary. In both cases the change in capacitance was not well predictable [37]. There can also be electromagnetic interference (EMI) [38], caused by nearby electronics or conductors. Furthermore, as mentioned before, in order to reduce parasitic capacitances and crosstalk from external sources, proper shielding is necessary [32].

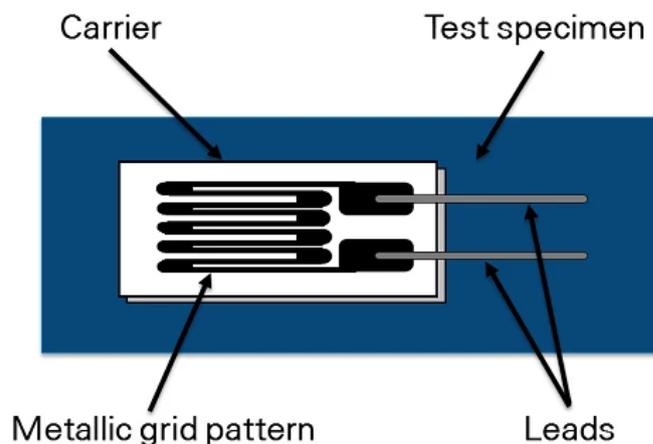
### 2.2.2 Piezoresistive sensing

Piezoresistive sensing utilizes the change in resistance in a conductive material as a result of mechanical *stress*. This stress results in a change in length, or mechanical *strain*. Another name for these type of sensors are *strain gauges*. Equation 2.2 & 2.3 show that the stress  $\sigma$  is the ratio of the applied force  $F$  (normal or shear) over the area  $A$ . The strain  $\epsilon$ , on the other hand, is the relative change in length  $L$  and is the ratio of the stress and the material's Young's modulus  $E$ .

$$\sigma = \frac{F}{A} \quad (2.2)$$

$$\epsilon = \frac{\Delta L}{L_0} = \frac{\sigma}{E} \quad (2.3)$$

A measured resistance in a strain gauge increases with tension and decreases with compression. There is a wide variety of applications such as load-cells, tactile sensors [19], flow sensors in area's of aviation (e.g. wing deflection, wind speed [39]), construction (e.g. structural integrity, critical load bearing) and more [19]. A schematic of a strain gauge is viewed in Fig 2.3. This typical example of a strain gauge shows the main components. The leads through which a resistance is measured, the carrier on which the strain gauge is fixed (mechanical strain is induced on the carrier). It also shows the common meandering pattern. While this is not necessary, it increases the surface over which the applied strain is captured (e.g. the same stress is applied over multiple parts of the length of the gauge, thus increasing the change in resistance).



**Figure 2.3:** Typical strain gauge where electrical resistance changes in proportion to the applied strain on the specimen [3].

These gauges are typically glued to the structure at places where strain is to be evaluated. 3D printing, using conductive filaments, allows for the integration of strain gauges in structures at

specific places of interest. Examples of current research on 3D printed sensors using piezoresistive sensing are [40] on a 3D printed tactile sensor, [41] on control of a vibrating beam using stiffness modulation, [42] on multi-axis force sensors using carbon nanotubes and [43] on highly elastic 3D printed strain sensors.

**Performance** The main advantages of piezoresistive sensing are the simple rudimentary readout and a potentially large sensing range. The main disadvantages are nonlinearities and hysteresis [40] in the piezoresistive behaviour of the conductive material and the relatively complex and thick designs needed to measure shear stresses using this sensing technique [24]. In order to combat some of these nonlinearities a differential measurement can be made. A differential measurement captures the response of two mirrored strain gauges. In an ideal symmetric differential the two responses are opposing each other and when subtracted remove nonlinearities in the response [44].

**Design** The methodology of a sensor using the strain gauges creates a lot of freedom as they are not restricted to a certain geometry (there are preferred ones such as the meandering pattern). Because of this, geometries used in mechanical engineering for torque measurements can be explored [45] [7]. Note that the aforementioned differential measurement requires two symmetrical and opposing gauges.

As for the cost, larger areas of conductive filament like ProtoPasta [33] are more expensive in material cost [34]. But, as with the capacitive sensors, material cost is minimal compared to the cost of powering the equipment, work-hours, etc.

**Usage** Strain gauges have a simple readout method requiring just an analog to digital converter (ADC) with the strain gauge in a bridge configuration [46] to measure the change in resistance.

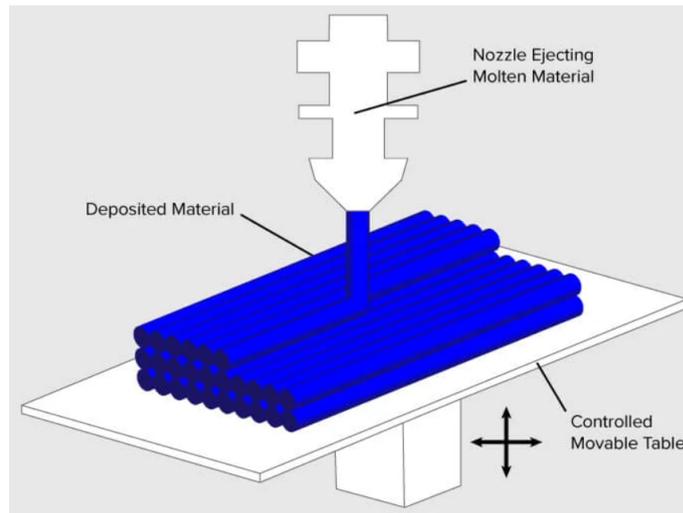
The durability is inherent to the motion the design is meant to make. Smaller motions lower risk of tearing and breaking. As for the capacitive sensing, staying in the elastic deformation range improves the durability.

As mentioned before, this sensing principle makes for freedom in design, this increases the possibilities for implementation of the sensor. Especially with the simple readout method, compact implementation is possible.

**Mechanical dependency** The deformation of strain gauge based sensors changes by the design and application. Traditional strain gauges are typically used in stiff environments (on metals or concrete). 3D printing creates the possibility for more flexible strain gauges with much larger deformation range [39] [21].

### 2.3 Fabrication

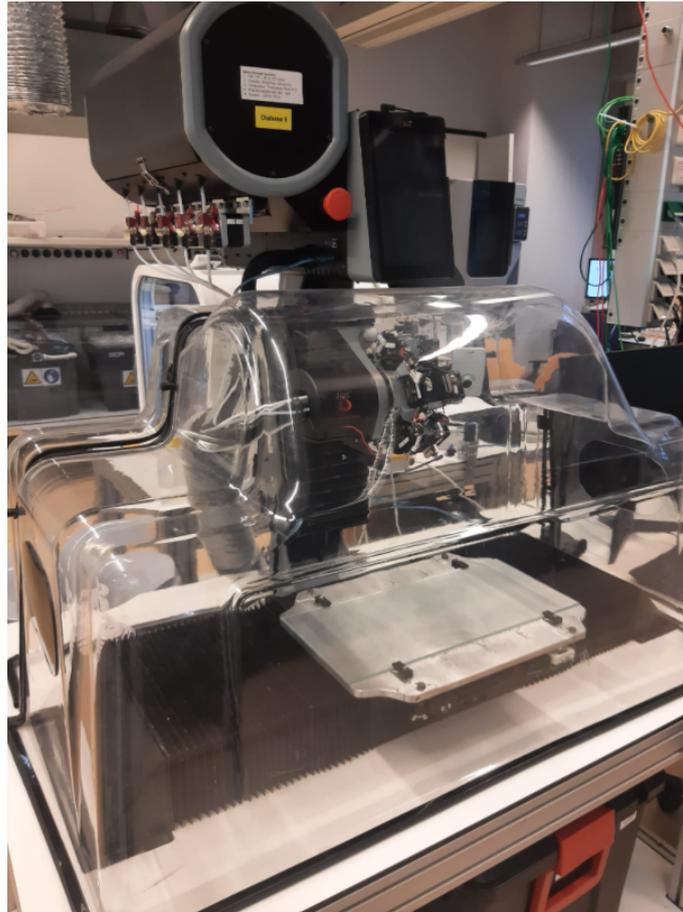
The fabrication process used for this work is FFF, an AM technology that deposits molten thermoplastic materials through a nozzle on a printing surface (or bed) for it to then solidify and harden. This creates a layer of material. By printing different layers on top of each other, 3D objects can be made. Figure 2.4 shows this process.



**Figure 2.4:** Schematic overview of FFF [4]. A nozzle deposits molten material on a surface creating a layer. In this image the nozzle is stationary and the bed position is controllable. This can also be the other way around.

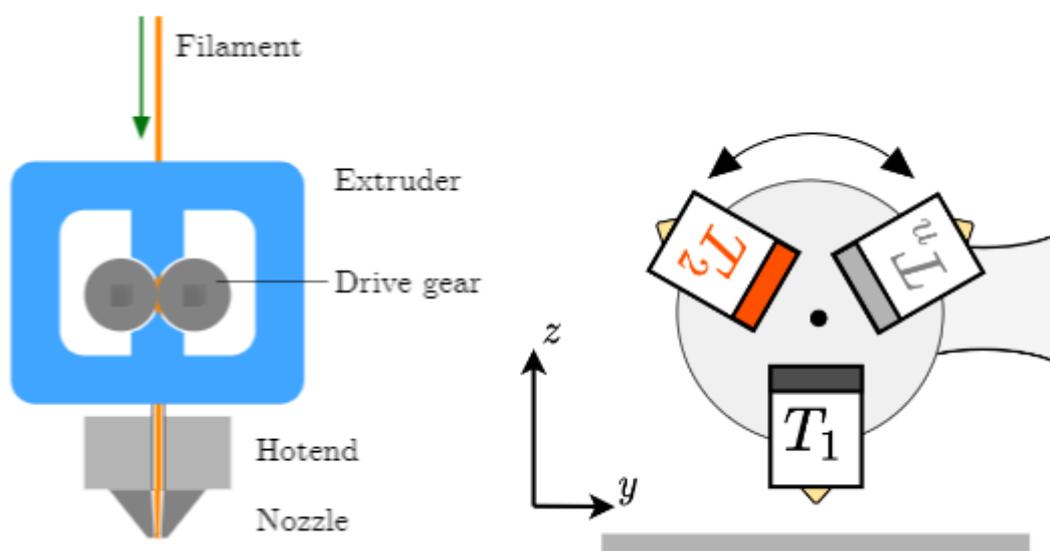
Jonkers et al [34] explain that this process is able to print both conductive and insulating materials that are widely available, all while maintaining a low-cost compared to other processes such as SLA or selective laser sintering (SLS). Comparing these (SLA & SLS) to FFF makes the latter seem inferior based on print quality, but this is not the goal. 3D printing offers high enough accuracy and precision for the application of this research and allows for fast and user friendly prototyping.

**Diabase printer** For production of the sensors, the Diabase H-series 3D Printer [47] is used. This printer is capable of printing with up to 5 different filaments.



**Figure 2.5:** The Diabase Engineering H-Series Multi-Material 3D Printer.

The Diabase uses a Direct-drive (Figure 2.6 left) extrusion system in which the extruder is placed on top of the print head (hot end & nozzle). The printer uses a rotary multi-toolhead as its filament switching mechanism (Figure 2.6 right).



**Figure 2.6:** Left: Direct-drive extrusion method [5]. Right: Rotary multi-toolhead system for filament switching [5]

**Table 2.1:** Properties of ABS and PLA [12]

Properties	ABS	PLA
Tensile Strength	27 MPA	37 MPA
Elongation	3.5%	6%
Flexural Modulus	2.1 GPa	4 GPa
Density	$1.0 \frac{g}{cm^3}$	$1.3 \frac{g}{cm^3}$
Melting Point	200°	173°
Biodegradeable	No	Yes
Glass Transition Temperature	105°	60°

**Structural Material** The structural material is a choice between polylactic acid (PLA) and acrylonitrile butadiene styrene (ABS). Both are thermoplastics suitable for 3D printing. Table 2.1 shows the material properties for both.

PLA has useful properties such as high flexural and tensile strength, making PLA a stiff material. It also has a low melting temperature thus making it easier to print [48]. A downside to PLA is the low glass temperature, it loses most strength at temperatures over 50 degrees [49]. PLA is a widely used structural material in 3D printing and comes at low cost.

ABS is a less stiff with lower flexural and tensile strengths. It is however, more impact resistant while being lighter than PLA. It thus is more durable, but at a cost of being harder to print with since its melting temperature is higher, requiring higher energy consumption [50] and is prone to warping.

**Conductive Material** Conductive filament are type of 3D printing material that are electrically conductive and can be used to create functional, electrically connected components within a 3D printed object. This material is often made of carbon-filled plastics

There are several commercially available conductive filaments for FFF [34] [5]. Two materials were considered. A PLA based conductive polymer Protopasta Electrically Conductive Composite PLA [33] and NinjaTek Eel Flexible Conductive Filament TPU [51].

## 2.4 Control

In this section it is formulated how the dynamics of the system will be described. The result of this description will be used to design a controller for the system such that it can be controlled with an input torque.

### 2.4.1 Lagrangian Mechanics

The first step will be to describe the dynamics of the system. The equations of motion for a robot joint are:

$$M(q)\ddot{q} + C(q, \dot{q}) + g(q) + h(q, \dot{q}) + \tau_{\text{ext}} = \tau \quad (2.4)$$

This Equation 2.4 is a description of the torque in each joint of a robot and will be elaborated upon later after its derivation. It is derived using the Lagrangian  $L$  as

$$L(q, \dot{q}) = K(q, \dot{q}) - P(q) \quad (2.5)$$

with  $K$  being the kinetic energy and  $P$  the potential energy in a joint. With the kinetic energy of the  $i^{\text{th}}$  link in case of a linear velocity  $v$  being

$$K_i = \frac{1}{2} m_i v_i^2 \quad K_i = \frac{1}{2} \frac{p_i^2}{m_i} \quad (2.6)$$

with mass  $m_i$ . Or in case of a rotational velocity  $\dot{\theta}$  in link  $i$

$$K_i = \frac{1}{2} I_i \dot{\theta}_i^2 \quad L_i = I_i \cdot \omega_i = L_i \cdot \dot{\theta} \quad K_i = \frac{1}{2} \frac{L_i^2}{I_i} \quad (2.7)$$

with the inertia of the body being  $I_i$ . Potential energy  $P$  in a link  $i$  is the energy held by that joint from its position and is described as

$$P_i = -m_i g h. \quad (2.8)$$

With this description of the Lagrangian the torque of a joint can be described by applying the Euler-Lagrange equation

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} + \frac{\partial L}{\partial q_i} = \tau_i. \quad (2.9)$$

Combining Equation 2.9 and 2.5 the initial description of the equations of motion for a robot can be derived (2.4). This derivation is viewed in [52].

Now that the derivation is explained, Equation 2.4 is elaborated upon.

$M(q) \in R^{n \times n}$  is the so called inertia matrix with  $n$ -degrees of freedom (DOF) and  $q \in R^n$  the generalized coordinates of the system. This matrix contains the parts that are linear to the acceleration. The matrix  $M(q)$  is positive definite and symmetric with dimensions  $n \times n$ .

$C(q, \dot{q})$  are the Coriolis forces acting on the masses, it contains the parts that are dependent on the velocity.

$g(q)$  the gravitational contributions that are independent of the acceleration and velocity.

$h(q, \dot{q})$  are torques from frictional forces and stiffness. The term  $\tau_{\text{ext}}$  is the measured external torques introduced to the system by the environment (in this case measured by the sensor).

In order to compute the matrices of 2.4 listed above, we first define some important variables. First, the *pose*, it describes the position and orientation of a 1 DOF joint as

$$P_i = \begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix} \quad (2.10)$$

for every joint  $i$  in a system [53]. From the pose we can compute the velocity vector. This vector is called the *twist* of a pose.

$$T = \dot{P}_i = \frac{\partial P_i}{\partial q} \frac{dq}{dt} = J_i(q) \dot{q}. \quad (2.11)$$

With  $J_i$  being the Jacobian of the pose defined as

$$J_i(q) = \frac{\partial P_i}{\partial q}. \quad (2.12)$$

**Inertia Matrix** It is shown that the inertia matrix  $M(q)$  is related to the kinetic energy function by [54] as

$$K = \frac{1}{2} \dot{q}^T M(q) \dot{q}. \quad (2.13)$$

To get to this inertia matrix we express the kinetic energy (Equation 2.6 and 2.7) in a matrix form for a pose results in the expression

$$K = \frac{1}{2} \dot{P}_i^T O_i \dot{P}_i, \quad (2.14)$$

where  $O$  represents the matrix  $O_i = \begin{bmatrix} m_i & 0 & 0 \\ 0 & m_i & 0 \\ 0 & 0 & I_i \end{bmatrix}$ , containing the masses corresponding with the pose  $P_i$ . Substituting Equation 2.11 in Equation 2.14 yields

$$K = \frac{1}{2} T_i^T O_i T_i = \frac{1}{2} \dot{q}^T J_i(q)^T O_i J_i(q) \dot{q}. \quad (2.15)$$

Using the generalized velocity vector  $\dot{q}$ . Now Equation 2.13 energy to extract an expression for the inertia matrix  $M(q)$  as

$$M(q) = J_i(q)^T O_i J_i(q). \quad (2.16)$$

**Coriolis Matrix** The Coriolis matrix [53]  $C(q, \dot{q})$  has dimensions  $n \times n$  and is a function of both the position  $q$  and velocity  $\dot{q}$ . It is a combination of Coriolis and centrifugal forces. One thing to note is that when the velocity  $\dot{q}$  is zero, the matrix result of  $C(q, \dot{q})$  becomes zero. Because the Coriolis matrix depends on velocity.

To compute this Coriolis matrix, the Christoffel symbols of the first kind  $\Gamma$  are used. They can be related to the inertia matrix [55] by

$$\Gamma_{ijk}(q) = \frac{1}{2} \left[ \frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{jk}}{\partial q_i} \right] \quad (2.17)$$

where the indices  $i, j, k$  represent the so called connection coefficients.

The Coriolis matrix can then be computed [55] by

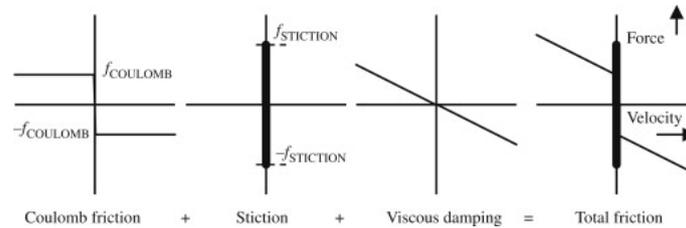
$$C(q, \dot{q}) = \sum_j \Gamma_{ijk}(q) \dot{q}_k, \quad (2.18)$$

**Gravitational Forces** The gravitational torque vector  $g(q)$ , of dimension  $n \times 1$  is a function of the current position. It is denoted as

$$g(q) = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix}. \quad (2.19)$$

Where for each link in the system the gravitational component is calculated depending on the state  $q$ .

**Residual Torques** The last term  $h(q, \dot{q})$  contains the residual torques, or torques imposed on the system from other sources. Within this term we will account for any type of friction. There are several types of friction, [6] explains a commonly used friction model based on three frictional forces (Figure 2.7). The three being, Coulomb friction, static friction (stiction) and viscous friction.



**Figure 2.7:** Commonly used friction model from [6].

In this work the friction is generalized in the viscous damping. Viscous damping is velocity dependent on the velocity as seen in Figure 2.7.

## 2.5 Conclusion

### 2.5.1 Sensing Principle

Both piezoresistive and capacitive sensing are discussed based on the four aspects of their workings (Performance, Design, Usage, Mechanical Dependency). These aspects were pairwise weight and ranked. Appendix B contains both the pairwise comparison matrix and the final weight matrix. The results of the ranking matrix is used in Table 2.3. To determine the best suited option, a weighing is done based on a 5-point scale [24]:

**Table 2.2:** 5-point scaling.

Symbol	Performance	Points
++	excellent	2
+	good	1
o	satisfactory	-
-	weak	-1
--	inadequate	-2

This weighing is multiplied with final weights and summed to find a utility value for both sensing principles. The highest utility value corresponds with the best suited option [56]. The weighing is done based on the discussed literature research.

**Table 2.3:** Hierarchical weighing of aspects of sensing principles.

	Weight	Piezoresistive	Capacitive
<b>Performance</b>			
Linearity	23%	-	+
<b>Design</b>			
Complexity	14%	+	o
Cost	3.5%	+	+
<b>Usage</b>			
Readout	21.5%	+	-
Durability	1.8%	o	+
Implementation	19,5%	+	-
<b>Mechanical Dependency</b>			
Deformation	9%	++	o
External factors	7%	--	++
<b>Value</b>		0.4	0.02

From Table 2.3 we conclude that the piezoresistive sensing principle is the best suited option for this research.

### 2.5.2 Fabrication

Concluding on the materials used in this research the chosen options are PLA for the structural material and ProtoPasta for the conductive material.

The PLA is chosen based on the lower melting point that is in line with the melting point of Protopasta. This is mainly to avoid any problems arising from contact between the different materials when with a high temperature difference (e.g. molten ABS might cause melting of PLA or ProtoPasta, this might compromise the bonding between both).

The choice of ProtoPasta is done based on the fact that it is PLA based and thus of similar composition to the structural material. Also, it is stiffer than the NinjaTek and since the PLA is of similar stiffness makes it logical. Furthermore, research shows a more desirable linear behavior in ProtoPasta for small deformations.

### 2.5.3 Control

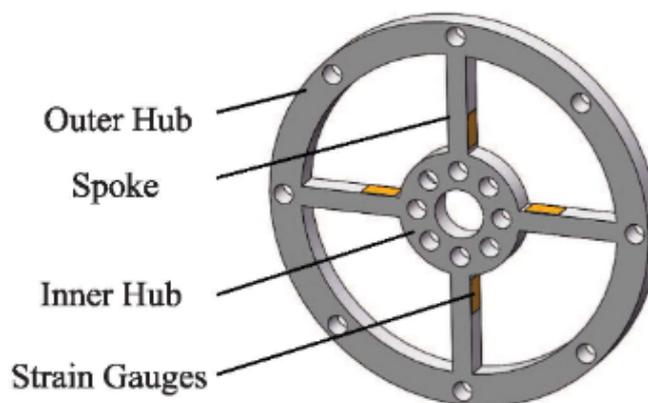
The dynamics of the system will be described using the Lagrangian and Euler-Lagrange formulation. This will result in a notation to calculate torque for any joint in a system and can be used to design a controller based on the systems dynamics. The sensor torque will be used to control the behavior of the system.

## 3 Design

### 3.1 Introduction

This chapter introduces the application that is meant to showcase the feasibility of the idea of a 3D printed torque sensor. This is done by integrating the sensor in a single DOF pendulum system after which it will be controlled by applying a position control loop on the system. From this application a list of requirements is made, requirements for both the pendulum and the setup. An interaction control layer with help of the torque feedback from the sensor can then be applied. In the following sections both the application, and the control methods are discussed.

Next, the sensor design process is presented. The sensor design is inspired by the work of Lou et al. [7]. They describe a one-dimensional torque sensor for robot collision detection. It consists of an outer ring and an inner ring about which a torque is applied. The two rings are connected by four spokes (beams) that are equally spaced along the geometry, as presented in Figure 3.1.

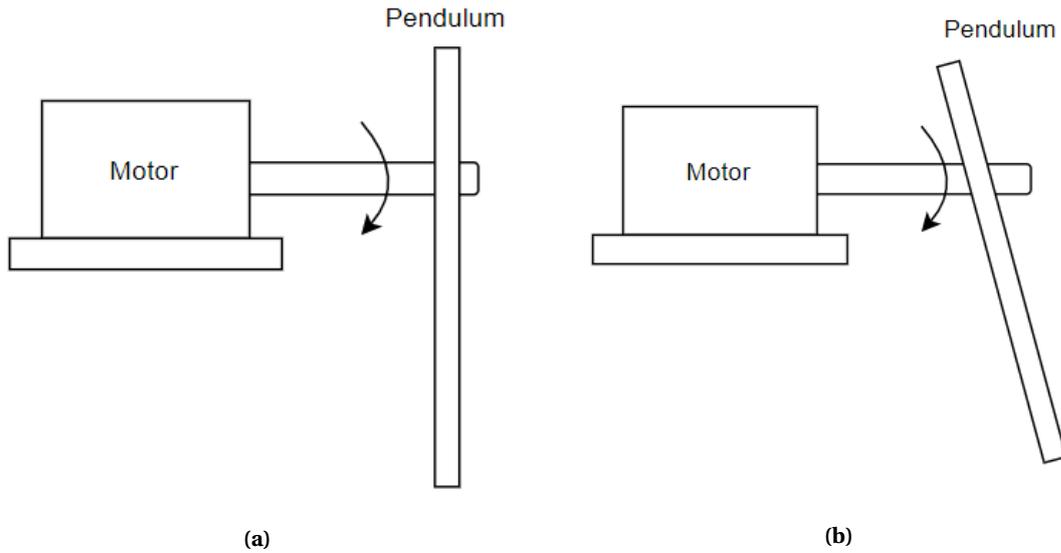


**Figure 3.1:** Early sensor design from Lou [7]. With outer and inner ring diameter respectively being 78 mm and 10 mm. Strain gauges are placed on the beams near the inner ring (both sides).

Applying a torque to the sensor will cause a rotation along its central axis (perpendicular to the sensor). This torque results in a strain of the beams, causing them to deform. To get insight in the strain and stress, a structural analysis is done. In this chapter beam theory is used to analyse the beams and optimize the placement and effectiveness of the strain gauges.

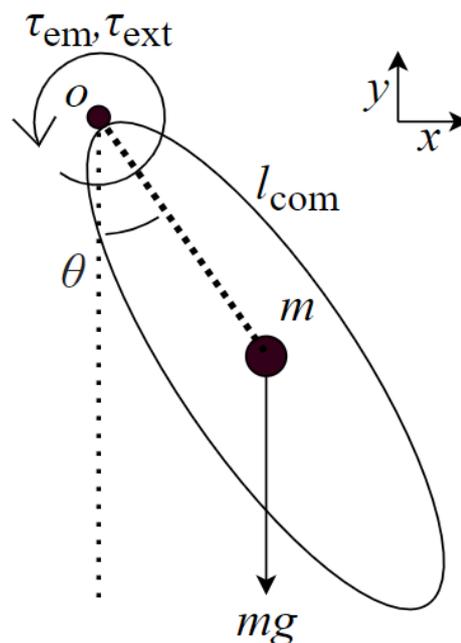
### 3.2 Application

The application for the sensor is chosen to be a 1 DOF single pendulum system with an integrated torque sensor, driven by a stepper motor. The system is chosen to be 1 DOF with one sensor in order to not over complicate the system. Introducing multiple sensors to the system would accumulate errors making validation of the system harder. The pendulum is fixed on the rotating shaft of the stepper motor such that the torque imposed by the motor is isolated and parasitic forces are minimized (for example a bending force due to the sensor not being perpendicular to the shaft as per Figure 3.2).



**Figure 3.2:** (a) Well aligned sensor to the motor shaft. (b) Misaligned sensor to motor shaft, induces a bending motion on the sensor which will influence the sensor response.

Figure 3.3 shows a schematic overview of the rigid body pendulum system with its main variables. Note that the angle  $\theta$  rotating around point  $O$ , when in rest (vertical), is zero. Furthermore, two torques  $\tau_{em}$  and  $\tau_{ext}$  are denoted.  $\tau_{em}$  is the electromagnetic torque induced by the motor and  $\tau_{ext}$  are torques applied to the system from external sources.  $l_{com}$  is the length to the centre of mass of the body with respect to the origin.  $m$  represents the mass of the body and  $g$  the gravitational constant.



**Figure 3.3:** Schematic overview of a rigid body pendulum driven by a stepper motor around  $o$ .

### 3.2.1 Requirements

From this description a list of requirements for both the pendulum and the setup can be constructed. The setup requires a rotating actuation of the pendulum, a microcontroller and in

order to read out the sensor several cables a half-bridge configurations is to be constructed (for a differential measurement to be made). The required components thus consist of the following:

- Pendulum
- Stepper motor
- Microcontroller
  - Motor controller
  - ADC for sensor readout
  - Power supply
- Half-bridge configuration setup

Next, a table with design requirements for the sensor/pendulum is made (Table 3.1). This entails the shape, dimensions, its weight, length and the force range. Quantifying these parameters helps the design process and define a scope for the design.

**Table 3.1:** Table of design requirements for sensor design.

Aspect	Value	Unit
Diameter	< 100	mm
Thickness	< 7	mm
Weight	60-100	g
No. of Gauges	4	N/A
Peak Torque	$\pm 4$	Nm
Pendulum Length	250-400	mm

### 3.2.2 Control

The system as described in this chapter will be position controlled based on a calculated torque input. At this point a part of the sensor validation can be done by comparing the calculated input torque to the measured torque (it will feed back a voltage difference that is converted to a torque, more on that in Chapter 4). This is because the measured torque is a combination of the electromagnetic torque induced by the motor and any external torques.

The next step is to introduce an interaction control layer to the system. An external torque threshold is set, with the external torque being the difference between the motor torque and measured torque. This threshold can represent the result of an interaction with the environment, be it a bump in with a person, hitting a wall or an object. When this threshold is exceeded the system immediately halts the position control as long as the threshold is not exceeded (See Section 5.2.1).

## 3.3 Structural Analysis

### 3.3.1 Beam Theory

For insight in the dynamic behavior of the sensor, beam theory is applied to the structure. This way, strain and stress can be evaluated. The stress in a cross section of a solid beam is denoted by

$$\sigma = \frac{My}{I} \quad (3.1)$$

where  $M$  is the applied moment at the point,  $y$  the distance perpendicular to the neutral axis at which to calculate the stress  $\sigma$ , and  $I$  the cross-section area moment of inertia, which changes with respect to the cross section geometry. The largest stress of a cross section is found at  $y_{\max}$  at the surface. At that point the equation [11] can be rewritten to

$$\sigma_{\max} = \frac{M}{Z} \quad (3.2)$$

With  $Z = \frac{I}{y_{\max}}$  being the section modulus of the geometry at the largest distance from the neutral axis. Figure A.1 in Appendix A shows equations for  $Z$  for various geometries.

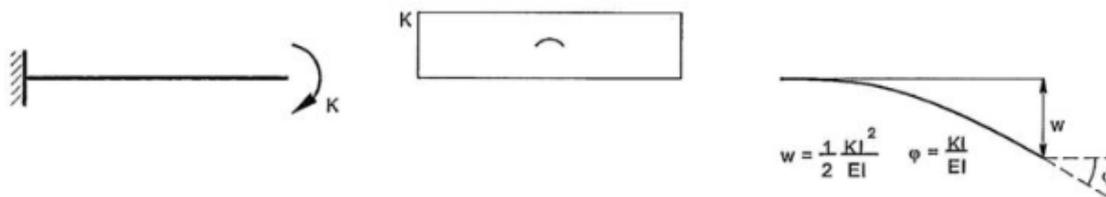
As we are interested purely in stress and strain at the surface of the structure, the equation for strain [11]  $\epsilon$  can be simplified to

$$\epsilon_{\max} = \frac{\sigma_{\max}}{E} = \frac{M}{ZE} \quad (3.3)$$

With  $E$  being the young's modulus of the material. For both Equation 3.2 and 3.3, the maximum stress ( $\sigma_{\max}$ ) and strain ( $\epsilon_{\max}$ ) can be found at  $M_{\max}$ . Note that this Equation 3.3 assumes homogeneous isotropic materials which 3D printed parts are not which, for this analysis, is assumed.

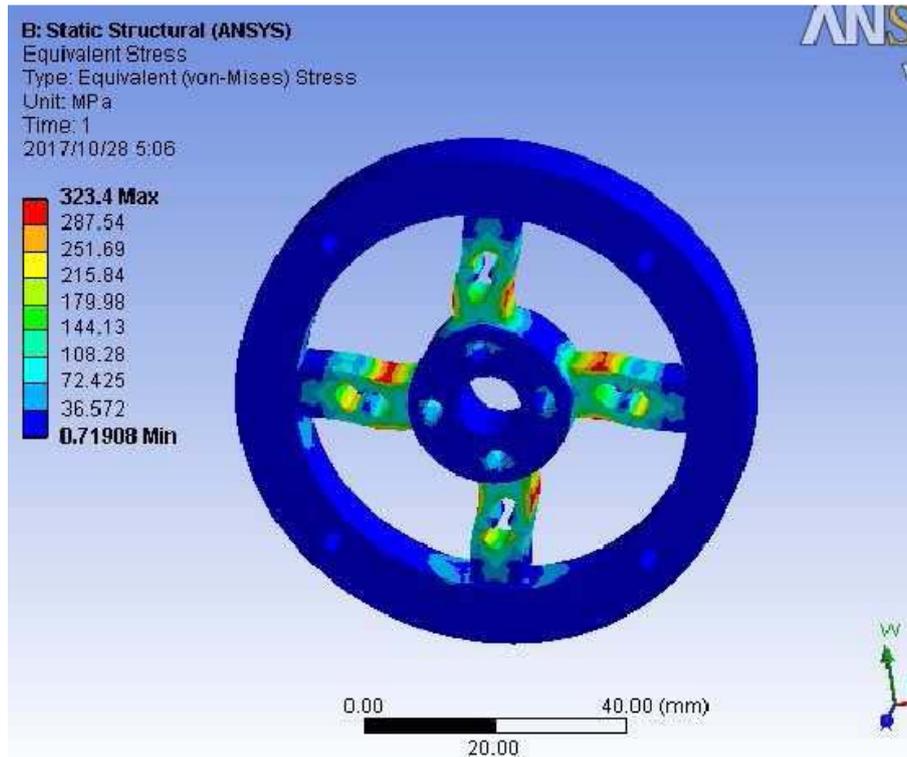
### 3.3.2 Analysis

To analyse the structure, a free body diagram (FBD) is made to an equivalent mechanical model. The case by Lou et al. [7] states that for a rotation for a body with a fixed outer ring, the equivalent mechanical model is cantilever with one end fixed, and a concentrated moment on the other end. This implies that the moment diagram shows a continues moment along the entire length of the beam and a continues bending as pictured in the FBD of this model in Figure 3.4. Lou argues that when a torque is applied at the outer ring it behaves as a flexure with an open end that is free to move.



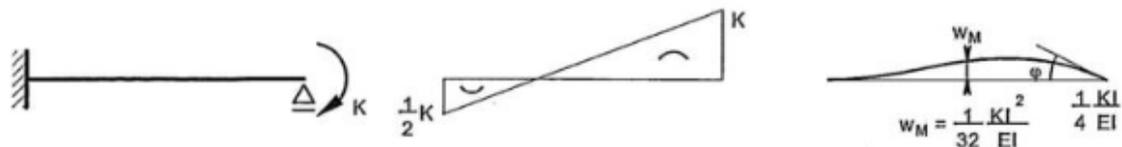
**Figure 3.4:** Left: FBD of a single free end cantilever with end moment [8]. Middle: Moment diagram of this equivalent model. Right: Bending diagram with largest deformation at the end point.

We argue that this model is to far a simplification. When a moment is applied to the inner body with more then one beams attached to it, the inner body will resist free movement and rotates around its axis and thus will not move as seen in Figure 3.4. This argument is in line with the expected s-shape bending in the beams as seen in the resulting finite element method figures from Lou [7] in Figure 3.5 where the largest stress is located near the rotating inner ring and minimal stress at the outer ring.



**Figure 3.5:** Finite element method (FEM) results from [7]. Largest stress is concentrated near the inner ring (red).

With this in mind the equivalent model should include a pinned end about which it can rotate. This is represented in Figure 3.6.

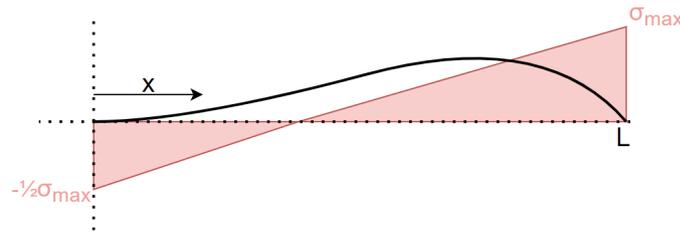


**Figure 3.6:** Left: FBD of a single pinned support cantilever [8]. Middle: Moment diagram of this equivalent model. Right: Bending diagram that represents an s-shape.

Analyzing this case shows a linearly declining moment over the length of the beam from  $K$  at the end of the beam to  $-\frac{1}{2}K$  at the fixed side. This results in the more expected s-shaped bending.

### 3.3.3 Sensor Placement

For Equation 3.3 we conclude that strain gauges should be placed along the area of the largest moment (assuming  $Z$  and  $E$  constant) with distribution of strain as evenly as possible. This ensures a more linear voltage output as a result of a torque input [7]. In the case of Figure 3.6 the maximum strain occurs at the point of the largest moment. This being at the edge of the inner ring. However, due to limited space at this point, placement of the gauges is hard and the stress is not evenly distributed. A graph of the stress along the beam is viewed in Figure 3.7.

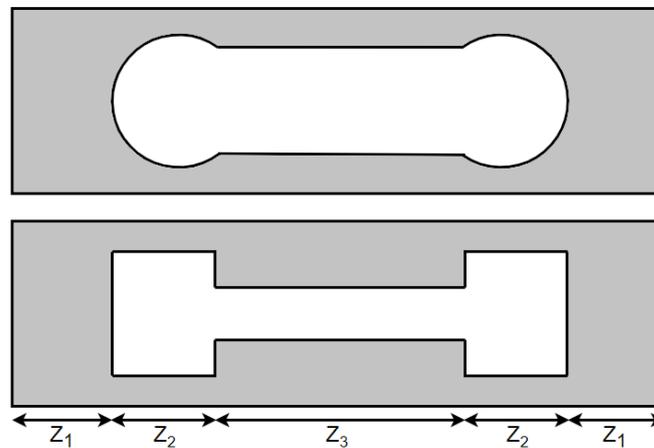


**Figure 3.7:** Stress graph depicted in red with  $\sigma_{\max}$  at length  $L$  and bending diagram from  $0 < x < L$  depicted in black (exaggerated). With positive stress at the right side where the moment is applied and negative stress at the fixed side.

In order to maximize effectiveness of a strain gauge, it should be placed in pure tension or compression region of the beam ( $0 < x < \frac{1}{3}L$  and  $\frac{1}{3}L < x < L$  being the two regions) as overlap with tension and compression results in a net zero gain of stress. With the larger stress at the side of the inner ring, placement of the gauge at that side is optimal. Thus the strain gauge should be placed in region  $\frac{1}{3}L < x < L$ .

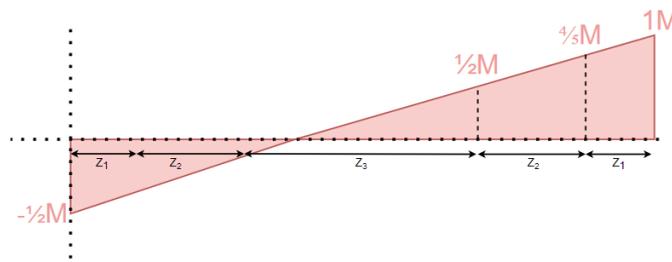
### 3.3.4 Stress Optimization

To optimize the effectiveness of the strain gauge further, the strain along its length should be increased and distributed evenly. As mentioned, a more evenly distributed stress at the position of the strain gauge is beneficial. This can be achieved by changing the geometry of the beam along its length  $L$ . Looking at the Equation 3.3, strain is depending on the moment  $M$  and section modulus  $Z$  and  $E$ . This implied that decreasing  $Z$  increases strain  $\epsilon$ . The case of Lou et al. [7] considers a slot in the beam as seen in Figure 3.8.



**Figure 3.8:** Beams with slots. gray area is the material, white area the removed geometry to decrease section modulus  $Z$ . Top: Model used by Lou et al. [7] to increase strain distribution. Bottom: Simplified model for analysis where  $Z$  has three constant values.

Using this model, the following  $Z, M$  diagram can will be considered (Figure 3.9). Note that the ratios of  $M$  (other than  $1M$  and  $-\frac{1}{2}M$ ) are not accurate but chosen to illustrate the decaying moment along the length. For analysis we are interested in the ratio  $\frac{M}{Z}$ , which is the highest at the end point being  $\frac{M}{Z_1}$ , from this point moment  $M$  decreases linearly to  $-\frac{1}{2}M$ . To optimize the strain along the length,  $Z_2$  should be sufficiently small to even the strain distribution.



**Figure 3.9:**  $Z, M$  graph of the simplified model with three different values of  $Z$ .

### 3.4 Sensor-Cable Interface

3D-printed strain gauges from ProtoPasta provide the conductive structure through which a resistance can be measured. In order for a measurement to be done, probes or wires have to be attached to the structure. There are several ways about this problem.

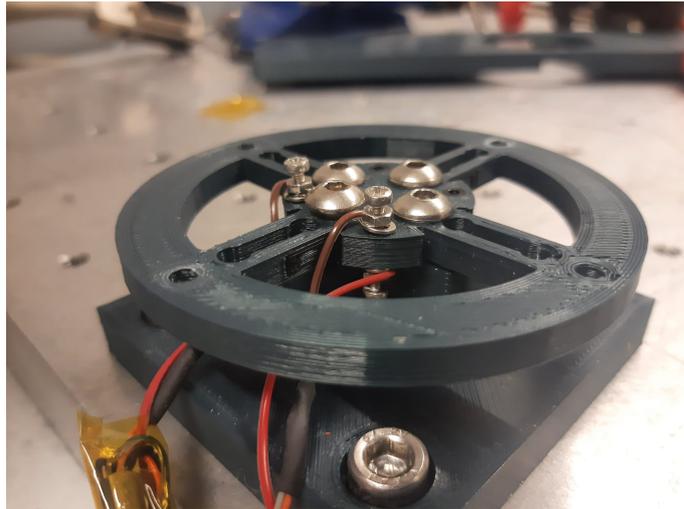
**Silver epoxy** Silver epoxy [57] (conductive) allows for wires to be connected to surfaces by applying it to the surface and the wire and letting the epoxy harden with the wire fixed in place, this process is also referred to as *cold soldering*. The epoxy is silver based for its high conductivity [34]. Due to the viscosity of the compound, applying it accurately is a challenge and requires an area larger than the wire that will be attached. Figure 3.10 shows an early iteration of the sensor with two wires soldered to one of the strain gauges.



**Figure 3.10:** Silver epoxy connection for the attachment of wires to the strain gauges of an early iteration of the sensor.

This process allows for placement of wires any place along the beam. One downside to this process is the time consumption, as the process is done by hand and requires a full day to cure at room temperature. The datasheet [57] recommends a heat treatment of 15 minutes at  $65^\circ$  to harden the epoxy, unfortunately the glass transition temperature of PLA is around  $60^\circ$  and is thus not able to be heat treated. Another downside is the durability, as the beam will deform it will stretch and compress the epoxy connection. This, over time, will lead to tearing and breaking.

**Screwed connections** Another way to form a connection between wires and the conductive surface is the use of a screw with a nut and washer to tighten wires in between. This way a modular connection to the wires can be made with help of a bolt securing the wires in place (Figure 3.11). The screw will tap itself into the structure in designated holes that connect to the strain gauges, creating a strong connection. These holes are placed in a thicker part of the structure that remains unaffected when torque is applied. Such that loosening over time by vibration and movement is minimized.

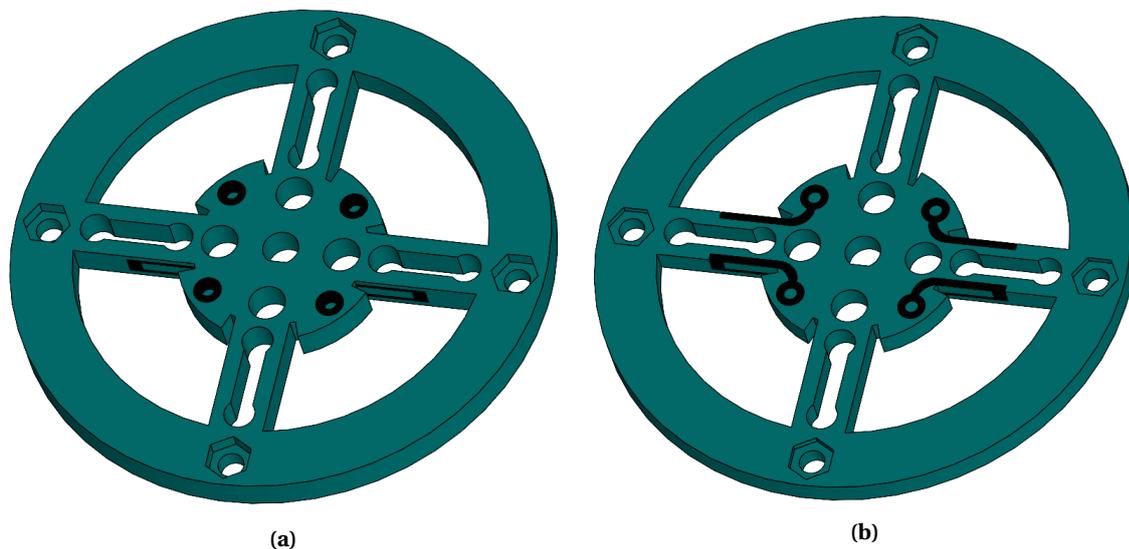


**Figure 3.11:** A screw and wire connection on the sensor.

The main advantage is the modularity of the solution. It allows for faster setup time and no need for designated cables. While this provides a non-permanent solution, it can be prone to problems such as loose wiring and loosening of screws over time. Connections should thus be checked before use.

### 3.5 CAD Model

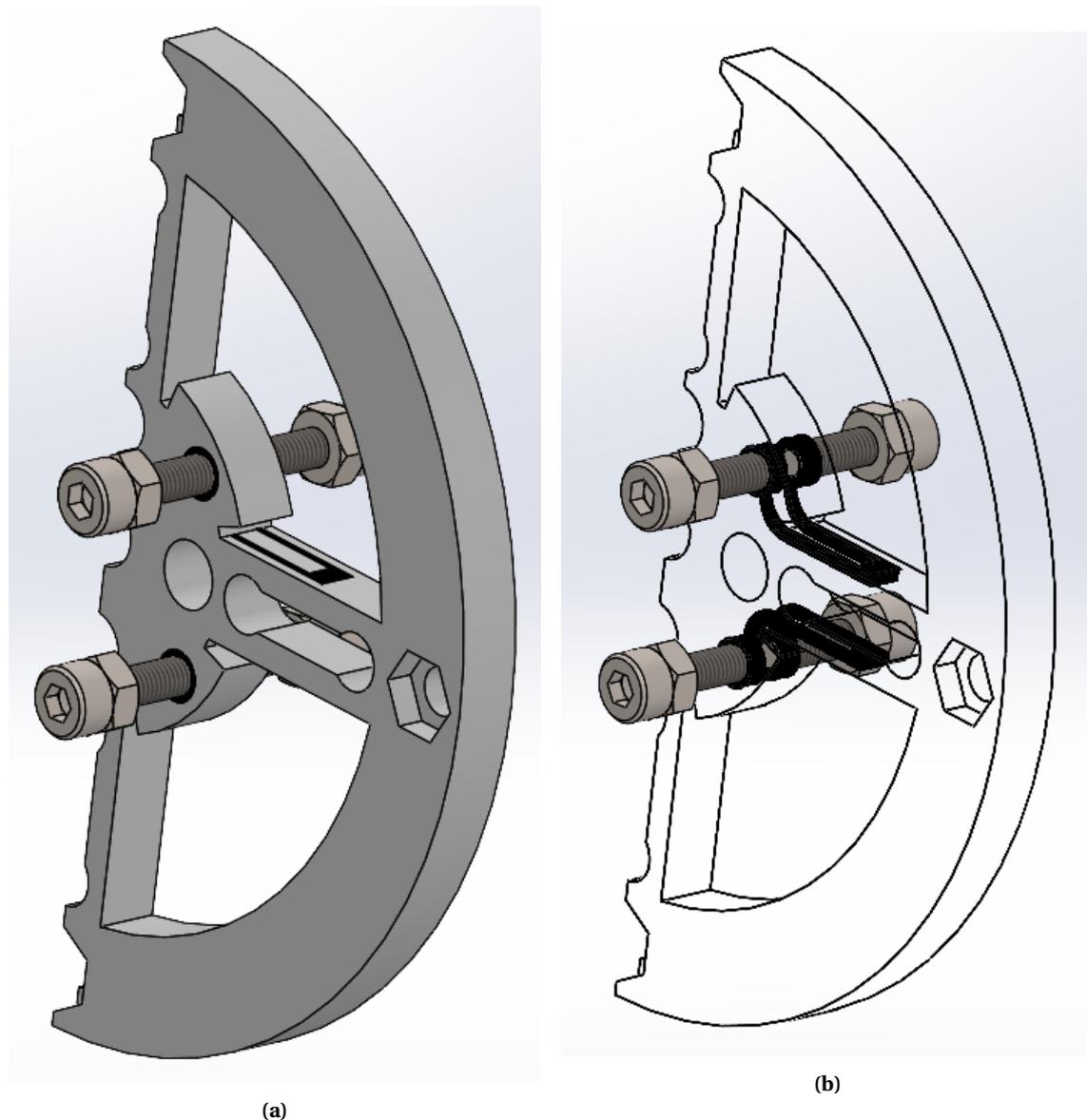
The computer aided design (CAD) model was created using SolidWorks [58], it has gone through multiple iterations (Table G) that were printed and evaluated. The resulted model is viewed in Figure 3.12 (a). The outer ring has a dimension of  $78\text{ mm}$ . For more detailed dimensions a technical drawing is viewed in Appendix C.



**Figure 3.12:** (a) Final design of the sensor with two sets of strain gauges on opposing sides of a spoke (left & right spokes). (b) Inner connection of the screw holes to the gauges on the spokes. Green material is PLA and black the conductive ProtoPasta.

As seen from Figure 3.12, two sets of strain gauges are present on each side of the model, a set being two gauges on opposite side of a spoke. Such that a differential measurement can be made. The screws are placed in the designated holes (black) on both sides of the sensor (only

front is shown) which are printed from the same conductive material as the gauges, a detailed figure of the strain gauges is viewed in Figure 3.13.



**Figure 3.13:** A detailed look at the strain gauges on one side of the sensor, with screws placed. (a) Shows the solid structure. (b) The structure is hidden such that the connection is shown. The gauges loop from one side of the sensor to the other.

As per Figure 3.13 we see the connection of the strain gauge to the screws, the curved geometry creates a nozzle trajectory without sharp corners. This is done in order to extend the life expectancy of the gauges, as sharp corners in 3D-printing are prone to breaking and tearing. Also, continuous printing lines have a more homogeneous electric conductivity [59]. In the case of the strain gauges, this would lead to invalid resistance measurements. Furthermore, the slots that were discussed in section 3.3.4 are present in all spokes. The inner circle contains five spokes. The inner circle contains five holes from which the outer four are used to secure the sensor to the test setup (more on that in Chapter 4), and the middle one for the output shaft of the actuator. In the outer ring, slots for nuts and bolts are present for fixing the external components (more on that in Chapter 4).

As mentioned in Chapter 3.2, the final design is a 3D-printed pendulum weight with an integrated sensor. The structure is extended to shift the centre of mass (COM) away from the motor

shaft hole, thus creating a pendulum structure. Figure 3.14 shows the final CAD. Notice the extra structure in the middle of the sensor, this is for fastening to the motor shaft using a grub screw.



**Figure 3.14:** Pendulum with integrated sensor design CAD.

**Iterations** Table 3.2 contains a rough overview of the main iterations of the sensor. The main changes are briefly mentioned and reasoned. Appendix G contains visuals of the design iterations.

**Table 3.2:** Table of design iterations and their main changes.

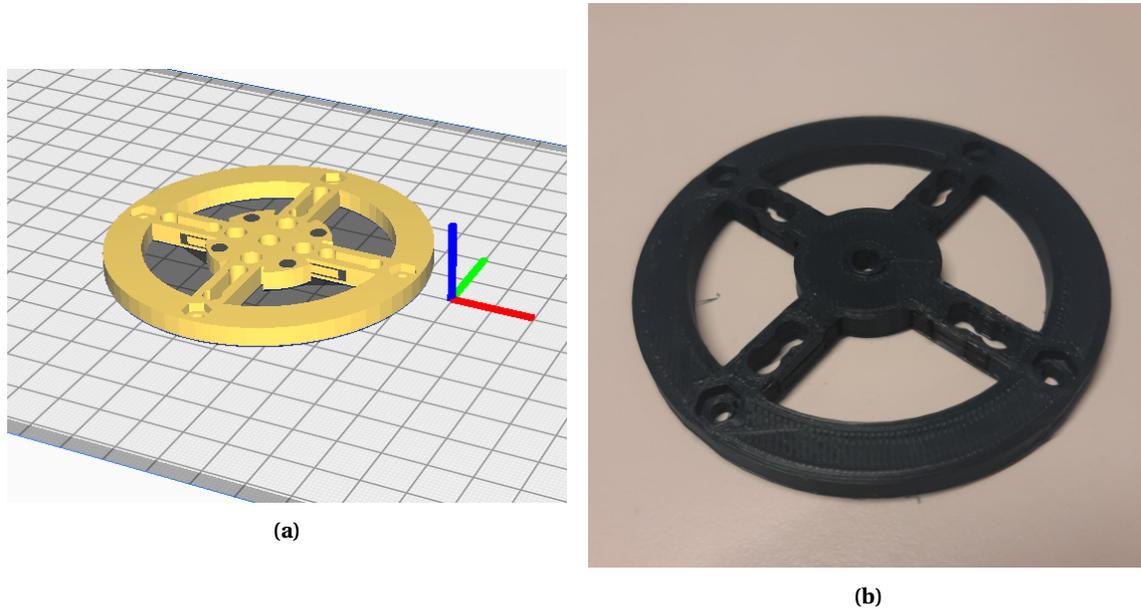
Version	Main Changes
0.1	Circular design with 8 strain gauges on surface of beams.
0.2	Wider holes in slots to increase strain & added holes for fixation.
0.3	First iteration with screw connections on both inner and outer ring on one side & gauges along the length of the beams.
0.4	New design with inverted slots (hinge design). Design was rejected as strain distribution would be unpredictable and gauge would require a curve.
0.5	First design with 4 gauges. 1 connection on beam (cold soldered) & 1 on inner ring (screwed).
0.6	Longer & wider slots for higher deformation and larger strain along the entire beam.
0.7	Added structure to make space for screw connections on inner ring.
0.8	First iteration with 2 screwed connections. 1 connection on inner ring & one halfway on beam. Rejected because of unstable structure prone to breaking over time.
0.9	First design with looping strain gauges with 2 screwed connection. 1 one either side of the sensor.
0.10	Added structure to make space for screw connections on inner ring.
0.11	Optimized strain gauge geometry.
0.12	Added pendulum structure and motor shaft structure.

## 3.6 Fabrication

### 3.6.1 Layering

Previous 3D-printed sensors [41] [40] [5] have their respective strain gauges printed horizontally and layered vertically. Meaning that each layer contains a full loop of the strain gauge from one connection to the other. This design has its strain gauges printed in the vertical orientation (each layer contains part of the loop), because this orientation results in a stronger print as the

stacked layers are (near) equal and no support is needed. To achieve this, the sensor will need to be printed flat on the printing bed. Figure 3.15 shows the sensor in the Cura Software [60] environment that is used to create instruction for the 3D printer [47].



**Figure 3.15:** (a) The CURA orientation showing layering direction. (b) Resulting print of the model.

The design with the integrated sensor was printed with the parameters shown in Table 3.3 and is shown in Figure 3.16.

**Table 3.3:** Printing parameters

Parameter	Value
<b>Layer settings</b>	
Layer thickness	200 $\mu\text{m}$
Wall line count	3
Top & Bottom layers	4
Infill	35%
<b>Hotend temperature</b>	
PLA	215 $^{\circ}\text{C}$
ProtoPasta	225 $^{\circ}\text{C}$
Heatbed temperature	60 $^{\circ}\text{C}$
<b>Printing speeds</b>	
Printing speed PLA	60 $\text{mms}^{-1}$
Printing speed PLA	15 $\text{mms}^{-1}$



**Figure 3.16:** Pendulum with integrated sensor design print.

### 3.7 Conclusion

#### 3.7.1 Application

The application will be a 1 DOF pendulum with the piezoresistive torque sensor integrated, to which position control loop is imposed. This loop provides a torque reference to be followed. Comparing this reference to the output of the sensor will be the first validation of the sensor. Based on the description of the application a list of hardware and design requirements is made. An interaction control layer will simulate interaction with the environment triggering a secondary control loop when a external torque threshold is met.

#### 3.7.2 Structural Analysis

From the analysis we conclude that the largest strain along the beams is near the point of rotation. The optimal placement of the sensors would thus be along that part of the sensor on the surface of the beam. An even distribution of the strain along the gauge is beneficial, so holes are added in the beams to lower the cross sectional inertia, consequently increase the strain as the surface of the beams along those holes.

#### 3.7.3 Connectors

From the methods for connection that are explored the screwed connection is the best suited option. Placing the holes for the screws near the hole for the motor shaft in a thicker structure prevents movement and minimizes vibration that can cause loosening over time. Furthermore, the modularity of this method makes the process of setting up the sensor much faster. Additionally, silver epoxy can be used to secure the screws in place and creating a more rigid connection.

### 3.7.4 CAD Model

The CAD model was created in SolidWorks with strain gauges looping from the front side of the sensor the back. The strain gauge structure is designed in such a way that it minimizes the risk of breaking and tearing due to movement of the beam by avoiding sharp corners. The gauges are mirrored along the beam such that differential measurements are possible. The design was altered by adding a structure, to shift the COM from the rotating axis to create a pendulum structure.

## 4 Characterisation

### 4.1 Introduction

In the chapter, the characterization process of the sensor is discussed. This is done by elaborating on the equipment that is used, the measurements setup in order to gather data and analysis of the acquired data both before and after processing.

### 4.2 Components

With the sensor designed and printed, the next step is to realize an experimental setup in order to characterize and assess the sensor behavior and performance. Within a lab setting, a well controlled and isolated environment is created, as to exert the desired motion and constrains. The main components consist of the following:

- 3D printed sensor
- 3D printed mount & arm extension
- DEWE-43-A Data Acquisition System (DAQ)
- SMAC linear actuator LCA25-050-15f
  - SMAC controller LC-10
- LCMFD-50N load-cell
- FUTEC IAA100 Load-cell Amplifier
- Power supply 1-30V
- A PC with required software
  - MATLAB
  - DEWE-43-A software

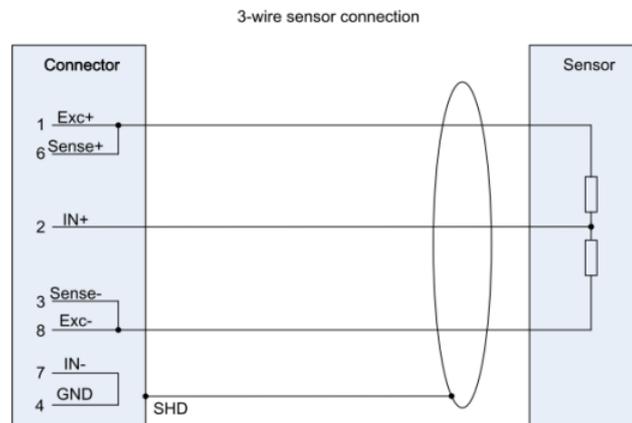
#### 4.2.1 DEWE-43-A

The DEWE-43-A4.1 is a DAQ with a wide variety of functionalities and an easy-to-use dedicated software [9]. For this setup its universal analog inputs are used to measure the voltage response of the strain gauges and the load-cell. Three of the eight available analog channels are used and are time synchronized within the DEWE, these channels have a maximum resolution of 24-bit and a maximum sampling rate of  $200 \text{ kS s}^{-1}$ . Furthermore, the DEWE's power supply is internally isolated which eliminates ground loops.



**Figure 4.1:** DEWE-43-A DAQ from [9].

In order to measure a voltage difference in the strain gauges during excitation, the gauges are put in a half-bridge configuration with a 10 k potentiometer (as the gauges measured to be around 7 k $\Omega$ , more on half-bridge configuration in Section 4.3). The bridge is excited with 10 V input. To achieve this configuration, several cables are modified according to Figure 4.2. This configuration is found in the provided hardware reference manual of the DEWE [61]. To balance the bridge, a multimeter was used to measure the resistance from the gauges and then matching the potentiometers to those resistances.



**Figure 4.2:** Half-bridge configuration of the connector cables with the sensor and a 10 K potentiometer [9].

By alternating provided DEWE example code a connection to the MATLAB environment is made (provided in Appendix D). This way a synchronized stream of data from the DEWE is logged to MATLAB for further processing. This results in four data sets per measurement, three measured data sets (two from the strain gauges and one from the load-cell) and a time vector.

#### 4.2.2 SMAC

The linear actuator used in this setup is the SMAC LCA25-050-15-6. This device is a solenoid based actuator with a 5 mm stroke and a peak force of 15 N operating on 24 V. Further specs are viewed in Appendix I. The dedicated software of the SMAC provided freedom of actuation

methods of either position or force control based on user preferences. The SMAC is controlled by the dedicated controller (LC-10) communicating via a USB-serial to the PC. The controller uses an embedded motion control library (EMCL) to manage all functions of the SMAC [62]. The SMAC can be controlled via the provided graphical user interface (GUI) in the software, but also a connection with this serial can be made in MATLAB such that the SMAC can be controlled through the MATLAB interface using commands provided by the manual [62].

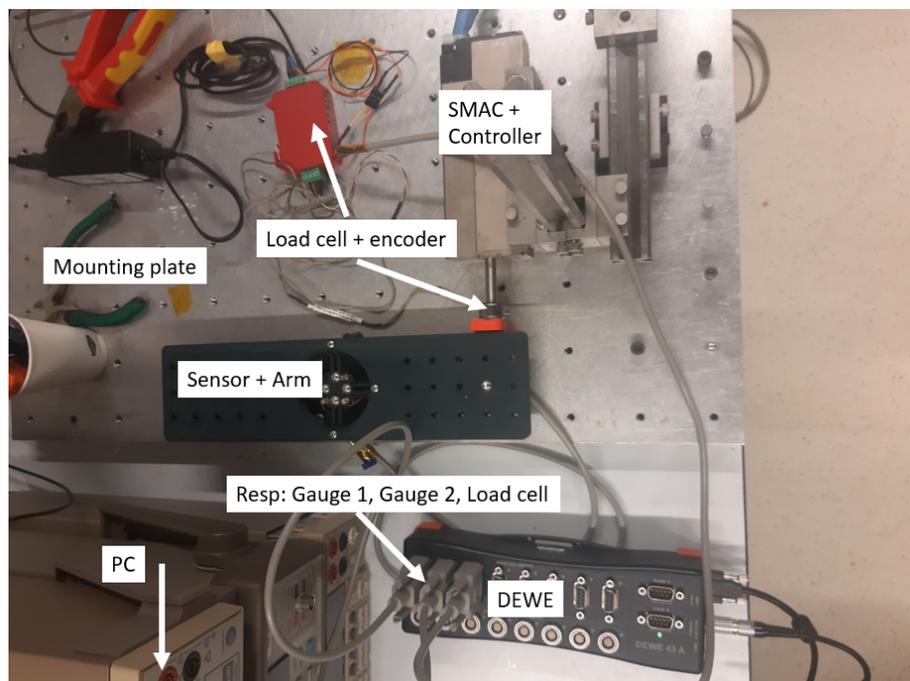
In this research the SMAC is initialized with the provided GUI. After initialization, MATLAB scripts are used to control the desired behavior.

#### 4.2.3 Software

As mentioned, the provided software of the LC-10 is used for initializing and further controlling of the SMAC via its provided GUI. The same goes for the DEWE, the provided firmware initializes the hardware by recognizing the channels that are being used (in this case the 3 analog channels) and opening a USB-serial connection. After initialization the hardware can be controlled and all data acquisition can be done through the MATLAB environment. Next to acquisition, post processing of the acquired data is also done in this environment. The code is to be found in Appendix E

#### 4.2.4 Setup

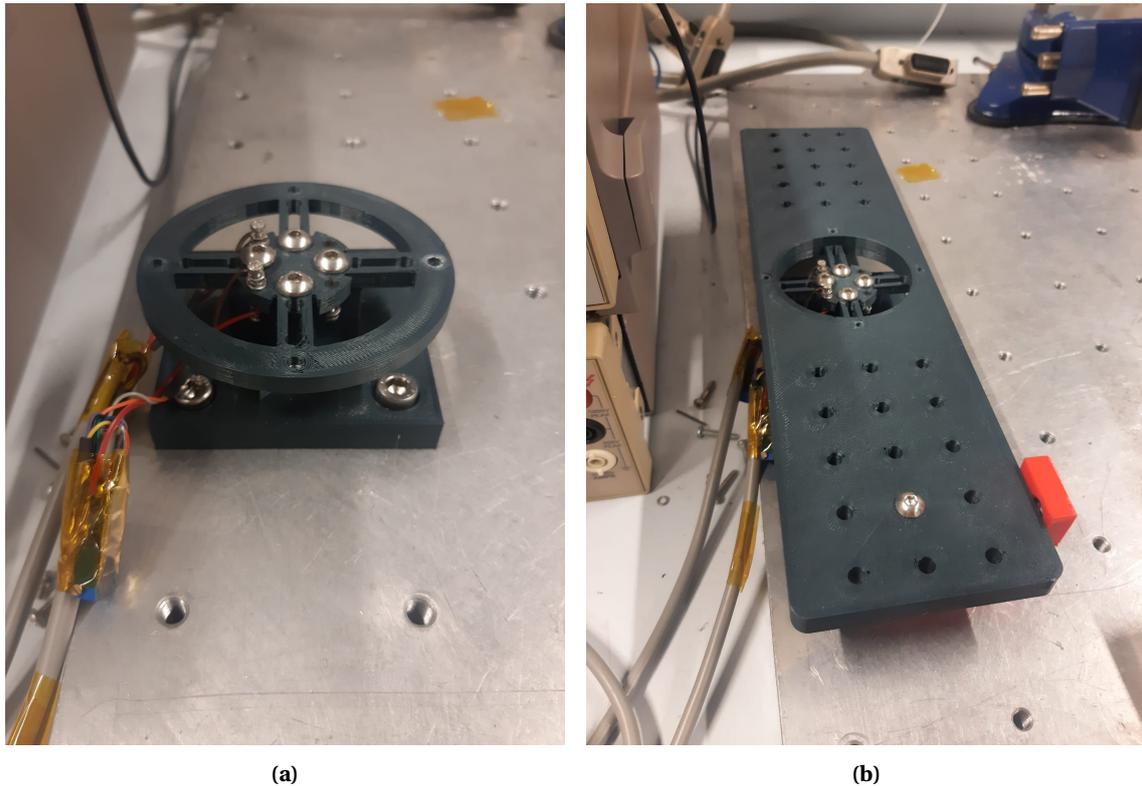
Figure 4.3 shows an annotated overview of the setup that is used for the measurements. The setup is fixed to a aluminum plate with M5 tapped holes for fast assembly. Both 3D printed and metal supporting structural pieces are used to secure the SMAC to the plate. The goal of the setup is to be able to exert a controlled torque on the sensor.



**Figure 4.3:** Annotated overview of the measurements setup.

In order to fix the sensor such that load can be applied, a mount was 3D printed and bolted to the aluminum plate (see Figure 4.4 (a)) such that the sensor can be fixed horizontally. This ensures that any applied force is translated to the strain gauges directly. The figure also shows the cables with the half-bridge configuration that are secured on the strain gauges. Since the SMAC produces a linear force over an arm that is connected to the sensor this results in a torque

(see Figure 4.4 (b)). This figure also show the attachment part (red) for securing the SMAC to the arm.



**Figure 4.4:** (a) Sensor mounted on the aluminum mounting plate. (b) Sensor with extension (dark green) such that applied torque is amplified. SMAC attachment mount (red) for in-plane attachment of SMAC to avoid parasitic movements.

### 4.3 Measurements

In this section the measurements are described. Under force excitation from the linear actuator, the sensor strain gauges changing voltage response is recorded and assessed.

The measurements done by the DEWE are at a sample rate of 20 kHz with a 10 V input voltage to the bridges. From the specifications of the SMAC (Table I.1) we see a peak force of 15 N can be achieved with 1.6 A input. In order not to strain the hardware, it was decided that the maximum force to be exerted would be 12 N. With the extension, an arm of  $l = 0.125\text{m}$  the maximum torque that would be applied would be  $\tau_{\max} = 12l = 1.5\text{Nm}$ .

The SMAC is initialized in force control mode exerting a sine wave torque output. In total, 5 amplitudes are used (2 N, 4 N, 8 N, 12 N), and 6 frequencies (0.2 Hz, 0.5 Hz, 1 Hz, 2 Hz, 4 Hz, 8 Hz), resulting in 30 measurement combinations. Each measurement combinations is done in threefold at 120 seconds per measurement.

### 4.4 Results

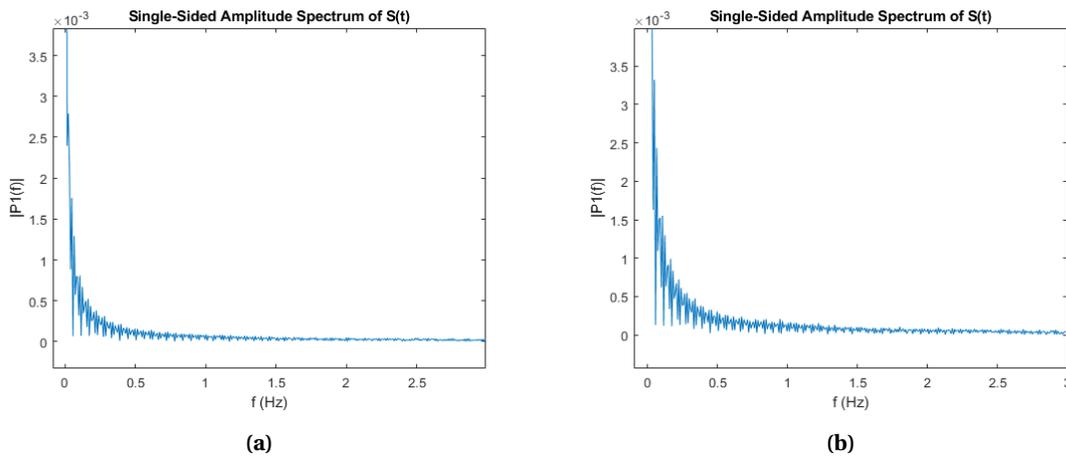
The goal is to present the data such that voltage is plotted against the applied torque to the sensor and later the relative change in resistance. A fit is applied that will describe the behavior of the sensor as a change in voltage with respect to applied torque. Then, the difference between the data and the fit describe the error of the measurement.

The results presented and described in this chapter are based on two data sets. The first being the data of the 2 N, 0.2 Hz sine wave measurement (data set 182) and the second being the data set of the 12 N, 8 Hz sine wave measurement (data set 262). These data sets are chosen to

present such that the largest differences can be seen. Appendix F contains the results of all the measurements.

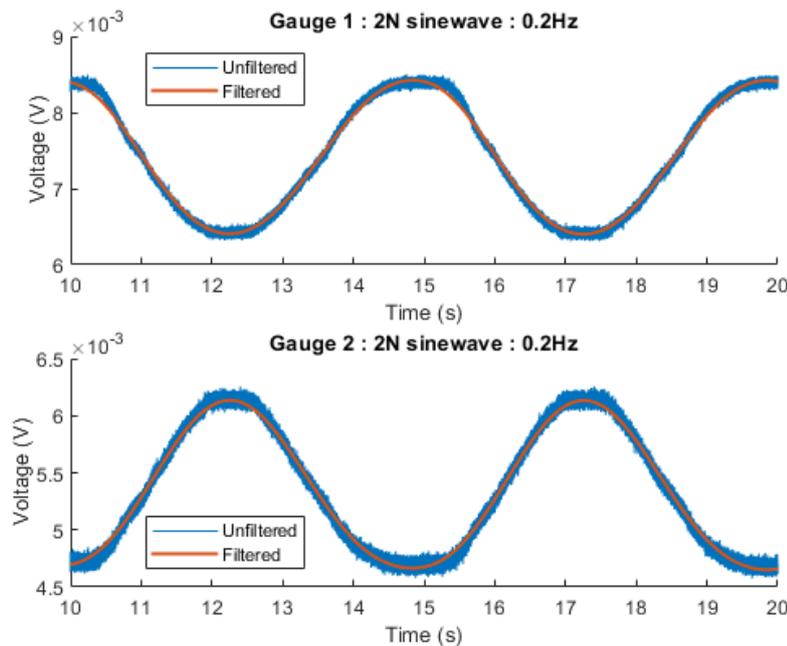
A 2<sup>nd</sup> order Butterworth low-pass filter is applied to the data to suppress high frequent noise and to smooth the signal. This can be done as the filtering is done in post processing as the zero-phase filter filters in both forward and backward direction.

The cutoff frequency at which the filter is applied is chosen to be  $f_c = 10f_{\sin}$ . Where  $f_{\sin}$  is the frequency of the applied sine wave. This is found by analysing the amplitude spectrum in Figure 4.5. The 2 Hz sine wave is used to demonstrate. Around a frequency of 2 Hz the higher order harmonics are well faded into the noise. The factor 10 is thus generously chosen and is used for further frequencies as well.

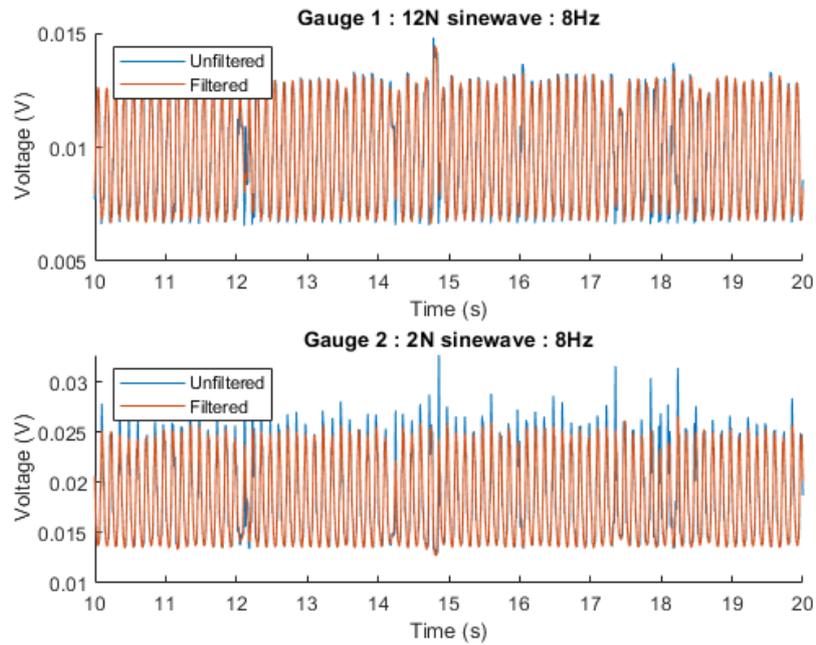


**Figure 4.5:** Single-Sided Amplitude Spectrum of gauge 1 (a) and gauge 2 (b) for an input of  $f_{\sin} = 0.2$  Hz.

Now that the filter is constructed, it is applied to the data sets. Figures 4.6 & 4.7 view the individual unfiltered and filtered sensor responses, plotted against time, of the individual gauges.

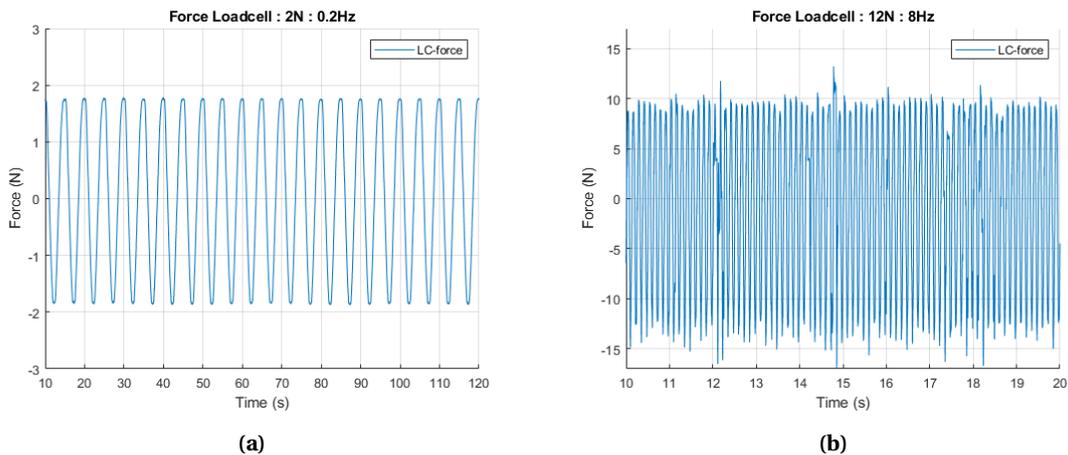


**Figure 4.6:** Sensor response of both individual gauges. In blue the unfiltered data, in orange the filtered data. Sensor is excited with a 2 N, 0.2 Hz sine wave.



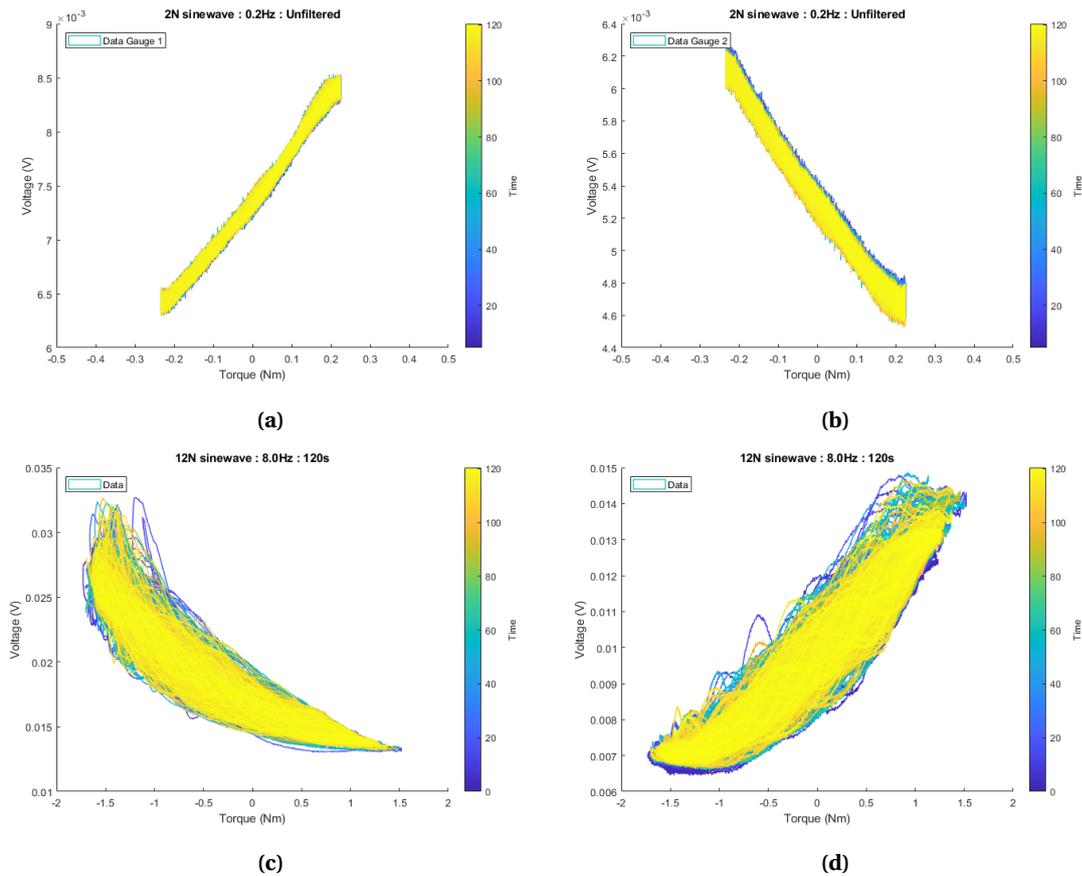
**Figure 4.7:** Sensor response of both individual gauges. In blue the unfiltered data, in orange the filtered data. Sensor is excited with a 12N 8Hz sine wave.

In order to be able to present data against an applied torque, the third data set is used (force measurement with the calibrated load-cell). This contains set of applied forces at any time step during the measurement. Figure 4.8 shows the load-cell output (converted from voltage to force) versus the time. Multiplying the force by the arm, yields the applied torque at any time step.



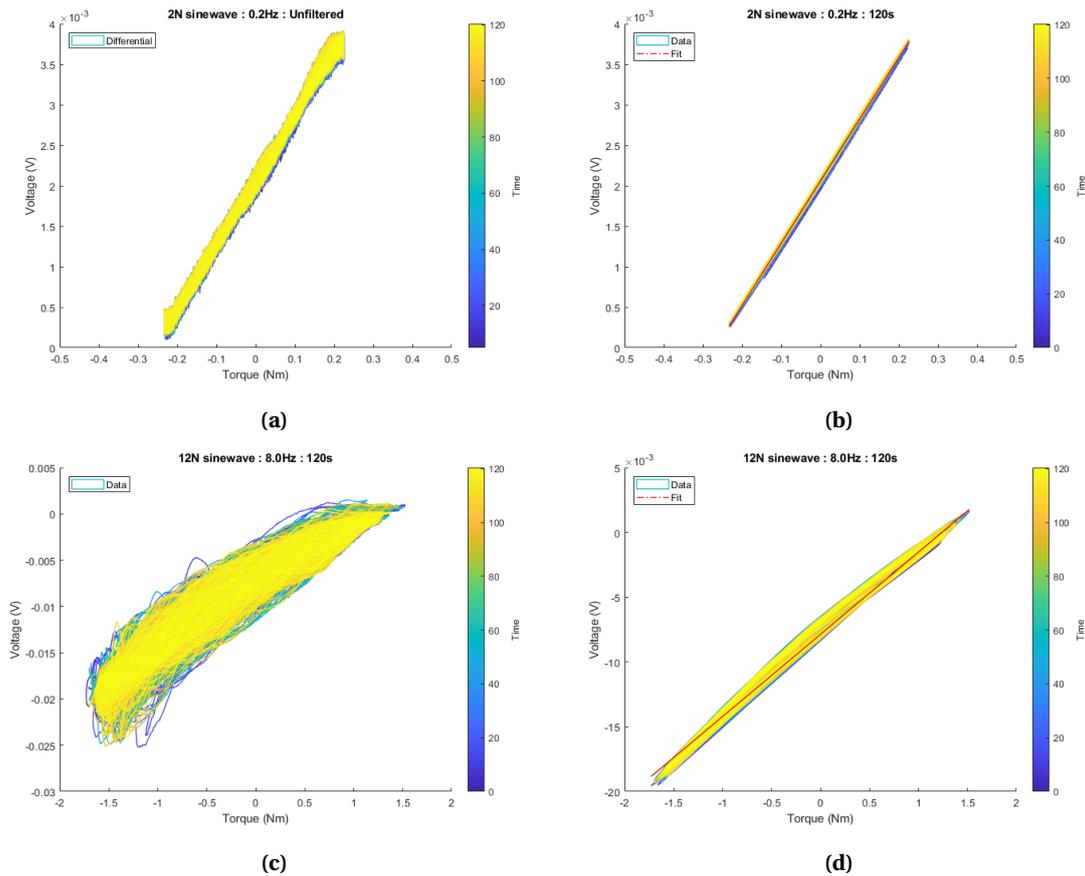
**Figure 4.8:** Load-cell response, converted to force, plotted against time for both data sets.

The next step is to plot the change in voltage to the torque. This will provide further insight in the behavior of the sensor. Figure 4.9 show the unfiltered data of each gauge in both the measurements plotted to the torque. It is seen that the higher frequency generate a larger hysteresis than its lower frequency counterpart. An observation that is made is that the strain gauges present opposing behavior, which is expected.



**Figure 4.9:** Unfiltered sensor response of individual gauges plotted against torque. (a) & (b) respectively correspond to the first and second gauge of the 2N, 0.2Hz set. (c) & (d) respectively correspond to the first and second gauge of the 12N, 8Hz set.

Figure 4.10 (a) & (b) presents the differential measurement in both unfiltered and filtered data of the 2N, 0.2Hz sine wave measurement and Figure 4.10 (c) & (d) presents the differential of the individual gauges in both unfiltered and filtered data of the 12N, 8Hz sine wave measurement. Additionally, a first order polynomial fit (linear fit) is plotted to the filtered data. This linear fit presents the characterized response of the sensor as a voltage difference with respect to torque. It provides an approximation that can be used to predict the behavior.

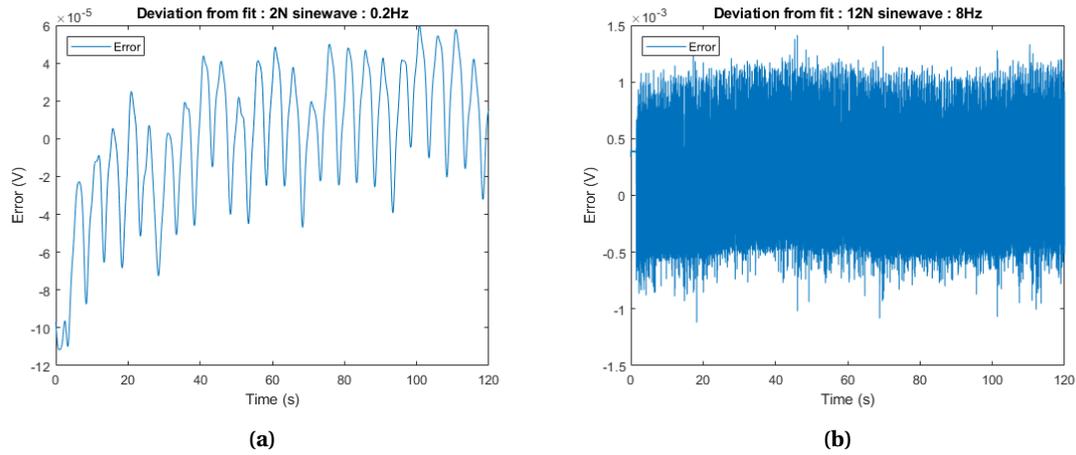


**Figure 4.10:** (a) & (c) present the unfiltered differential of both data sets & (b) & (d) present the filtered differential with a linear fit plotted against the applied torque.

The fits of these data sets are respectively

$$y = 0.0076x + 0.0019 \quad \& \quad y = 0.0063x - 0.0079 \quad (4.1)$$

for data sets 182 and 262. With the linear fit presenting the approximation of the behavior, error is then shown in Figure 4.11. The error of Figure 4.11 (a) is observed to be two orders of magnitude lower than the corresponding data from Figure 4.10 (b). This is a marginal error, one that allows for response to be assumed linear in this case. The error of Figure 4.11 (b) is observed to be only one orders of magnitude lower than the corresponding data from Figure 4.10 (d). This error is more significant (up to 0.3 N m) and assuming linearity is thus more risky.



**Figure 4.11:** Difference plot between filtered data and fit for both measurements against time. (a) Data set 182, (b) data set 262.

To calculate the relative change in the resistance  $\frac{\Delta R}{R}$  according to

$$\frac{\Delta R}{R} = \frac{R_s - \hat{R}_s}{\hat{R}_s} \quad (4.2)$$

we need to compute the sensor resistance  $R_s$  and the mean sensor resistance  $\hat{R}_s$ . The equation sensor resistance can be rewritten from the voltage divider equation

$$V_m = \frac{R_s}{R_p + R_s} V_0 \quad (4.3)$$

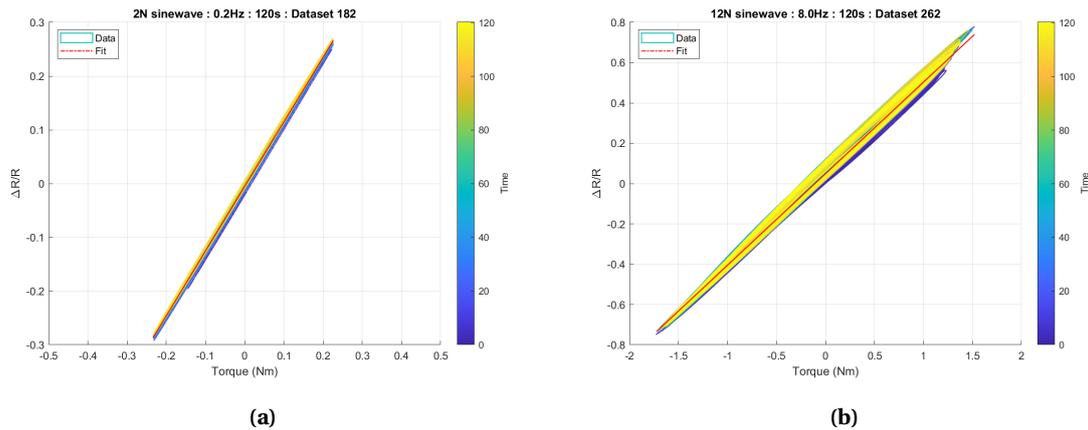
with  $V_m$  being the measured voltage,  $V_0$  the input voltage and  $R_p$  the potentiometer resistance (in balanced state). Solving Equation 4.3 for  $R_s$  yields

$$R_s = \frac{-V_m R_p}{V_m - V_0}. \quad (4.4)$$

The mean resistance  $\hat{R}_s$  is now computed as the sum off all values divided by the number of values

$$\hat{R}_s = \frac{1}{n} \sum_{k=1}^n R_{s_k} \quad (4.5)$$

Substituting Equation 4.5 & 4.4 into Equation 4.2 results in the relative resistance change for the data sets. Figure 4.12 view these plots and show relative resistance changes up to 75 % for a torque of 1.7 N m.



**Figure 4.12:** (a) & (b) present the filtered differential relative change in resistance with a linear fit plotted against the applied torque, respectively for both data sets.

The differential response significantly reduces out the non-linearities in the response of both gauges,

From these results we conclude that the symmetry of the gauges is of vital importance as the differential response significantly reduces the non-linearities in the response of the sensor.

This same conclusion is made in other studies such as [40] & [44]. From these results (Appendix F) we conclude a linear behavior that is more prevalent with a low excitation torque. An increase of the excitation frequency is observed to induce a hysteresis behavior which results in a larger error.

#### 4.5 Conclusion

The experimental setup to characterize the sensor is described in detail. Some of the obtained measurement results are presented and their post processing is discussed. For each measurement two opposing gauges are excited with the same input and their responses are logged from which a differential measurement is made. The resulting plots are used to characterize the sensor based on a linear fit that is used later in this research.

The results of the response when the sensor is excited with a low frequency, low torque input show a behavior that can be assumed linear with an error that is two orders of magnitude lower than its response. Increasing the input frequency and torque show a less linear response that is mainly attributed to hysteresis.

## 5 Control

### 5.1 Introduction

The main goal of this chapter is the realisation of the application that is described in Chapter 3.2, and controlling its behavior such that the sensor can be validated. This goal is separated in several sub-goals. The first sub-goal is to describe the dynamics of the pendulum plant numerically and use that description to simulate the plant and use this plant to simulate a controlled system with desired behavior. After that the next sub-goal is to compute the plants transfer function that is needed to design a dynamics cancelling controller by discretizing this transfer function such that it can be described by a difference function.

### 5.2 Dynamics

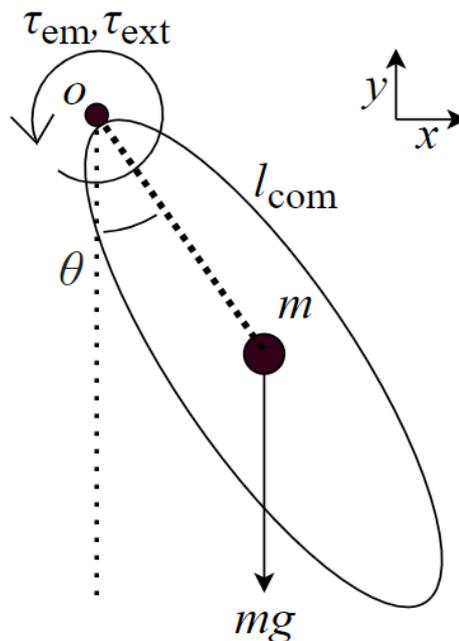
Recall the equation 2.4 for the computation of the torque in a robot joint space. For convenience, it is shown once again below.

$$M(q)\ddot{q} + C(q, \dot{q}) + g(q) + h(q, \dot{q}) + \tau_{\text{ext}} = \tau \quad (5.1)$$

The matrices  $M(q)$ ,  $C(q, \dot{q})$  and  $g(q)$  are to be computed according to the theory from Section 2.4.  $q$  is the current state of the system (angle  $\theta$  in this 1 DOF case). Before computing, the pose of the pendulums COM is defined as

$$P_{\text{com}} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} l \sin(\theta) \\ -l \cos(\theta) \\ \theta \end{bmatrix}, \quad (5.2)$$

using the orientations as defined in Figure 5.1.



**Figure 5.1:** Schematic overview of a rigid body pendulum driven by a stepper motor around  $o$ .

**Inertia Matrix** As seen in Equation 2.16, the computation of  $M(q)$  requires the analytical Jacobian  $J_{\text{com}}$  and mass matrix  $O_{\text{com}}$ . The Jacobian was defined in Equation 2.12 as

$$J_{\text{com}} = \frac{\partial P_{\text{com}}}{\partial q} = \begin{bmatrix} \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} \\ \frac{\partial \theta}{\partial \theta} \end{bmatrix} = \begin{bmatrix} l \cos(\theta) \\ l \sin(\theta) \\ 1 \end{bmatrix} \quad (5.3)$$

and the mass matrix was computed from the kinetic energy in Equation 2.14 as

$$O_{\text{com}} = \begin{bmatrix} m_{\text{com}} & 0 & 0 \\ 0 & m_{\text{com}} & 0 \\ 0 & 0 & I_b \end{bmatrix}. \quad (5.4)$$

Now, computing the Inertia Matrix as per Equation 5.5 yields

$$M(q) = J_{\text{com}}(q)^T O_{\text{com}} J_{\text{com}}(q) = \begin{bmatrix} l \cos(\theta) \\ l \sin(\theta) \\ 1 \end{bmatrix}^T \begin{bmatrix} m_{\text{com}} & 0 & 0 \\ 0 & m_{\text{com}} & 0 \\ 0 & 0 & I_b \end{bmatrix} \begin{bmatrix} l \cos(\theta) \\ l \sin(\theta) \\ 1 \end{bmatrix} = ml^2 + I_b. \quad (5.5)$$

**Coriolis Forces** Literature showed that the  $C(q, \dot{q})$  matrix has relations with the inertia matrix. The relation was shown in Equation 2.18 where the Coriolis Matrix is the summation of the Christoffel symbols and  $\dot{q}_k$  (the generalized velocity vector). With Inertia Matrix  $M(q)$  computed as a constant scalar we see that the Coriolis forces of this configuration are  $C = 0$  as its derivative will be 0.

**Gravitational Torque** For a single joint pendulum, the torque induced by gravity on the motor shaft is calculated as

$$\tau_g = m_{\text{com}} g l \sin(q), \quad (5.6)$$

with  $q$  being the current angle.

**Residual Torques** As mentioned, the residual torque are generalized through the viscous damping, for this system a coefficient  $B$  is used. Frictional torques are a considered linear to the rotational velocity  $\dot{q}$  and caused by resistance in a system. They also are in opposite direction of the motion. The frictional force will thus be

$$\tau_{\text{fric}} = B \dot{q}. \quad (5.7)$$

**External Torques** Any external torque will be generalized and denoted by  $\tau_{\text{ext}}$ .

**Result** The resulting equation for computation of torque in a robot joint as per Equation 2.4 can now be written as

$$\tau = (ml^2 + I_b) \ddot{q} + B \dot{q} + m_{\text{com}} g l \sin(q) + \tau_{\text{ext}}. \quad (5.8)$$

### 5.2.1 Simulation

Simulation is done in Simulink. A continuous-time plant is described with the differential equations of a simple pendulum. This plant represents the system in the simulation. Table 5.1 contains the system variables that are needed in this simulation.

**Table 5.1:** Variables of the pendulum system.

Variable	Description	Value	Unit
$I_b$	Inertia of body	0.00077	$\text{kg} * \text{m}^3$
$I_{pa}$	Inertia parallel axis	0.0022	$\text{kg} * \text{m}^3$
$l_{com}$	Length to COM	0.145	m
$m_b$	Mass of body	0.0668	kg
$B$	Friction coefficient	0.01	$\frac{\text{Nm} * \text{s}}{\text{rad}}$

Inertia  $I_b$  is estimated by considering the pendulum a solid cuboid as this simplifies the estimation. The resulting value will thus not entirely accurate. Its effect will however not be significant, especially when the acceleration (the effect of the inertia is depending on the acceleration) is low.

This yields the equation for the moment of inertia around its centre of mass as

$$I_b = \frac{1}{12} m_b (l^2 + w^2) \quad (5.9)$$

with  $l$  and  $w$  respectively being the length and with of the cuboid. In our case, length  $l$  is  $0.368m$  (estimation for a solid cuboid) and width  $w$  is  $0.0486m$ . The inertia thus becomes

$$I_b = \frac{1}{12} 0.0668 (0.368^2 + 0.0486^2) = 0.00077 \text{ km}^{-3}. \quad (5.10)$$

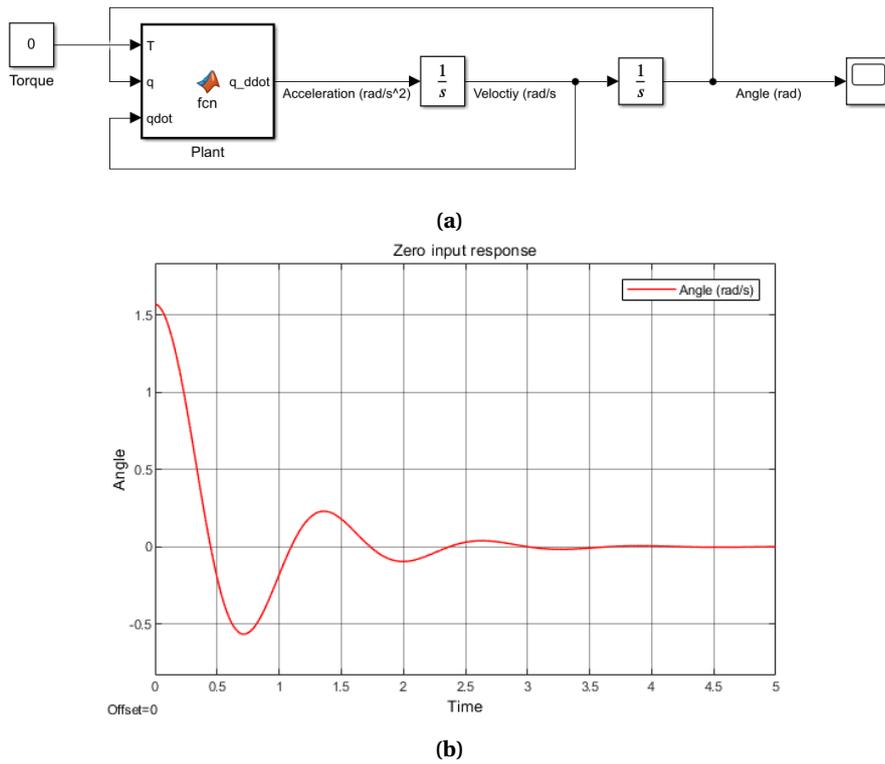
We need the inertia around the rotating axis of the motor shaft. Shifting the axis of the moment of inertia to a parallel one  $I_{pa}$  (the axis of the motor shaft) is done by applying the Parallel Theorem [63] from Equation 5.11. In this equation  $r$  is the distance to the parallel axis, this is distance  $l_{com}$

$$I_{pa} = I_{com} + m_b r^2 = 0.00077 + (0.0668 * 0.145^2) = 0.0022 \text{ km}^{-3} \quad (5.11)$$

**Simulink** The plant is described using the dynamics as seen from Equation 5.8, and the system variables from Table 5.1 in a function block in MATLAB's Simulink. Solving the dynamics for the acceleration yields the following equation of motion (EOM) for the pendulum:

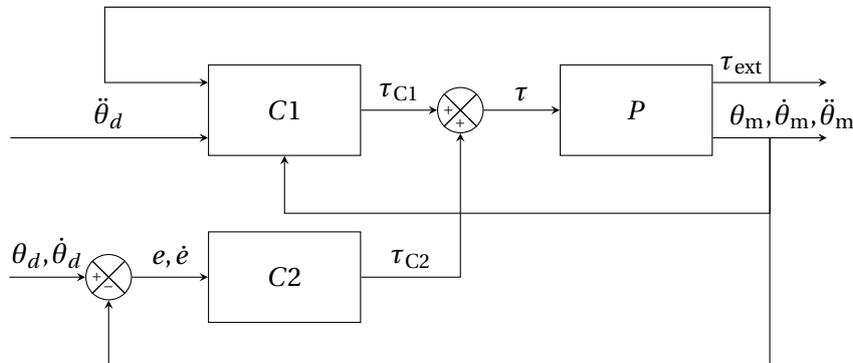
$$\ddot{q} = \frac{\tau - B\dot{q} - m_{com}gl \sin(q) - \tau_{ext}}{ml^2 + I_{pa}}. \quad (5.12)$$

To test the dynamics, a zero input response is plotted with an initial angle of  $\frac{1}{2}\pi$  to see if the response is as expected. The Simulink model and the zero input response is viewed in Figure 5.2. From Equation 5.12 we see that for a torque input an acceleration output is computed. To get to the angle, the output is integrated twice by the an integration block ( $\frac{1}{s}$ ).



**Figure 5.2:** (a) Simulink model with the EOM implemented. (b) Response of the plant to zero input torque.

The next step is to simulate the system in a controlled environment with external inputs. A block schema of this system is presented in Figure 5.3.



**Figure 5.3:** Block schema of the simulated system.

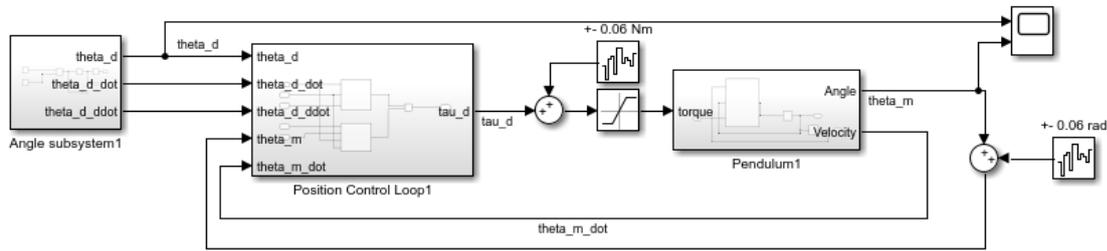
In this figure, block  $P$  is the plant as described earlier, block  $C1$  is the dynamics cancelling controller following from Equation 5.8. It takes the desired acceleration  $\ddot{\theta}_d$ , measured velocity  $\dot{\theta}_m$  and external torques  $\tau_{ext}$  as inputs. It results in torque  $\tau_{C1}$ .

$C2$  is a PD controller that adjusts for errors caused by the disturbances and is implemented as

$$C2 = k_p(\theta_d - \theta_m) + k_d(\dot{\theta}_d - \dot{\theta}_m). \tag{5.13}$$

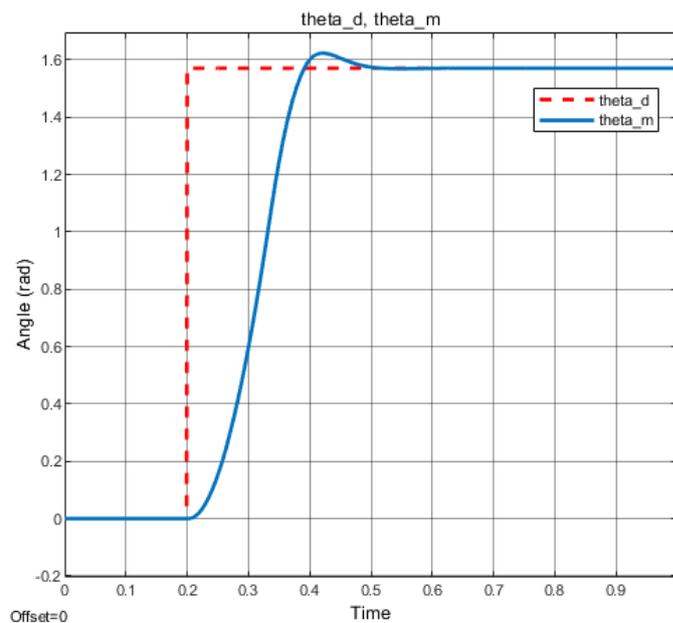
With proportional gain  $k_p = 5$  and derivative gain  $k_d = 0.2$ , these values were chosen based on several simulations and were deemed sufficient. Resulting in the adjustment torque  $\tau_{C2}$ .

Figure 5.4 shows the Simulink model. The disturbances on the motor input and output are generated and set randomly with to  $\pm 0.06$



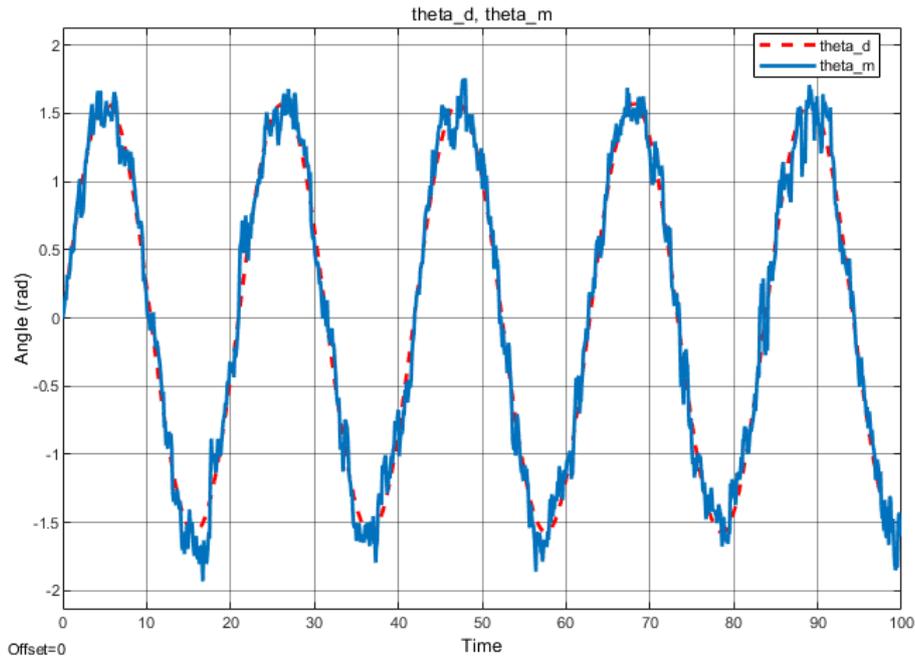
**Figure 5.4:** Simulink model based on the block scheme from Figure 5.3. Both controller  $C1$  and  $C2$  are bundled in the Position Control Loop block and the disturbance is added before the plant.

The step response (without any disturbances and a step of  $\frac{1}{2}\pi$ ) is viewed in Figure 5.5. It shows the transient behavior of the controller from which some behavioral aspects can be analyzed. The response is slightly underdamped resulting in minimal over- and undershoot with a settling time around 0.3 s. The steady state error is zero.



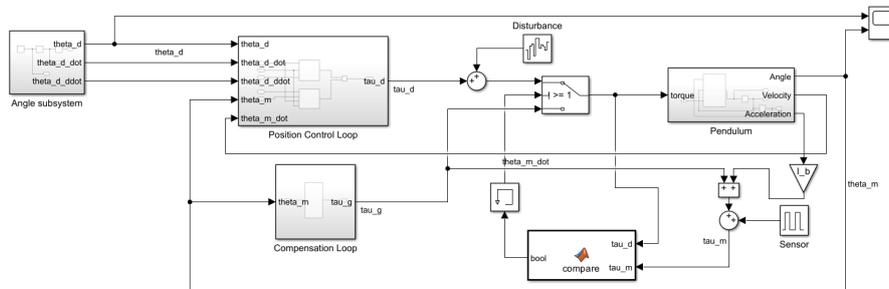
**Figure 5.5:** Step response of the simulated system. A step of  $\frac{1}{2}\pi$  is made.

For an sinusoidal input signal with a  $\frac{1}{2}\pi$  amplitude and  $0.3 \frac{\text{rad}}{\text{s}}$  frequency, and a random disturbance on both the input of the motor and the output, the resulting plot is viewed in Figure 5.6. It shows that the response with disturbance (which is up to 30% of the motor input and  $4^\circ$  on the output) follows the reference well. Note that is this an input response.



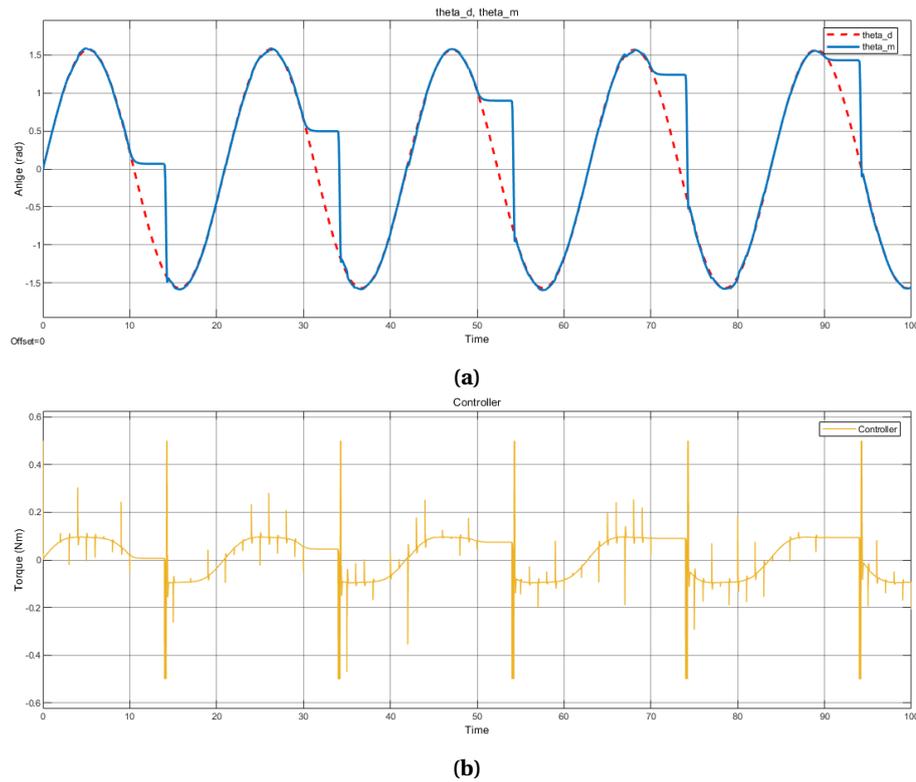
**Figure 5.6:** Resulting plot for the controlled system with time on the x-axis and the angle  $\theta$  on the y-axis. Blue presents the measured angle  $\theta_m$  and dashed red the desired angle  $\theta_d$ .

Finalizing the simulation, the aforementioned interaction layer is implemented. In this layer the desired torque is compared to the torque measured by the sensor. When the measured torque exceeds a set threshold (e.g. a collision) the system should stop moving and stay in place. It thus switches between the position control loop and a gravity compensation loop when the threshold is exceeded. The Simulink model of this system is presented in Figure 5.7.



**Figure 5.7:** Simulink model of the system with interaction control.

This Simulation yields the following result as seen in Figure 5.8. The same reference signal is used as before. The external torque measured by the sensor is simulation using a pulse generator and added to the output of the plant. A 0.5 N m pulse is generated every 20 seconds (with a 10 second delay).

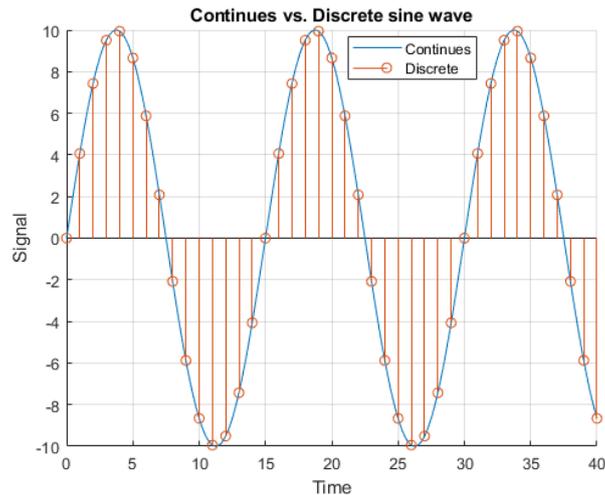


**Figure 5.8:** Resulting plots for the controlled system with the interaction control layer. (a) With time on the x-axis and the angle  $\theta$  on the y-axis. Blue presents the measured angle  $\theta_m$  and dashed red the desired angle  $\theta_d$ . (b) Shows the corresponding controller output.

We see the expected behavior of the interaction, at intervals of 20 seconds the system encounters a collision and stops its motion for several seconds before continuing its desired path.

### 5.2.2 Discretization

The application is driven by a micro-controller (more on that in Section 5.4). In order for the desired controller to be used it will have to be described in discrete-time. If a signal can be defined for all values of time, it is considered a continuous-time system, whereas a discrete-time system defines a signal at set time steps. Figure 5.9 shows a continuous sinusoidal signal and its discrete counterpart.



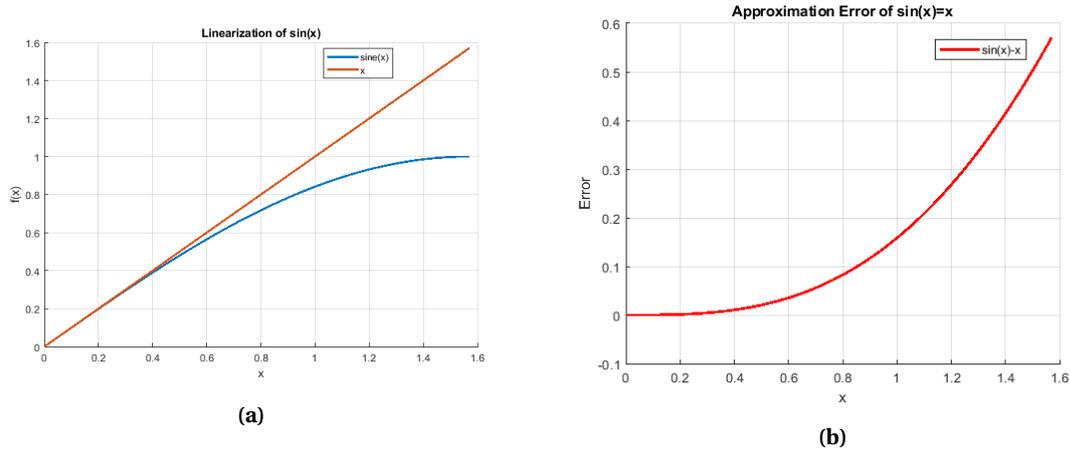
**Figure 5.9:** In blue a continuous-time signal, defined for all values of time. In red the same signal discretized, defined only at set time steps and considered constant until the next.

Discretization of a system makes them suitable for numerical implementation in digital environments. The first step of this process is to convert the continuous-time non-linear system to a transfer function in the Laplace domain. The Laplace transform is universally used and converts a function  $f(t)$  of time to a function  $F(s)$  in the frequency domain. A Laplace transformed function (now called transfer function) describes two signals as a function of  $s$  [64]. A large benefit of the Laplace domain is that complex operations such as integration and differentiation can now be expressed algebraically [64]. It does however introduce to concept of *linearization*. A non-linear time invariant (NTI) system, a system that is not a direct function of time such as our pendulum, cannot be expressed in the Laplace domain in its entirety. The Laplace transform convert an NTI system to a linear time invariant (LTI) version of that system at some initial conditions.

**Linearization** As mentioned, transforming an NTI system to its equivalent Laplace domain transfer function requires a linearization of that function at a certain operating condition. Linearization takes the response at the initial condition and considers that to be linear for all time. From our pendulum we know that its response is a non-linear function of its angle  $\theta$  where its position is determined by a sinusoid. Linearizing this function around  $\theta = 0$  yields the small-angle approximation [65].

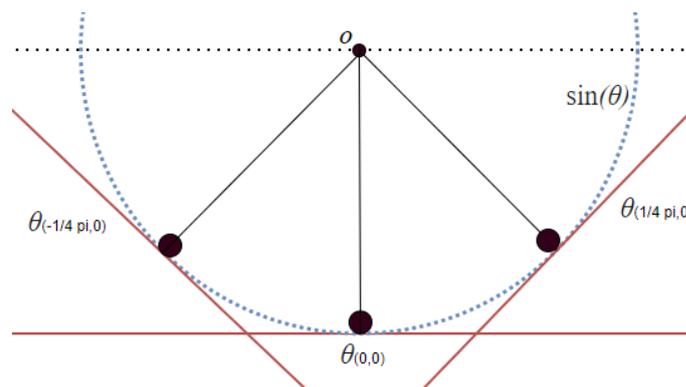
$$\sin \theta \approx \theta. \quad (5.14)$$

This is an approximation for a small range of  $\theta$ . Generally, equation 5.14 can be assumed to hold for angles up to  $15^\circ$  (0.26 rad). Figure 5.10 shows this approximation and the corresponding error between the linear and non-linear function. From this graph a linearization error of  $e = 0.003$  rad at  $\theta = 0.26$  rad is found.



**Figure 5.10:** (a) Comparison between  $\sin(\theta)$  and  $\theta$ . (b) Approximation error. Both axis are in radians.

While this error is small it increases and thus cannot be used as an expression of the entire system. A visualisation of this concept is viewed in Figure 5.11 with different initial conditions for the angle and velocity  $(\theta, \dot{\theta})$ . The blue (half) circle represents the continuous-time non-linear position of the pendulum at  $\theta$ . The red lines are linearizations at certain initial conditions. They are represented by a tangent line to the circle at the position, three linearizations are viewed.



**Figure 5.11:** The system at three positions  $(0, 0)$ ,  $(\frac{1}{4}\pi, 0)$  and  $(-\frac{1}{4}\pi, 0)$ . The blue dotted line represents the position of the pendulum using the non-linear sine function of the angle  $\theta$ . The red lines represent three linear approximation around these positions.

**Gain Scheduling** From Figure 5.11 it becomes clear that for this system to be controlled over a larger bandwidth (say  $-\frac{1}{2}\pi$  to  $\frac{1}{2}\pi$ ), a linearization around one operating point will not suffice. Having several linear controllers, at multiple operating points, and switching between them, can provide a method for accurately controlling an extended range of angles. This process is a form of the so-called gain scheduling. Gain scheduling is a method for controlling nonlinear systems by adjusting the controller based on a set of system variables [66]. In case of our pendulum, the system variables would be the current position and based on this position different controllers are activated. As mentioned, the small-angle approximation holds for a range of  $15^\circ$  (0.26 rad). Using this method, in total 12 linear controllers are needed to cover the entire circle effectively.

**Continuous-time transfer function** To describe the pendulum EOM as a continuous-time transfer function in the Laplace domain and define controller  $C1$  from Figure 5.3 the following notations are used.

$$\begin{aligned}\vec{x} &= \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} \\ \dot{\vec{x}} &= \begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ -\frac{mgl}{I_b} \sin \theta - \frac{B}{I_b} \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I_b} \end{bmatrix} \tau = f(x, \tau).\end{aligned}\quad (5.15)$$

The rate of change is then described as

$$\begin{aligned}\Delta \vec{x} &= x - x_0 \\ \Delta \dot{x} &= f(x_0, \tau) + \left( \frac{\partial f}{\partial x} \Big|_{x=x_0} (x_0, \tau) \right) \Delta x,\end{aligned}\quad (5.16)$$

where

$$\frac{\partial f}{\partial x} = \frac{\partial}{\partial x} \begin{bmatrix} \dot{\theta} \\ -\frac{mgl}{I_b} \sin \theta - \frac{B}{I_b} \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta} & \frac{\partial f_1}{\partial \dot{\theta}} \\ \frac{\partial f_2}{\partial \theta} & \frac{\partial f_2}{\partial \dot{\theta}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{mgl}{I_b} \cos \theta & -\frac{B}{I_b} \end{bmatrix}.\quad (5.17)$$

With  $x_0$  being the initial condition and point around which is to be linearized. The next step is to introduce the Laplace operator  $s$  using the state transition matrix. Equation 5.16 becomes

$$\begin{aligned}\left( sI - \frac{\partial f}{\partial x} \Big|_{x=x_0} (x_0, \tau) \right) \Delta x &= f(x_0, \tau) \\ \Delta x &= \left( sI - \frac{\partial f}{\partial x} \Big|_{x=x_0} (x_0, \tau) \right)^{-1} f(x_0, \tau).\end{aligned}\quad (5.18)$$

With

$$\left( sI - \frac{\partial f}{\partial x} \Big|_{x=x_0} (x_0, \tau) \right) = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\frac{mgl}{I_b} \cos \theta & -\frac{B}{I_b} \end{bmatrix} = \begin{bmatrix} s & -1 \\ \frac{mgl}{I_b} \cos \theta_0 & s + \frac{B}{I_b} \end{bmatrix}.\quad (5.19)$$

Taking the inverse of this square matrix  $A$  (solution of Equation 5.19), following

$$A^{-1} = \frac{1}{\det(A)} \text{adj}(A),\quad (5.20)$$

yields

$$A^{-1} = \frac{1}{s(s + \frac{B}{I_b}) + \frac{mgl}{I_b} \cos \theta_0} \begin{bmatrix} s + \frac{B}{I_b} & 1 \\ -\frac{mgl}{I_b} \cos \theta_0 & s \end{bmatrix}.\quad (5.21)$$

Substituting Equation 5.21 in Equation 5.18 gives

$$\Delta \vec{x} = \frac{1}{s^2 + \frac{B}{I_b} s + \frac{mgl}{I_b} \cos \theta_0} \begin{bmatrix} s + \frac{B}{I_b} & 1 \\ -\frac{mgl}{I_b} \cos \theta_0 & s \end{bmatrix} f(x_0, \tau).\quad (5.22)$$

Substituting  $f(x_0, \tau)$  results in expression for the transfer function  $H(s)$ , for input  $\tau$  to output  $\theta$ , in the first row of  $\Delta x$ . So

$$\Delta x_1 = \frac{s\dot{\theta}_0 - \frac{mgl}{I_b} \sin\theta_0 + \frac{\tau}{I_b}}{s^2 + \frac{B}{I_b}s + \frac{mgl}{I_b} \cos\theta_0} = H(s)\left(s\dot{\theta}_0 - \frac{mgl}{I_b} \sin\theta_0 + \frac{\tau}{I_b}\right). \quad (5.23)$$

With the output  $\Delta x$  equals an expression times the input  $\tau$  and its initial input  $u_0$ . Thus our system's continuous-time transfer function  $H(s)$  is expressed as

$$H(s) = \frac{1}{s^2 + \frac{B}{I_b}s + \frac{mgl}{I_b} \cos\theta_0}. \quad (5.24)$$

As described, the second controller from Figure 5.3 contains the error handling PD controller. In the  $s$ -domain a PD controller looks like

$$K(s) = k_p + k_d s \quad (5.25)$$

with  $k_p$  as the proportional gain and  $k_d$  the derivative gain [67].

**Discrete-time transfer function** The next step in this process is converting the continuous-time transfer function to its discrete counterparts. First, the MATLAB `c2d` command is used to convert the continuous-time transfer function to the discrete  $z$ -domain via a zero-order hold (ZOH) discretization method and a time step  $t_s = \frac{1}{3000}$  s. ZOH assumes a piecewise constant input over the sample time (e.g. torque input does not change in between time steps). The time step is a result of the maximum clock frequency of the micro-controller, more on this in the next section. The discrete-time transfer function is then described by

$$H(z) = \frac{5.554e^{-8}z + 5.552e^{-8}}{z^2 - 1.999z + 0.9991}. \quad (5.26)$$

The discretized transfer function of the PD controller becomes

$$K(z) = \frac{65z - 55}{z + 1} \quad (5.27)$$

with proportional gain  $k_p = 2.5$  and derivative gain  $k_d = 0.01$ . These values were selected through trial and error. The ZOH discretization method is used for  $H(z)$  and the `tustin` method for  $K(z)$  as the ZOH method cannot handle improper systems. A improper system being a system in which degree of the numerator has exceeded the degree of the denominator, such as  $K(s)$ .

**Difference equation** A micro-controller cannot interpret transfer functions even when they are in the discrete domain. For that, a discrete LTI transfer function has to be converted to a difference equation. The difference equation is a description of a transfer function based on differences of successive values of its input  $X(z)$  and output  $Y(z)$  [68]. Converting these values back to the time domain, with timestep  $k$ , yields an expression where old values for input and output are used to compute the current output, where  $X(z)z = x[k]$ ,  $X(z)z^{-1} = x[k-1]$ . Using the transfer function from 5.26 the relation to its input and output is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{5.554e^{-8}z + 5.552e^{-8}}{z^2 - 1.999z + 0.9991}. \quad (5.28)$$

Cross multiplication and division by the higher order of  $z$  yields

$$Y(z)(1 - 1.999z^{-1} + 0.9991z^{-2}) = X(z)(5.554e^{-8}z^{-1} + 5.552e^{-8}z^{-2}). \quad (5.29)$$

Converting this expression gives the difference equation

$$y[k] - 1.999y[k - 1] + 0.9991y[k - 2] = 5.554e^{-8}x[k - 1] + 5.552e^{-8}x[k - 2]. \quad (5.30)$$

Solving for the output  $y[k]$  gives

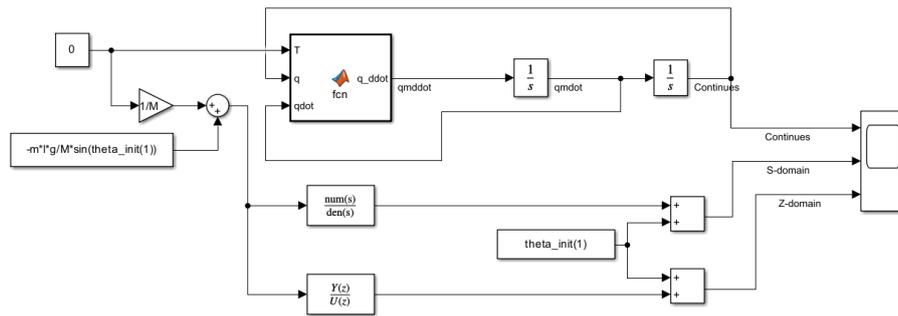
$$y[k] = 5.554e^{-8}x[k - 1] + 5.552e^{-8}x[k - 2] + 1.999y[k - 1] - 0.9991y[k - 2]. \quad (5.31)$$

This equation can be used in a micro-controller to determine output  $y[k]$  with input  $x[k]$  based on the results from the previous two time steps.

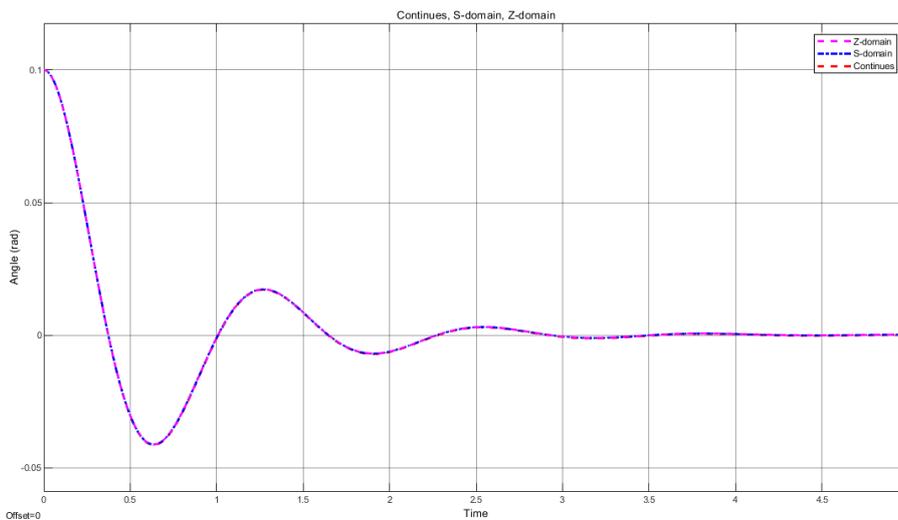
This same process is done for the the PD controller from Equation 5.27 resulting in the difference equation

$$y[k] = 65x[k] - 55x[k - 1] - y[k - 1]. \quad (5.32)$$

**Simulation** To see if the transfer function is indeed a representation of our system, we simulate the response to zero input torque of both transfer functions 5.24 & 5.26. The behavior is expected to be the same as for the simulated pendulum in Figure 5.2 and it will thus be compared to it. In Figure 5.12 the Simulink system and its response are shown. The system is linearized around operating point  $(\theta_0, \dot{\theta}_0)$  that is  $(0, 0)$ . The initial position  $(\theta, \dot{\theta})$  is at  $(0.1, 0)$ .



(a)



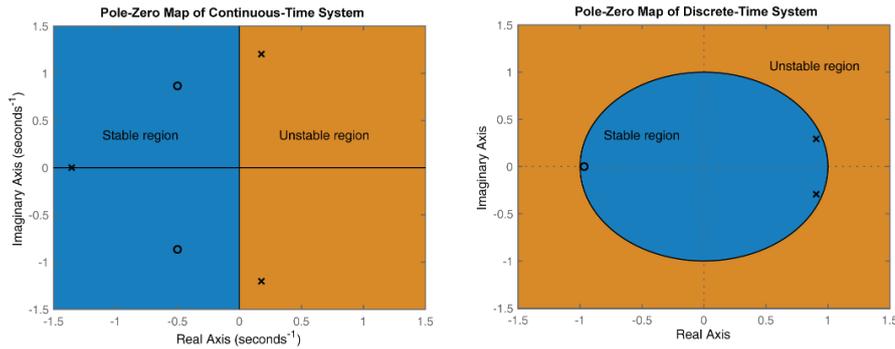
(b)

**Figure 5.12:** (a) Simulink model applying a zero torque input to the continuous-time system, the continuous-time s-domain transfer function and the discrete z-domain transfer function. (b) System response where the angle is shown in time. The initial position is  $(0.1, 0)$ .

The response of both transfer functions is as expected, an oscillation and an equilibrium at  $\theta \approx 0$ . A small steady state error is present due to the linearization but is neglectable.

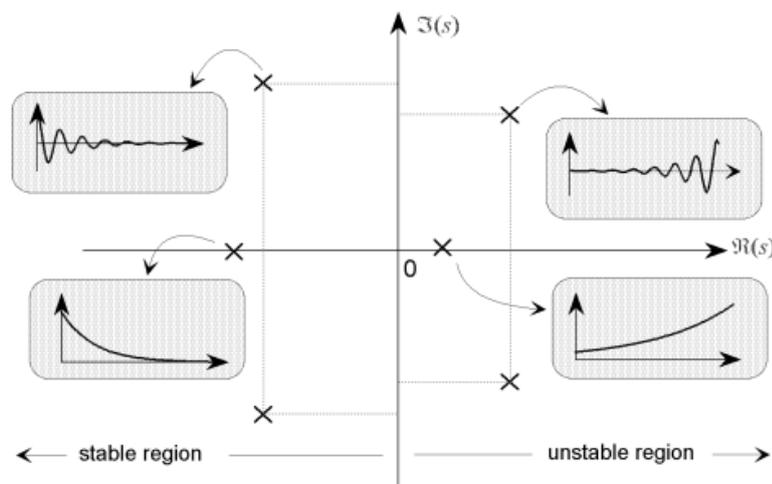
### 5.3 Analysis

We analyze the transfer functions by looking at their pole and zero distribution on the complex plane. The results of the numerator and denominator respectively present the zeros (o) and poles (x). From the location of the poles and zeros insights into the response characteristics of a system is gained. In Figure 5.13 the stability regions for both continuous-time and discrete-time responses are shown.



**Figure 5.13:** Pole zero maps and their stability regions for both continuous-time and discrete-time LTI transfer functions.

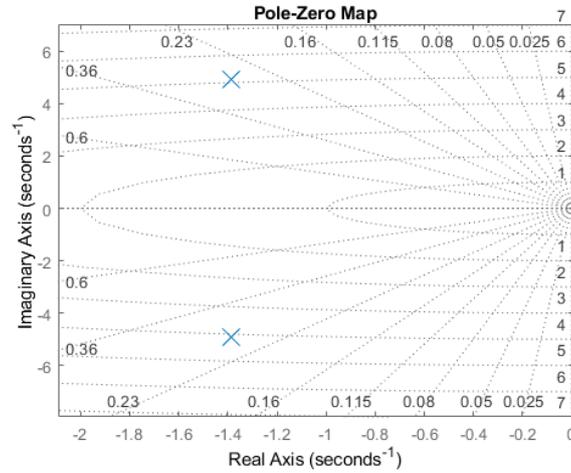
**Continuous-time Transfer Function** From Figure 5.13 we see that, for a system to be stable, its poles have to be in the negative side of the spectrum. Figure 5.14 shows the response of a continuous-time system based on its pole locations. Poles in the positive side of the spectrum causes oscillation and for the signal to diverge. Zeros on the other hand do not directly contribute to the stability of a system, they can affect the overall behavior of the system. For example, if the zeros are in the right half of the complex plane, the overall response of the system will be more sensitive to the poles and could lead to an unstable system even if the poles are in the left half of the complex plane.



**Figure 5.14:** Response of a continuous-time LTI system to pole locations [10].

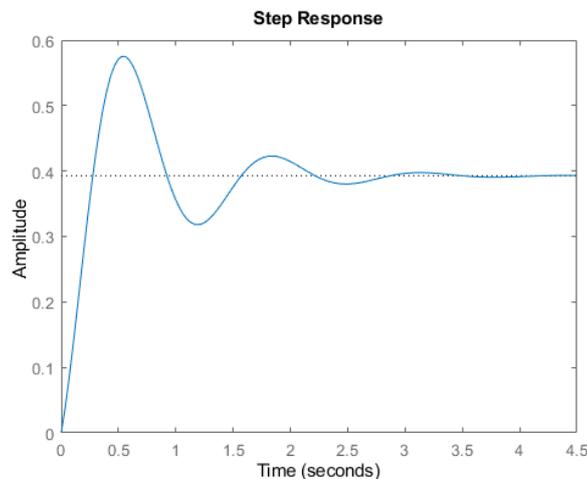
Looking at our computed transfer function from Equation 5.24 we can determine its poles and zeros using the `pzmap` command in MATLAB (using the values from Table 5.1,  $k_p = 5$ ,  $k_d = 0.02$

and the initial conditions of  $(\theta_0, \dot{\theta}_0)$  that is  $(0, 0)$ . Figure 5.15 shows this map, we see both the poles in the negative plane and can thus determine that this system is stable.



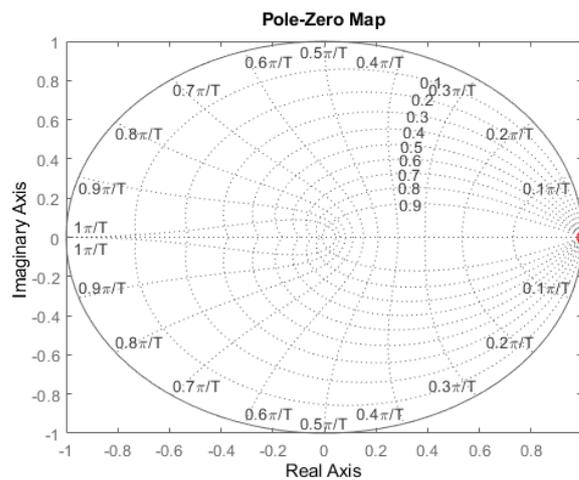
**Figure 5.15:** Poles of the continuous-time LTI transfer function.

This also becomes apparent when plotting the step response of this function in Figure 5.16 where we observe convergence of the response after some initial oscillation. This oscillation is expected as the friction coefficient  $B$  is relatively low.



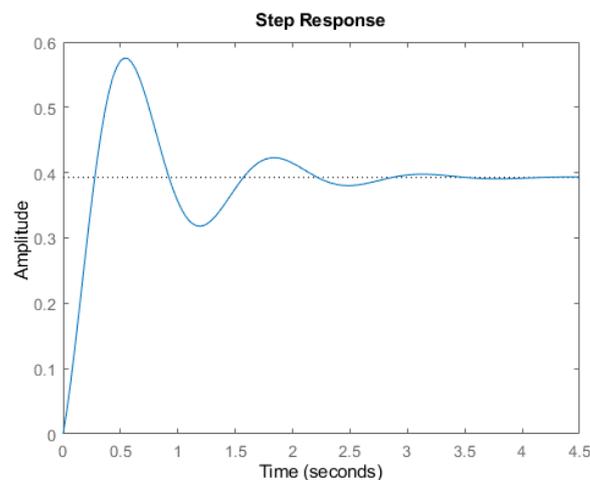
**Figure 5.16:** Step response of the continuous-time LTI transfer function.

**Discrete-time Transfer Function** The next step is to analyse the discrete-time LTI transfer function. We follow the same process by looking at its pole zero map and step response. In a pole zero map of a discrete system its stability is determined by the position of the poles and zeros compared to the unit circle. Its stable region is inside of this circle as per Figure 5.13. Computing the pole zero map of the discrete LTI transfer function the following results viewed in Figure 5.17.



**Figure 5.17:** Pole (red x) zero (red o) map of the discrete LTI transfer function.

This shows that both poles are on the unit circle. This indicates a marginally stable system. A marginally stable system is a system that is just on the border of being stable and unstable. A system that is marginally stable can still exhibit a certain amount of oscillation in its responses, but the amplitude of the oscillation will not increase over time, making it stable. The step response (Figure 5.18) of this system shows a stable response to a step input and when comparing it to the continuous-time one, it is the near equal.



**Figure 5.18:** Step response of the discrete LTI transfer function.

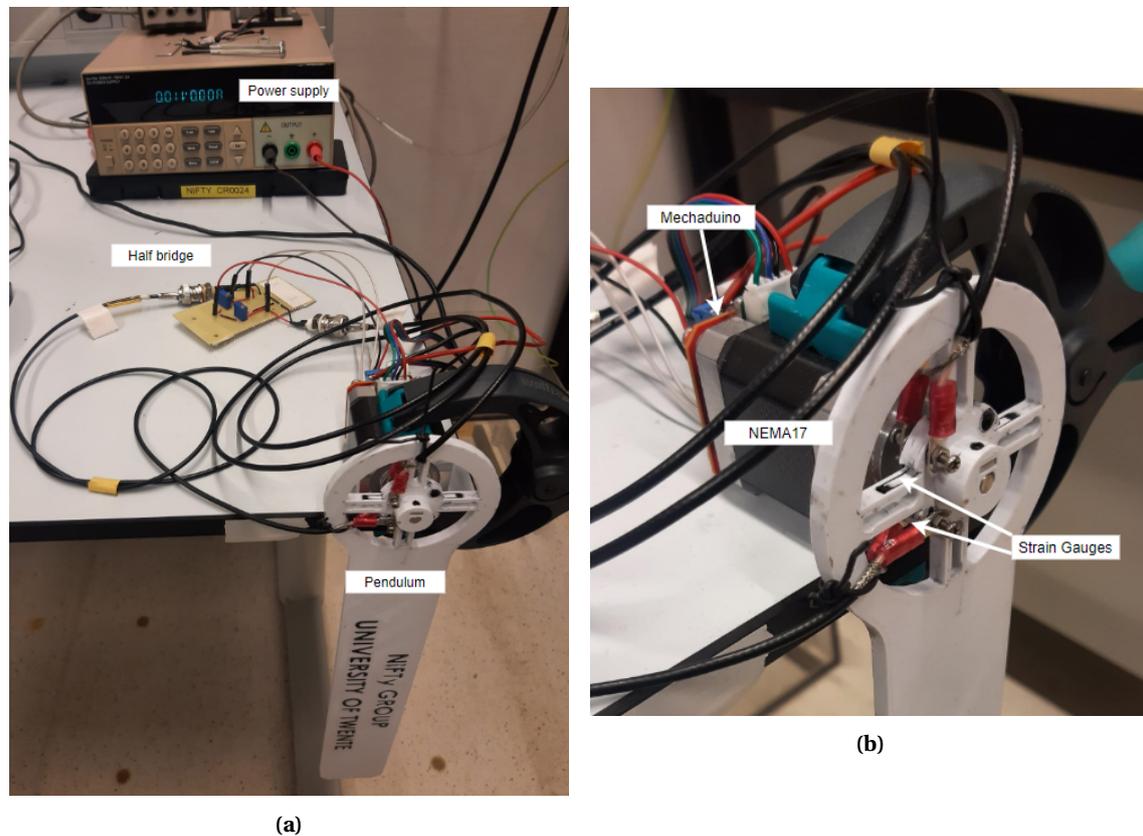
While a marginally stable system might be sufficiently stable it is not preferable.

## 5.4 Realisation

In this section the realisation part of this research is elaborated upon. The first part contains an overview of the application setup that is used (which is previously described in Chapter 3.2). Second, the Mechatronics is introduced and the code is explained. After that, the sensor output is validated with the torque that drives the motor.

### 5.4.1 Setup

The setup (as discussed in Chapter 3.2) is realised and viewed in the annotated Figure 5.19. The pendulum is secured to the shaft of the NEMA17 motor (Appendix H shows the datasheet of this motor). The driver and micro-controller are combined in the Mechaduino that is secured to the back of the motor (more on the Mechaduino in Section 5.4.2).



**Figure 5.19:** Left: Overview of the application setup. Right: Zooming in shot of the motor, strain gauges, the cable connection and the Mechaduino (red).

In this setup the gauges are numbered, the top gauge is number 1 and the bottom is number 2. For the differential gauge 2 will be subtracted from gauge 1. This means that a negative torque is applied in clockwise direction and a positive torque in counter clockwise direction.

### 5.4.2 Mechaduino & Code

**Mechaduino** The Mechaduino is an Arduino Zero based open-source control platform made by Tropical Labs [69]. It has an integrated motor driver (A4954) and micro-controller (SAM-D21G18A). The printed circuit board (PCB) is secured to the back of the stepper motor such that the on-board encoder (AS5047D 14-bit) can be read directly. The manual [70], provided by Tropical Labs, is used to set up the Mechaduino for use in this application.

There are three modes built-in for controlling the motor with the Mechaduino, position, velocity and torque mode. In this application the torque control mode is used, in which the output torque (or setpoint torque) of the motor is instructed by the user. In the dynamics description it was shown that a torque was calculated based on the desired position, this torque is used to calculate the input unit  $u$  of the motor. This controller output  $u$  is a value between 0 and 255 (8-bit value) and relates the setpoint torque  $\tau_s$  to a motor current (the unit of  $u$  is mA). This conversion for torque to  $u$  looks like

$$u = \frac{\tau_s}{0.0086K_t}. \quad (5.33)$$

Here  $K_t$  is the torque constant of the motor with

$$K_t = \frac{\tau_h}{I_p}, \quad (5.34)$$

where  $\tau_h$  is the holding torque of the motor and  $I_p$  is the peak current of the motor. Both values are present in the datasheet of the NEMA17 (Appendix H).

**Code** The structure of the code can be explained by breaking down what is to be achieved. Two distinct functions are determined.

- Positional control of the pendulum
  - Define a reference for the pendulum
  - Compute an input torque based on the reference(using the difference equations)
  - Convert this torque to motor input
- Signal processing of the sensor
  - Read sensor output (both gauges) and take a differential
  - Filter output data
  - Compute torque
  - Write values to console

The first function is looped at a sample frequency  $F_s$  of 3000 Hz. This time step is therefore used in the discretization of the transfer functions. The second function loops at a frequency of around 150 Hz. The main reason for this lower frequency is the execution time of the writing of the values to the console. These values are later imported to MATLAB for further processing and analysis.

For control of the position, only the difference equation for the PD controller was used as the implementation of dynamics cancelling controller proved more challenging than expected. This controller is sufficient for position control for a defined reference and computing an input torque to the motor. The difference equation (5.32) is implemented and a first order low-pass filter is applied to this signal to filter the measurement noise. Additionally, the low-pass filter can be used to prevent the controller from over-correcting or oscillating due to high frequency errors in the system. To find a suitable low-pass filter as a difference equation we follow the same process as described earlier in the chapter. The standard first order low-pass filter [71] in the s-domain looks like

$$F(s) = \frac{w_0}{s + w_0} \quad (5.35)$$

where  $w_0$  is the cutoff frequency set to 5Hz. Using MATLAB's `c2d` command we compute the discrete version of this transfer function and compute the difference equation, resulting in the following equation

$$y[k] = 0.0008326x[k] + 0.0008326x[k-1] + 0.9983y[k-1]. \quad (5.36)$$

Here  $y$  is the filtered output and  $x$  the raw input (and their previous values).

### 5.4.3 Validation

In order to analyze the actual behavior of the sensor a validation needs to be done. The goal of this validation is to see how the sensors behaves outside of the characterization setup. So, the calculated torque from the controller is to be compared against the measured torque by the sensor. A MATLAB script is started after initializing the Mechaduo. This script logs the output signals of the Mechaduo. Four outputs are generated for a total experiment time of 60 seconds, the calculated torque  $\tau_c$ , the raw sensor output  $V_s$ , the sensor based torque  $\tau_m$  and the elapsed time  $t$ .  $\tau_c$  is the output of the controller and is used as the input for the motor driver.  $\tau_m$  is the torque measured as a function of strain gauge voltage response. The gauges output a voltage  $V_s$ , this voltage is used to solve for the torque based on the linear fit that is generated during the characterisation of the sensor. As the anticipated torque is low ( $< 0.3\text{Nm}$ ) the linear fit of dataset 182 (Equation 5.37) is used. This fit was based on a torque of  $\pm 0.2\text{Nm}$ . Where  $y$  is the sensor output and  $x$  the measured torque ( $\tau_m$ ) and was chosen based on the fact that all fits in the  $0.2\text{ Nm}$  are very similar.

$$V_s = 0.0076\tau_m + 0.0019 \quad (5.37)$$

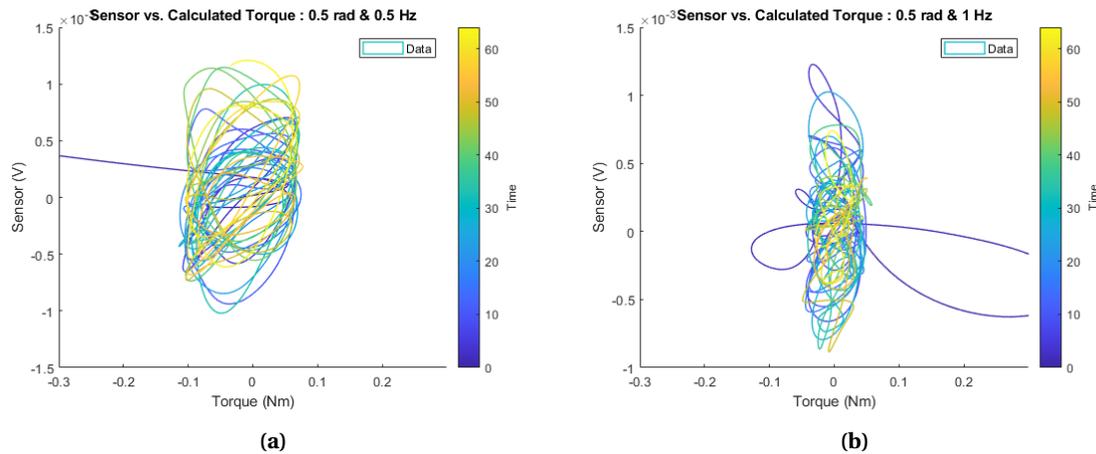
Inverting this equation yields an expression for the measured torque as a function of the sensor output (Equation 5.38).

$$\tau_m = \frac{V_s - 0.0019}{0.0076} \quad (5.38)$$

We need to be aware of the fact that there are likely external factors at play here, these factors will have an influence on the results. One is the use of a self-made half-bridge configuration instead of the high quality DEWE DAQ to measure the voltage difference of the strain gauges. Another is the addition of the motor, this can be a source of vibration.

Multiple experiments are done and the results are discussed. First, a similar comparison is made as was done in the characterization where the sensor output is plotted against the torque in time to see if the behavior of the sensor in the is the same as in the characterization setup, this includes post process filtering of the data. Next, a comparison between the calculated torque and the measures torque is made to see how the sensor follows the input torque. Then, the data is processed based on the idea that in future work the output of the sensor will be used in real time and thus no post processing is done.

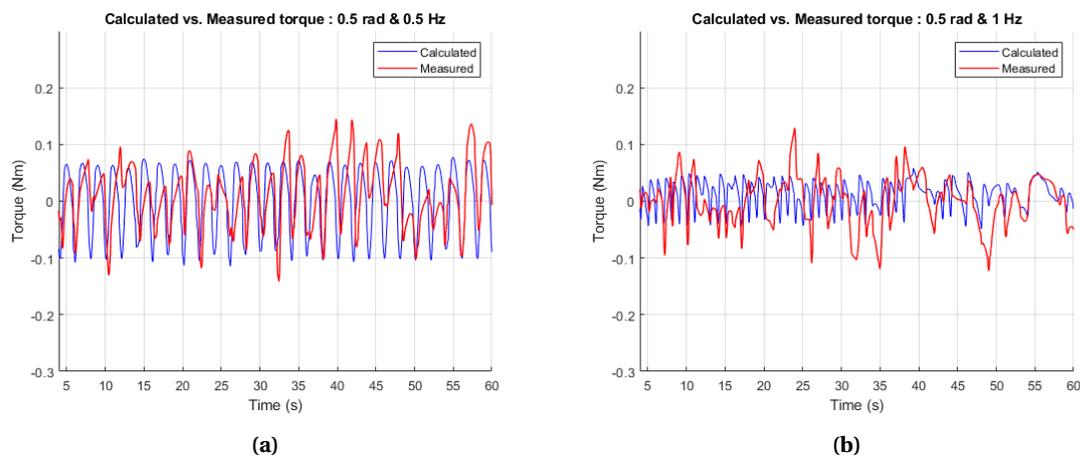
**Calculated Torque and Measured Torque** The first experiment was done with a constant sine wave excitation with an amplitude of 0.5 rad and a frequency of respectively 0.5 Hz& 1 Hz. In post processing both  $\tau_c$  and  $\tau_m$  are filtered using a 2<sup>nd</sup> order Butterworth low-pass filter with a cutoff frequency of 30 Hz to smooth the signal. First we take a look at sensor output against the calculated torque in time as was done in the characterization. Figure 5.20 shows these plots.



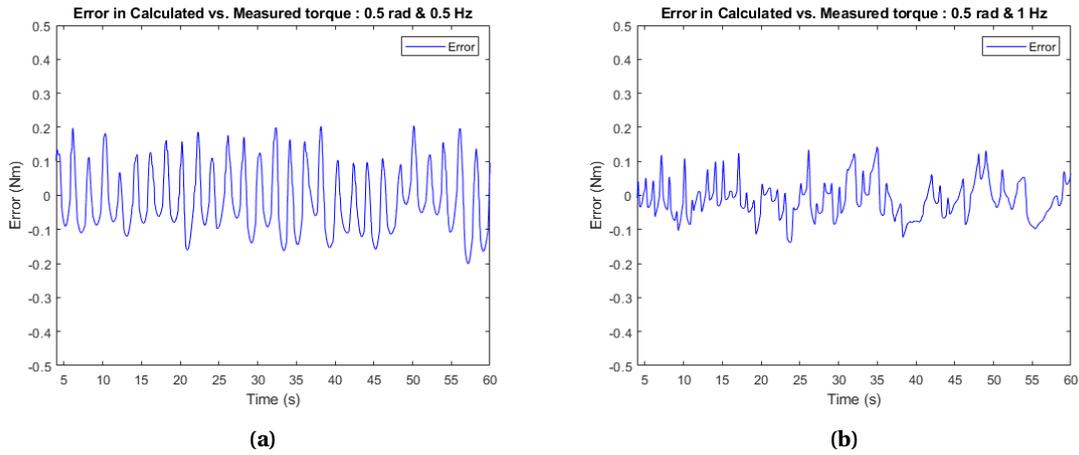
**Figure 5.20:** Sensor output plotted against the calculated torque  $\tau_c$ . (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz.

In Figure 5.20 (a) we observe a pattern over time that drifts, but does not match the form of the characterization. It also shows hysteresis behavior. Figure 5.20 (b) is more chaotic and shows no clear pattern, making it harder to interpret.

The resulting comparison between  $\tau_c$  and  $\tau_m$  is viewed in Figure 5.21 and their corresponding error plots in Figure 5.22.



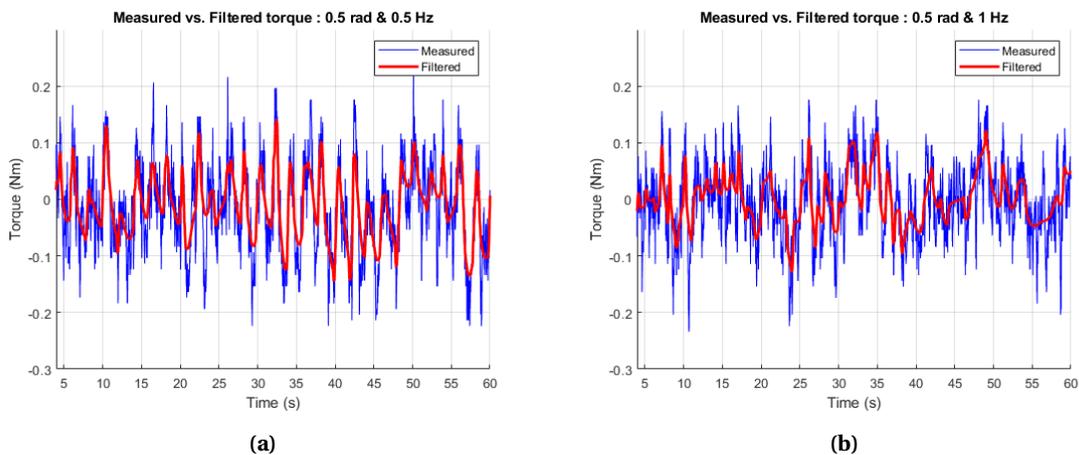
**Figure 5.21:** Comparison between  $\tau_c$  (blue) and  $\tau_m$  (red) plotted against time. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz.



**Figure 5.22:** Error plot corresponding to Figure 5.21. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz.

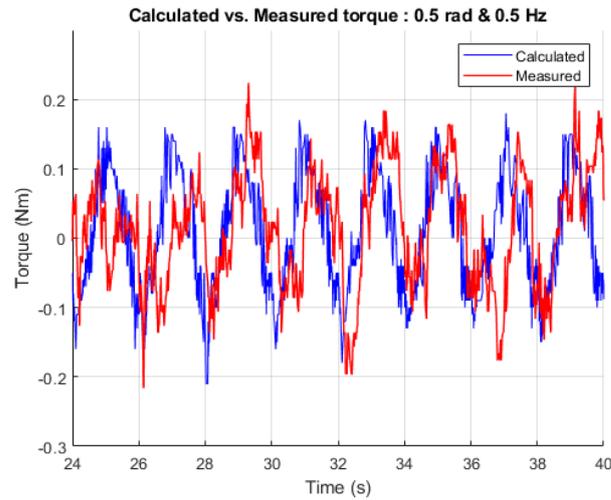
From Figure 5.21 we see that  $\tau_m$  has a more erratic behavior than the calculated  $\tau_c$ . This indicates an inconsistent signal from the sensor. We also observe that a higher frequency increases this erratic behavior considerably (as per Figure 5.21 (b)). In both cases we see a consistent pattern in the error as per Figure 5.22 (a) with errors up to twice the input and again a more chaotic pattern in the Figure 5.22 (b). Part of these errors is caused by phase shift due to the low-pass filter that is applied in the Mechaduino in real time, more on that in the next paragraph.

Note that this performance is after filtering in post processing. Figure 5.23 views the comparison between the unfiltered and filtered components of the measured torque  $\tau_m$ . It shows high frequency peaks in both measurements. These peaks are most prevalent at the points where the motor switches direction and can thus be caused by vibrations from the motor.



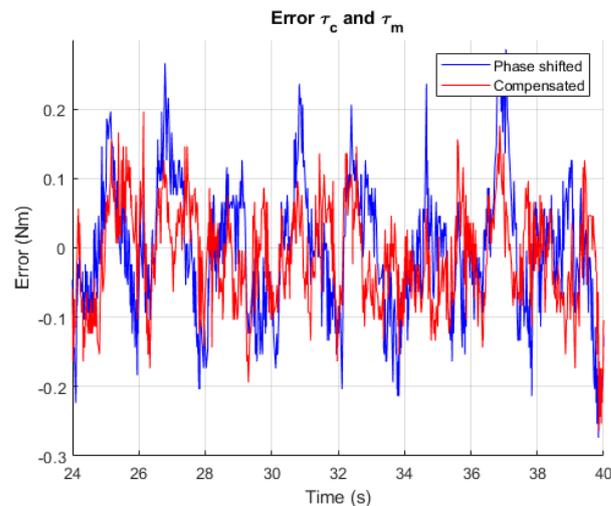
**Figure 5.23:** Comparison between unfiltered  $\tau_m$  (blue) and filtered  $\tau_m$  (red) plotted against time. (a) 0.5 rad & 0.5 Hz. (b) 0.5 rad & 1 Hz.

**No Post Processing** As mentioned, the idea for future work is to use the output of the sensor to control a system directly making post processing of the output signals not possible. This poses a challenge for real time use of the sensor output as low-pass filtering cause a phase shift (see Figure 5.24) in the signal and must be compensated for in real time.



**Figure 5.24:** Unfiltered (in post processing) comparison between  $\tau_c$  (blue) and  $\tau_m$  (red) plotted against time. A phase lag is present in the response.

To showcase the effect of the phase lag a crude phase lag compensation is done in post processing by shifting the signal. Now the error between the real output and the compensated one can be evaluated, this is shown in Figure 5.25. In this figure we see the error between  $\tau_c$  and the phase shifted  $\tau_m$  in blue and the error between  $\tau_c$  and the phase shift compensated  $\tau_m$  in red. While the compensated error is still prevalent it shows a decreased error up to 0.2 N.m.



**Figure 5.25:** Error comparison between the phase lagged  $\tau_m$  (blue) and compensated  $\tau_m$  (red).

## 5.5 Conclusion

The dynamics of the plant are described and validated in simulation. The proposed control scheme was implemented together with the desired interaction behavior. This resulted in a controlled simulation in which the pendulum reacts to external torques.

The system was then described by its continuous transfer function and later by its discrete version and their difference equation, in order for the dynamics cancelling controller and the PD controller to be implemented in the Mechaduino. The concept of linearization is elaborated upon and the method of gain scheduling is introduced. The linearized transfer functions are tested against the non-linear simulated counterpart and the response was as expected and thus describe our system correctly.

The setup is made and the sensor output is measured using a half bridge configuration bridge. Controlling the position of the sensor is done with only a discrete PD controller as the implementation of the dynamics cancelling controller deemed more challenging than expected. The fit from the characterization is used to calculate the sensor output in Nm.

Several measurements are done and discussed. It was found that reference wave frequency larger than 0.5 Hz result in an erratic response of the sensor. The response of lower frequency references is found to be more consistent and shows more potential. The main problem is the phase lag that is introduced by the real time low-pass filter. Without compensation the phase lag introduces a significant error compared to a compensated signal.

So, the main goal of realising the application is met, though not fully to the extent that was the initial plan, but sufficient to do a validation of the sensor.

## 6 Discussion & Future Work

In this chapter several aspects of the Design, Characterisation and Control are discussed. Some of the limitations of the research are brought to light and suggestions are done for future research in order to improve on this work.

### 6.1 Design

The design features a circular sensor design with four beams and two sets of strain gauges. While four beams increase the physical stability of the structure it is an arbitrary number and the effects of the additional beams has not been researched properly. Lowering the amount of beams thus might increase the stiffness of the sensor and consequently increase the sensor response to excitation.

At the moment the two sets of gauges are identical and only one is in use. There are several options as to how the second set can be utilized in a revised design. The half bridge configuration used in this work only requires one set of gauges, improving on the measurement setup a full bridge configuration can be constructed using the second set. A full bridge configuration can compensate for lead resistances of the wiring. While lead resistance might not be excessive in smaller setups, it is still present. This brings up another suggestion for improvement where the cables will need to be managed for use of the sensor for a longer period of time. Minimizing the movement of the cables minimized the risk of false measurements due to bad connections.

The last suggestions is a more extensive research in proper cable connections to strain sensor. While the screwed connection with silver epoxy works for this setup it might loosen over time due to movement, vibration and aging of the material.

### 6.2 Characterisation

It was shown that the characterization, especially in lower frequencies, yields good results. Though we should take into account that there is an accumulated error included in that process. This error is a combination of the sensor, the DEWE, the load cell and the SMAC. Any known errors in these parts is not accounted for in the post processing of the data.

The characterization was done with a constant force, frequency pattern (also the validation). It is recommended to test the sensor to a random set of input waves in future works.

### 6.3 Control

In the control section of this work the dynamics of our pendulum system are described and the systems transfer function is computed. This transfer function include several variables such as the damping coefficient  $B$  and the moment of inertia  $I_{pa}$ . Not that these variables are only an estimation based on crude calculations. There are several methods to determine these values better that include an analysis of the system response based on motor input.

The analysis of the transfer functions gave an insight in the behavior. The discrete version appeared to be marginally stable which in this case was sufficient as the equilibrium of the system was stable. Lowering the sample frequency of the system stabilizes this system as it reduces high-frequency components and thus its energy in higher frequencies.

The setup that was created to validate the sensor used a half bridge configuration such that the sensor output can be measured. This bridge has no internal filters or disturbance rejecting components and therefore is not a very reliable measurement equipment. In future work a more well designed bridge should be made in order to minimize the measurement errors (especially since the characterization is done with a far superior DAQ device). Furthermore, the

results that are logged by MATLAB has an inconsistent time step as the Mechaduino did not have a fixed timestep for sending values to the console due to this process be computationally heavy.

The results showed a phase lag due to the real time filtering in the Mechaduino. In future work this phase lag will have to be compensated for if the sensor is to be used in real time.

## 7 Conclusion

This work describes the design process, characterization and validation of an integrated 3D printed torque sensor in a pendulum that is used and controlled in an application. The following research questions were asked and answered.

- Which sensing technique is best suited for a 3D printed torque sensor?

Based on a quantitative process it was determined that the best suited technique for a 3D printed torque sensor would be the piezoresistive sensing technique based on strain gauges. This is mainly because of the readout capabilities of this type of sensor and design freedom it grants for the implementation of the strain gauges in a structure that allows for a consistent iteration and fabrication process.

- How should a 3D printed torque sensor be designed such that the non linearity of the sensor response is minimised?

Literature showed that, in order to minimize the non-linearity of strain gauges, a differential measurement needs to be done and required incorporation of two equal but opposite strain gauges on both sides of the strained structure. This, together with the beam analysis of a fixed cantilever with an pinned end (around which is to be rotated) and the analysis for optimal strain gauge placement along those beams has resulted in the design for a 1 DOF torque sensor.

- What application is suitable for characterizing, demonstrating and validating the sensing performance of the integrated 3D printed torque sensors?

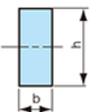
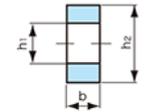
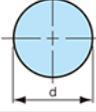
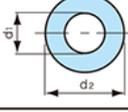
The application was decided to be a 1 DOF pendulum in which the sensor was integrated. This application provided the necessary components to characterize the sensor and validate its performance. This includes a single rotating DOF around which a torque is both measured and applied in both the characterization and validation. It also allows for freedom of control strategies in this, but also in future works.

- What does the control, needed to demonstrate such application, look like?

The control scheme proposed in this work to control the position of the pendulum includes two feedback control loops that requires positional and velocity feedback from the motor encoder. The first loop includes the dynamics cancelling controller based on the computed systems transfer function. The second loop provides error handling using a PD controller. The additional interaction layer sets an external torque threshold based on the calculated input torque. In simulation this scheme and the proposed interaction layer are validated. In the realisation phase only the error handling PD controller was implemented as the dynamics cancelling controller deemed more challenging than expected. The sensor output was validated by comparing it to the calculated input. The resulting responses after post processing showed that a low torque, low frequent harmonic input yielded the best response but still a large error was observed (up to twice the input). It became apparent that the main contributors of this error were the phase lag that was introduced by the real time low pass filter in the Mechaduino and the hysteresis behavior.

## A Section Modulus

Figure A.1 shows several equations to calculate section modulus  $Z$  from a cross section. In the case of this work the first and second equations are used.

Cross Section	Section Modulus $Z$
	$\frac{1}{6}bh^2$
	$\frac{1}{6} \cdot \frac{b(h_2^3 - h_1^3)}{h_2}$
	$\frac{\pi}{32}d^3$
	$\frac{\pi}{32} \cdot \frac{d_2^4 - d_1^4}{d_2}$

**Figure A.1:** Section modulus of various geometries. [11]

## B Pairwise Weighing

Pairwise weighing of the four aspects. Each column asks importance over the row [25].

**Table B.1:** Pairwise weighing of aspects of the sensing principles

	<b>Performance</b>	<b>Design</b>	<b>Usage</b>	<b>Mech. Dep.</b>	<b>External factors</b>			
	Linearity	Complexity	Cost	Readout	Durability	Implementation	Deformation	External factors
<b>Performance</b>	x							
Linearity	x	0	0	0.5	0	0	0	0
<b>Design</b>		x						
Complexity	1	x	0	1	0	1	0	0
Cost	1	1	x	1	0.5	1	1	0.5
<b>Usage</b>				x				
Readout	0.5	0	0	x	0	0.5	0	0
Durability	1	1	0.5	1	x	1	1	1
Implementation	1	0	0	0.5	0	x	0	0
<b>Mechanical Dependency</b>							x	
Deformation	1	1	0	1	0	1	x	0.5
External factors	1	1	0.5	1	0	1	0.5	x
<b>Rank</b>	6.5	4	1	6	0.5	5.5	2.5	2

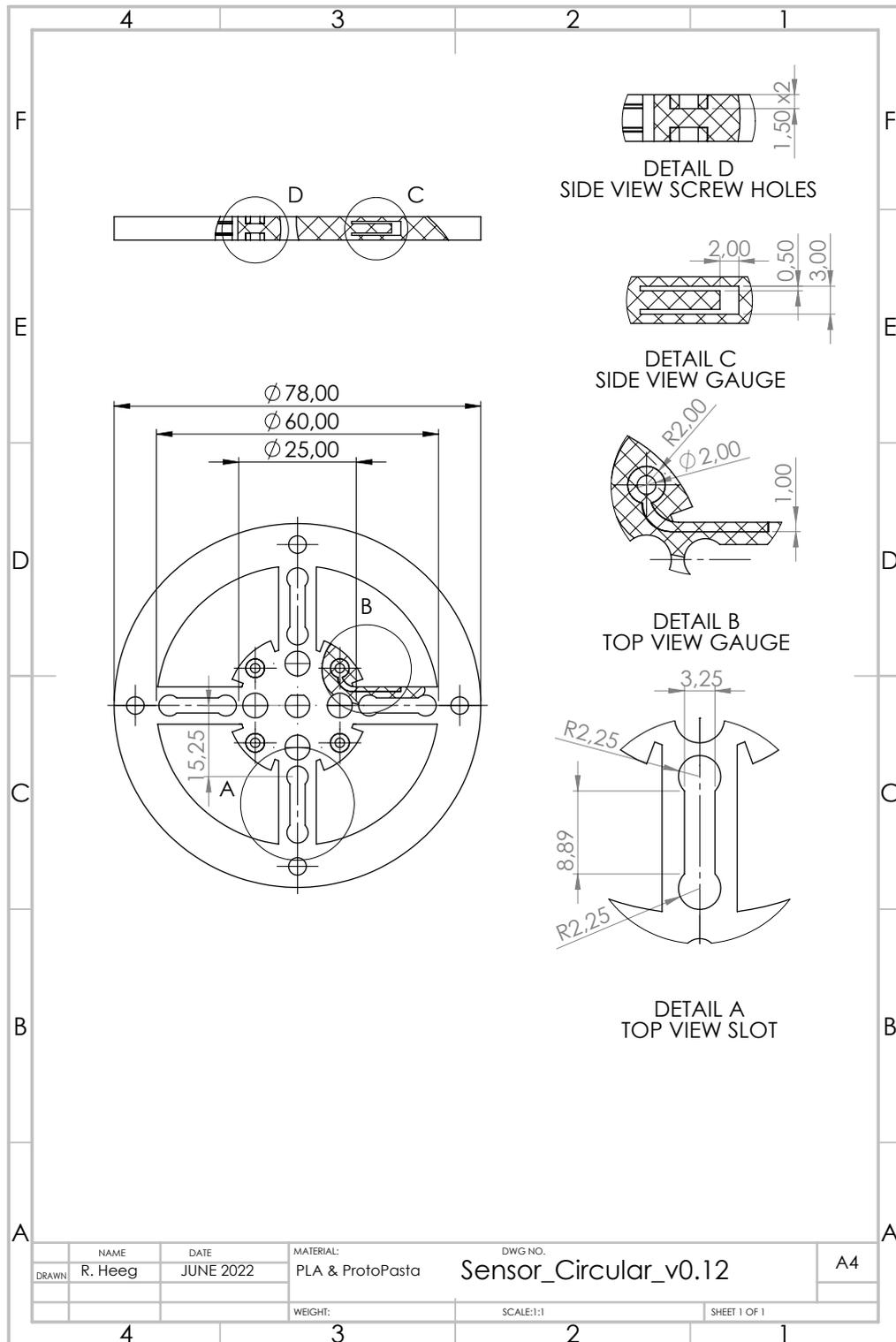
Each rank converts to a part of 100%. The total points are 28. E.g. the linearity with rank 6.5 has a final weight of  $\frac{6.5}{28} = 23\%$ .

**Table B.2:** Hierarchical weighing of factors of sensing principle. Final weight is based on the rank over a 100% total weight.

	Rank	Final Weight
<b>Performance</b>		
Linearity	6.5	23%
<b>Design</b>		
Complexity	4	14%
Cost	1	3.5%
<b>Usage</b>		
Readout	6	21.5%
Durability	0.5	1.8%
Implementation	5.5	19.5%
<b>Mechanical Dependency</b>		
Deformation	2.5	9%
External factors	2	7%

## C SolidWorks Drawings

Detailed drawing of the sensor from SolidWorks. Several cutouts are made to view internal structures of the strain gauges.



**Figure C.1:** Detailed drawing of sensor design showing internal structures as well dimensions.

Detailed drawing of the pendulum with integrated sensor from SolidWorks.

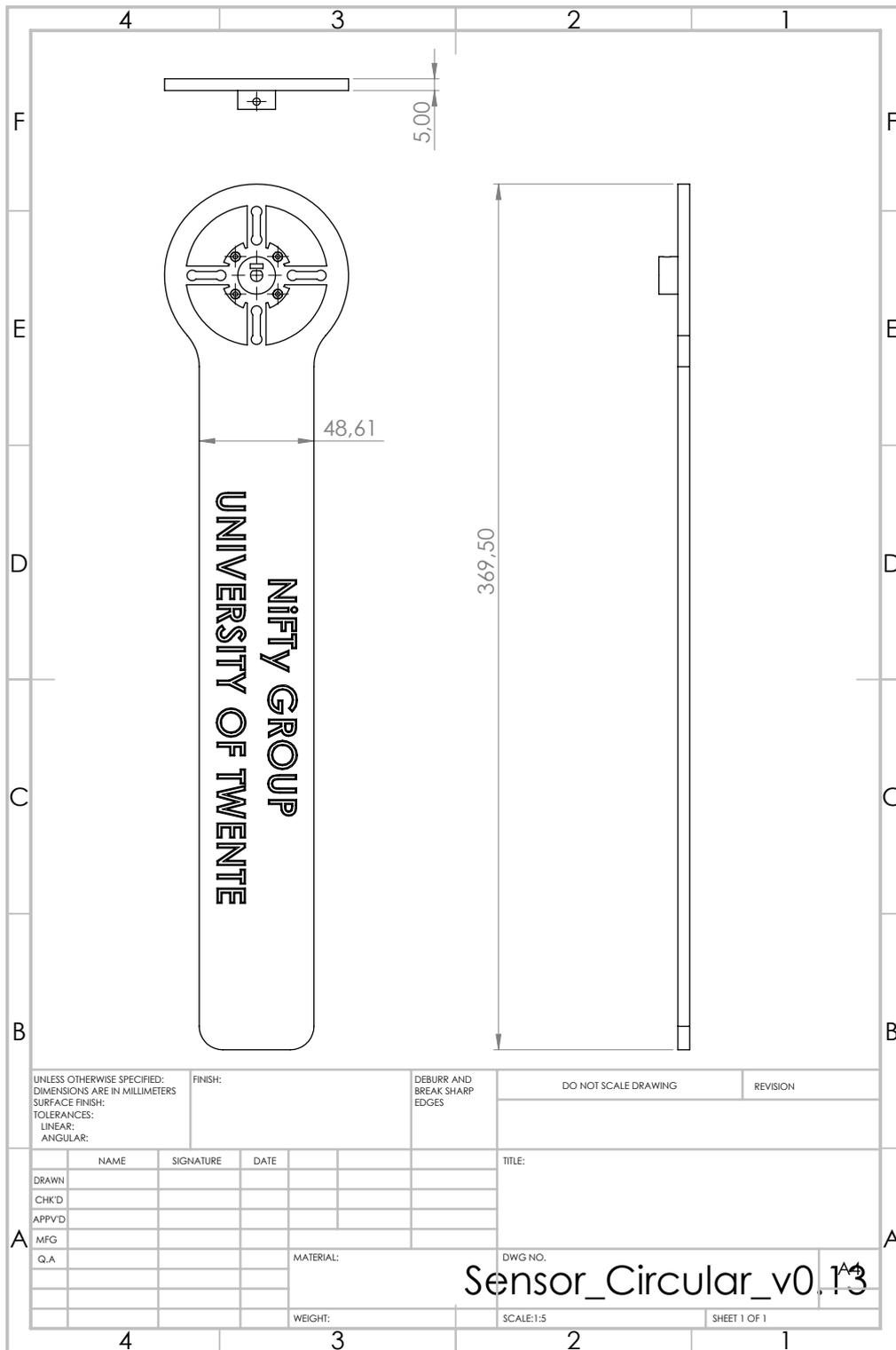


Figure C.2: Detailed drawing of sensor design showing internal structures aswell dimensions.

## D DEWE Code

```

%% Reading & Storing of measurement data from DEWE-43A
%
% Reads and stores data from DEWE-43A to .mat & .d7d files
%
% Equipment: [1] DEWE-43A : User defined amount of analog channels
%
% Author: Dimitris Kosmas & Rogier Heeg

% clear old variables
clearvars;
close all
load counter;

% DeweSoft server initialization
h = actxserver('Dewesoft.App');
h.Init();
h.Visible = true;
h.Width = 800;
h.Height = 600;

h.SetupScreen;

% Define amount of channels
amount_of_channels = 3;

% Activate channels in DEWE-43A
for i = 0:amount_of_channels-1
    h.Data.AllChannels.Item(i).Used = true;
end

% Measurement
time = 120; % [s], experiment time

% Store data
path = "C:\Users\NIFTy_PC1\Nextcloud\RaM-Internal\RaM-Subject-Groups\NIFTy\Rogier\M";
filename = string(path+"\data_"+mat2str(counter)+".d7d" );
h.StartStoring(filename);
pause(time);
h.Stop();
h.LoadFile(filename);

% Data readout
data_sections = h.LoadEngine.DataSections;
disp(sprintf('Number of data sections %d', data_sections.Count));
data_section = data_sections.Item(0);

sample_rate = h.Data.SampleRate;

```

```

disp(sprintf('Sample rate: %d', sample_rate));

% Read number of channels in Dewesoft
channel_count = h.Data.UsedChannels.Count;
disp(sprintf('Number of channels: %d', channel_count));

% Go through all channels and extract data
for i = 0:channel_count - 1
    ch = h.Data.UsedChannels.Item(i);

    % Get number of samples for channel
    sample_cnt = data_section.DataCount;

    % If there is some samples in the data file then read sample values and
    % Corresponding time stamps (also synchronous channels have timestamps)
    if sample_cnt > 0
        [out_data(:, i+1), out_time_stamp] = data_section.ReadData(ch);
        if isnan(out_time_stamp)
            t(:, i+1) = [0:1/sample_rate:(sample_cnt - 1)/sample_rate];
            data(:, i+1) = out_data(:, i+1);
        end
    end

end

end

% Plot data from channels
figure(1)
for il= 1:channel_count-1
    subplot(channel_count-1,1,il)
    plot(t(:, il), data(:, il))
end

title(sprintf('Number of channels: %d', channel_count));

% Deactivate channels in DEWE-43A
for i = 0:50
    h.Data.AllChannels.Item(i).Used = false;
end

% clear actxserver object
h = 0;

% Save data : t, data
% Only save one column of t
t = t(:,1);
save(['data\raw\DEWE\ ', 'data_', mat2str(counter), '.mat'], 't', 'data')

% Increment counter
counter = counter +1;
save('counter.mat', 'counter');

```

## E MATLAB Code

```

%% Plot/Save all data
close all;
clear all;

V0 = 10; % V in
Rbr1 = 7450; % R bridge 1
Rbr2 = 7040; % R bridge 2
data_sets = cell(1, 30);

% Data sets and corresponding N & Hz
% format = [N Hz xlim counter1 counter2 .. counterN]
data_sets{1} = [2 0.2 0.5 182 183 184];% counters for 2N 0.2Hz
data_sets{2} = [2 0.5 0.5 185 187 188];% counters for 2N 0.5Hz
data_sets{3} = [2 1.0 0.5 189 190 191];% counters for 2N 1.0Hz
data_sets{4} = [2 2.0 0.5 192 193 194];% counters for 2N 2.0Hz
data_sets{5} = [2 4.0 0.5 195 196 197];% counters for 2N 4.0Hz
data_sets{6} = [2 8.0 0.5 198 199 200];% counters for 2N 8.0Hz
%
data_sets{7} = [4 0.2 0.7 204 205 206];% counters for 4N 0.2Hz
data_sets{8} = [4 0.5 0.7 207 208 209];% counters for 4N 0.5Hz
data_sets{9} = [4 1.0 0.7 210 211 213];% counters for 4N 1.0Hz
data_sets{10} = [4 2.0 0.7 212 214 215];% counters for 4N 2.0Hz
data_sets{11} = [4 4.0 0.7 216 217 218];% counters for 4N 4.0Hz
data_sets{12} = [4 8.0 0.7 219 220 221];% counters for 4N 8.0Hz
%
data_sets{13} = [8 0.2 1.2 169 170];% counters for 8N 0.2Hz
data_sets{14} = [8 0.5 1.2 171 172];% counters for 8N 0.5Hz
data_sets{15} = [8 1.0 1.2 173 174];% counters for 8N 1.0Hz
data_sets{16} = [8 2.0 1.2 175 176];% counters for 8N 2.0Hz
data_sets{17} = [8 4.0 1.2 177 178];% counters for 8N 4.0Hz
data_sets{18} = [8 8.0 1.2 179 180];% counters for 8N 8.0Hz
%
data_sets{19} = [10 0.2 2 226 227 228];% counters for 10N 0.2Hz
data_sets{20} = [10 0.5 2 229 230 231];% counters for 10N 0.5Hz
data_sets{21} = [10 1.0 2 233 234 235];% counters for 10N 1.0Hz
data_sets{22} = [10 2.0 2 236 237 238];% counters for 10N 2.0Hz
data_sets{23} = [10 4.0 2 239 240 241];% counters for 10N 4.0Hz
data_sets{24} = [10 8.0 2 242 243 244];% counters for 10N 8.0Hz
%
data_sets{25} = [12 0.2 2 245 246 247];% counters for 12N 0.2Hz
data_sets{26} = [12 0.5 2 248 249 250];% counters for 12N 0.5Hz
data_sets{27} = [12 1.0 2 251 252 253];% counters for 12N 1.0Hz
data_sets{28} = [12 2.0 2 254 255 256];% counters for 12N 2.0Hz
data_sets{29} = [12 4.0 2 257 258 259];% counters for 12N 4.0Hz
data_sets{30} = [12 8.0 2 260 261 262];% counters for 12N 8.0Hz

%% Loop through specified data sets : i = start : end
for i = 7 : 30

```

```

N = data_sets{i}(1);
Hz = data_sets{i}(2);
% Save path for .png figures
save_path = %path
for j=4 : length(data_sets{i}) % Loop through all available data sets

dewedat = load(%path);

%% Filtering

% fsampling = 20000; % Hz
dt = (dewedat.t(end)-dewedat.t(1))/length(dewedat.t);
fs = 1/dt;
fcut = Hz*10; % Hz, < fnyq
fcut_norm = fcut/fs/2;
n=2;
[b,a] = butter(n,fcut_norm);
Vsens1 = filtfilt(b,a,double(dewedat.data(:,1))); % zero-phase filtering
Vsens2 = filtfilt(b,a,double(dewedat.data(:,2))); % zero-phase filtering
Floadcell = filtfilt(b,a,double(dewedat.data(:,3)));

%% Relative resistance

% This is correct
Vm1 = Vsens1;
Rs1 = (-Vm1*Rbr1)./(Vm1-V0);
dRs1 = sum(Rs1)/length(Rs1);
dRR1 = (Rs1 - dRs1)/dRs1;

Vm2 = Vsens2;
Rs2 = (-Vm2*Rbr2)./(Vm2-V0);
dRs2 = sum(Rs2)/length(Rs2);
dRR2 = (Rs2 - dRs2)/dRs2;

%% Loadcell to Torque

arm = 0.125; % cm
sensitivity = 0.04522; % Load cell sensitivity [mV/V/N] @wiki-calibration
gain = 456; % Load cell gain @wiki-calibration

fcell_ard = Floadcell; % filtered loadcell data
fcell_ard_uf = nonzeros(dewedat.data(:,3)); % unfiltered loadcell data

% Loadcell V conversion to F
idle_timestamp = 1; % Used for mean() reading
idx1 = find(dewedat.t >= idle_timestamp,1);

offsetCellResponse = mean(fcell_ard(1:idx1)); % use the idle time to compute the offset
offsetCellResponse_uf = mean(fcell_ard_uf(1:idx1)); % use the idle time to compute the

```

```

force_ard_nooffset = ((fcell_ard_offsetCellResponse)*1000)/(sensitivity*5*gain); % V
force_ard_nooffset_uf = ((fcell_ard_uf_offsetCellResponse_uf)*1000)/(sensitivity*5*gain); % V

torque = force_ard_nooffset*arm;

%% Plot voltage v torque
fig1 = figure('Position', [100 100 700 500]);
hold on;
grid on;
fancy_color_plot((torque)', (Vsens1-Vsens2)', dewedat.t', 'Time');
fit = polyfit(torque, Vsens1-Vsens2, 1);
f1 = polyval(fit, torque);
lh = plot(torque, f1);
lh.Color = 'r';
lh.LineStyle = '-.';
lh.LineWidth = 1;
hold off;
xlim([-data_sets{i}(3) data_sets{i}(3)]);
title(sprintf('%dN sinewave : %1.1fHz : 120s : Dataset %d', N, Hz, data_sets{i}(j)));
leg = legend('Data', 'Fit');
leg.Location = 'northwest';
xlabel('Torque (Nm)')
ylabel('Voltage (V)')
fig1name = sprintf('%dN-%1.1fHz-%d.png', N, Hz, data_sets{i}(j));
fullfile1 = fullfile(save_path, fig1name);
saveas(fig1, fullfile1);
close all;
%% Plot relative resistance v torque
fig2 = figure('Position', [100 100 700 500]);
hold on;
grid on;
fancy_color_plot((torque)', (dRR1-dRR2)', dewedat.t', 'Time');
fit = polyfit(torque, dRR1-dRR2, 1);
f1 = polyval(fit, torque);
lh = plot(torque, f1);
lh.Color = 'r';
lh.LineStyle = '-.';
lh.LineWidth = 1;
hold off;
xlim([-data_sets{i}(3) data_sets{i}(3)]);
title(sprintf('%dN sinewave : %1.1fHz : 120s : Dataset %d', N, Hz, data_sets{i}(j)));
leg = legend('Data', 'Fit');
leg.Location = 'northwest';
xlabel('Torque (Nm)')
ylabel('\Delta R/R')
fig2name = sprintf('%dN-%1.1fHz-%d-R.png', N, Hz, data_sets{i}(j));
fullfile2 = fullfile(save_path, fig2name);
saveas(fig2, fullfile2);
close all;
end
end

```

## F Measurement Results

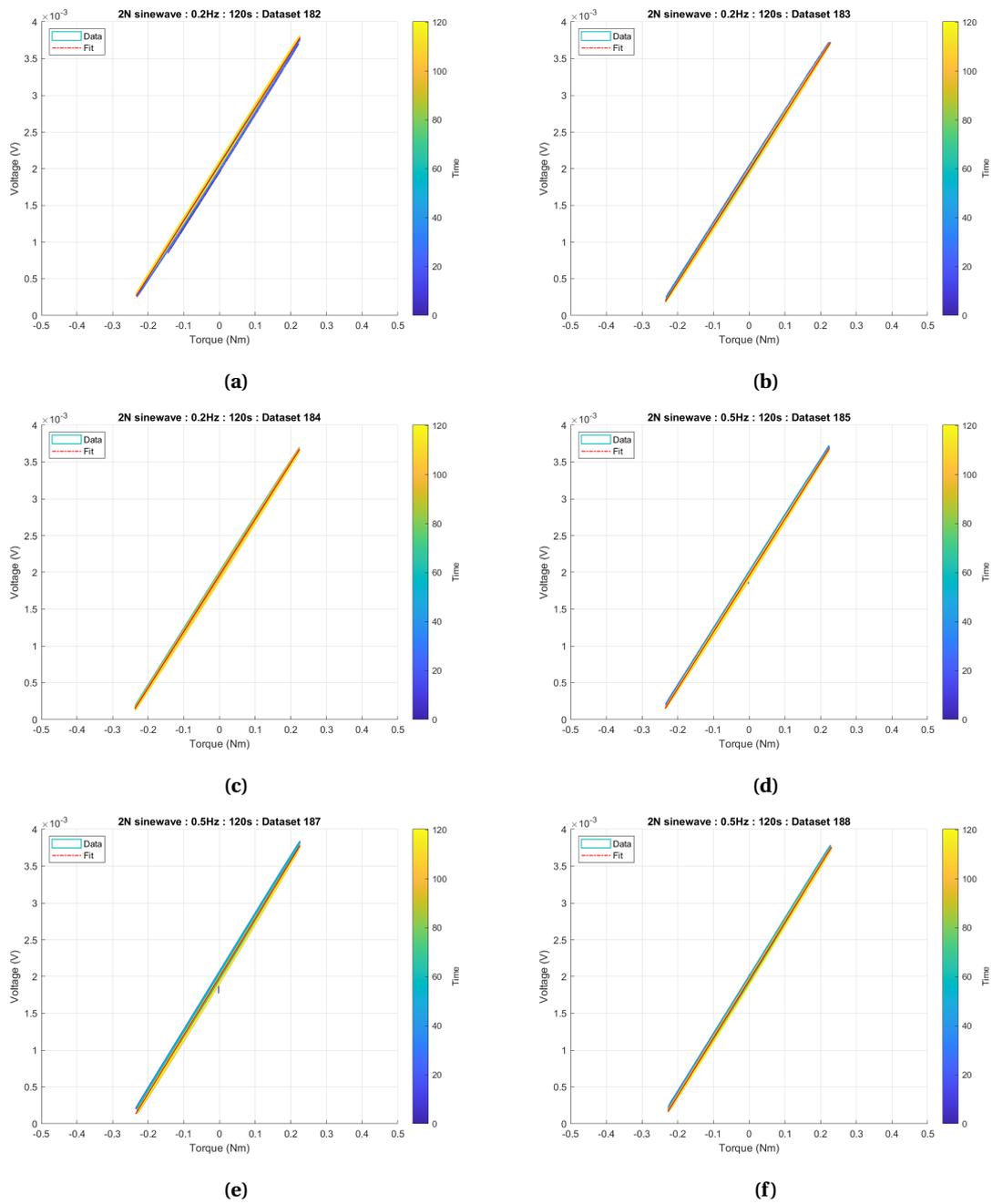
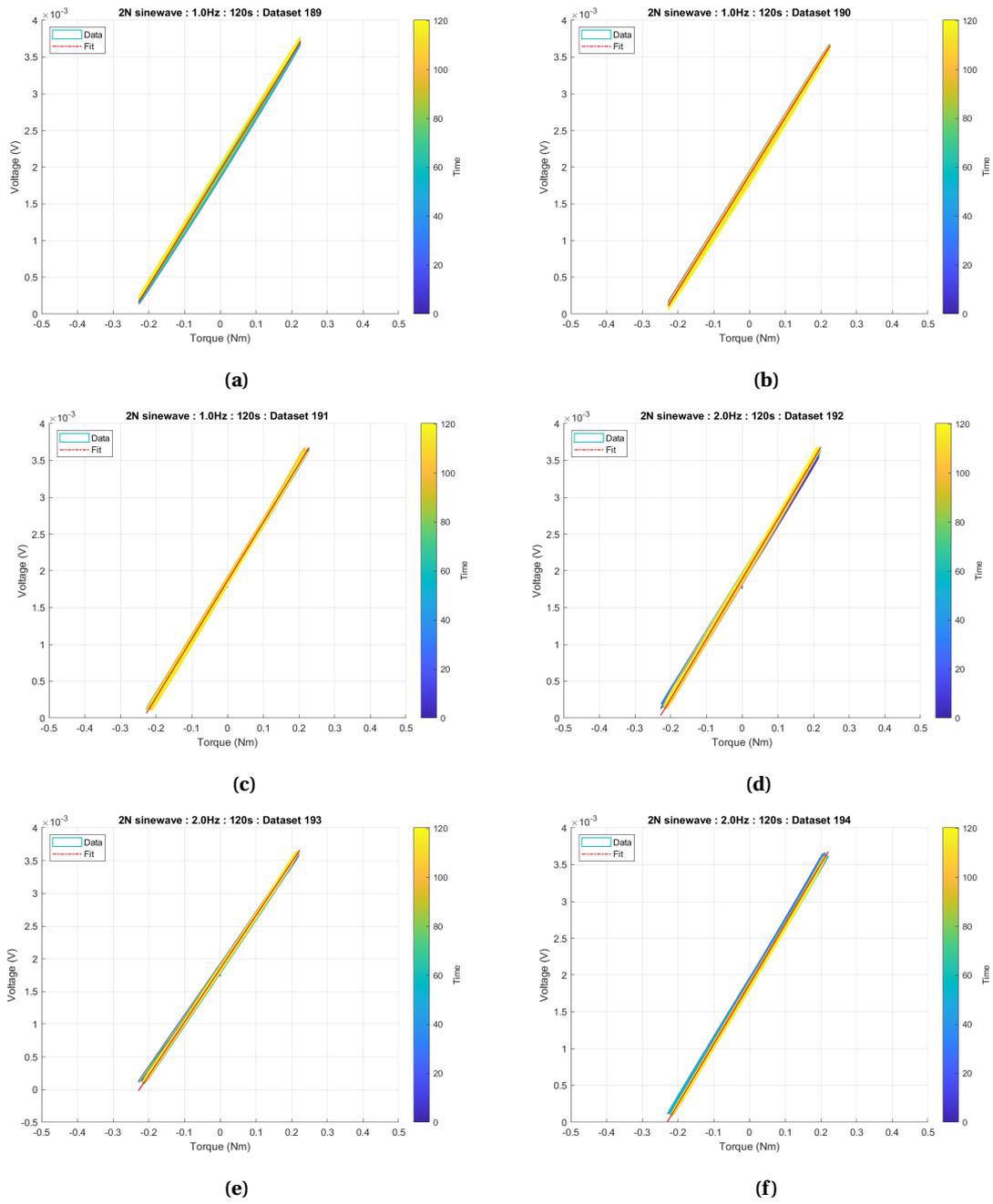
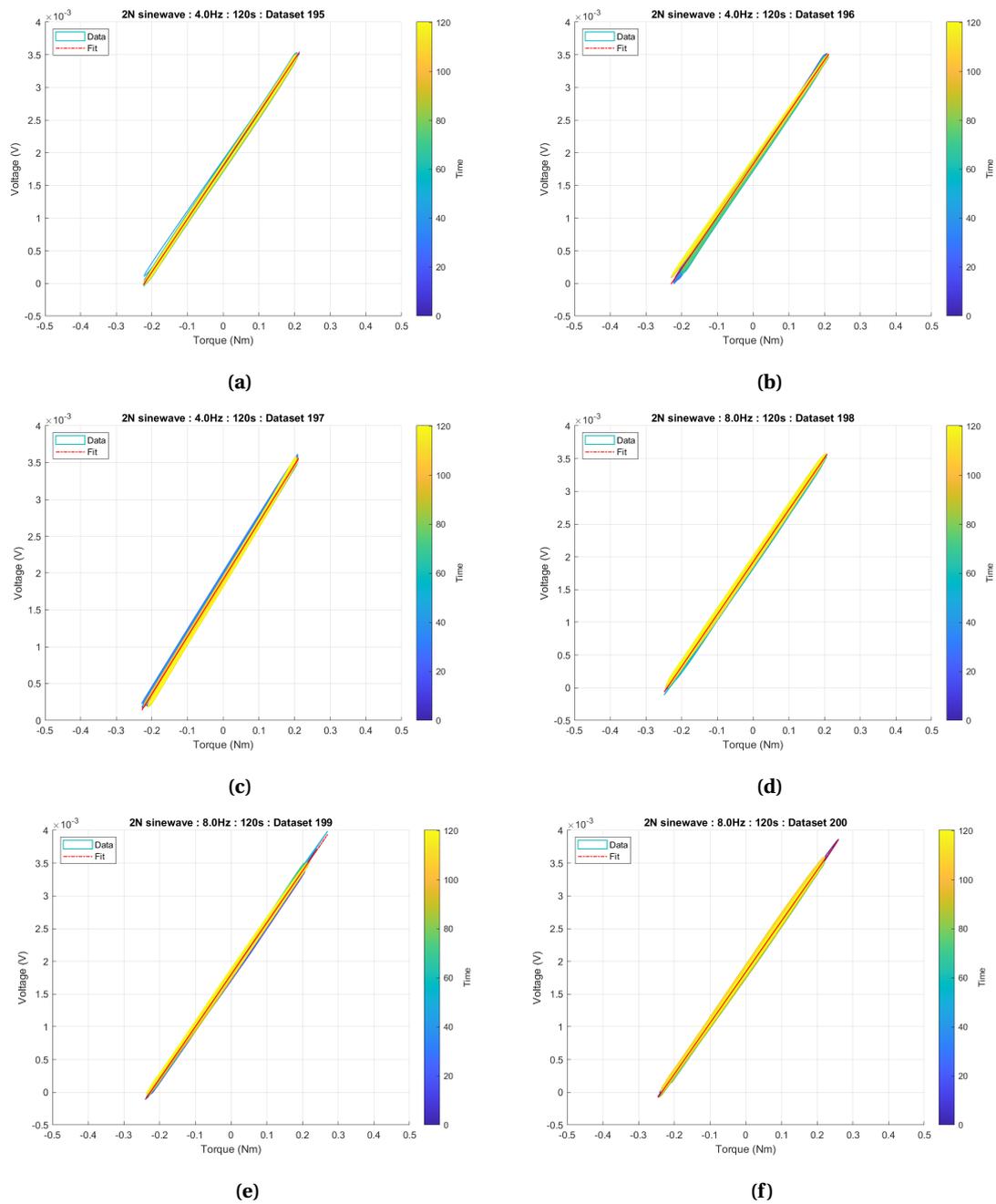


Figure F.1: Measurement results 2N sine wave, 0.2 & 0.5 Hz.



**Figure F2:** Measurement results 2N sine wave, 1 & 2 Hz.



**Figure E.3:** Measurement results 2N sine wave, 4 & 8 Hz.

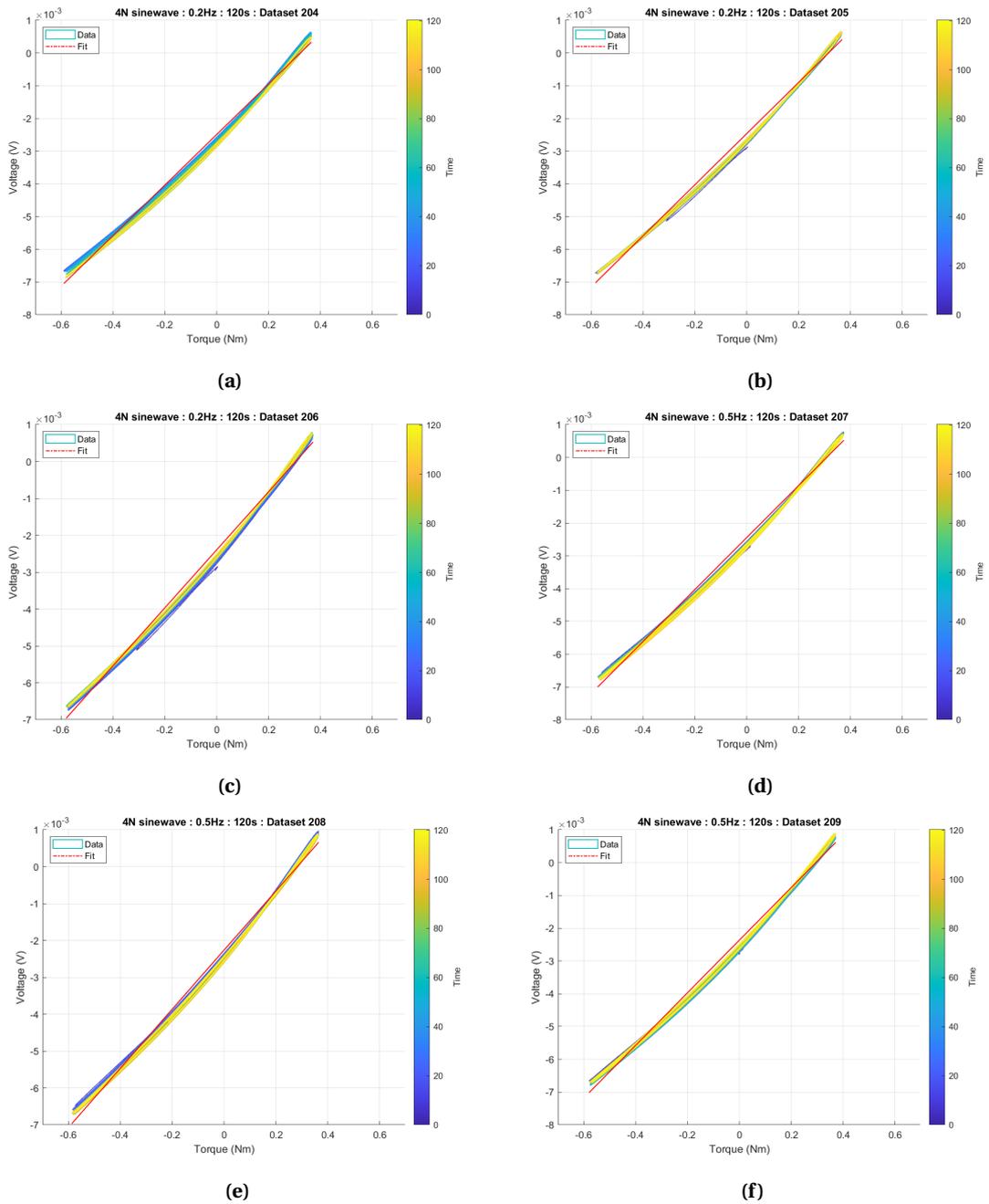
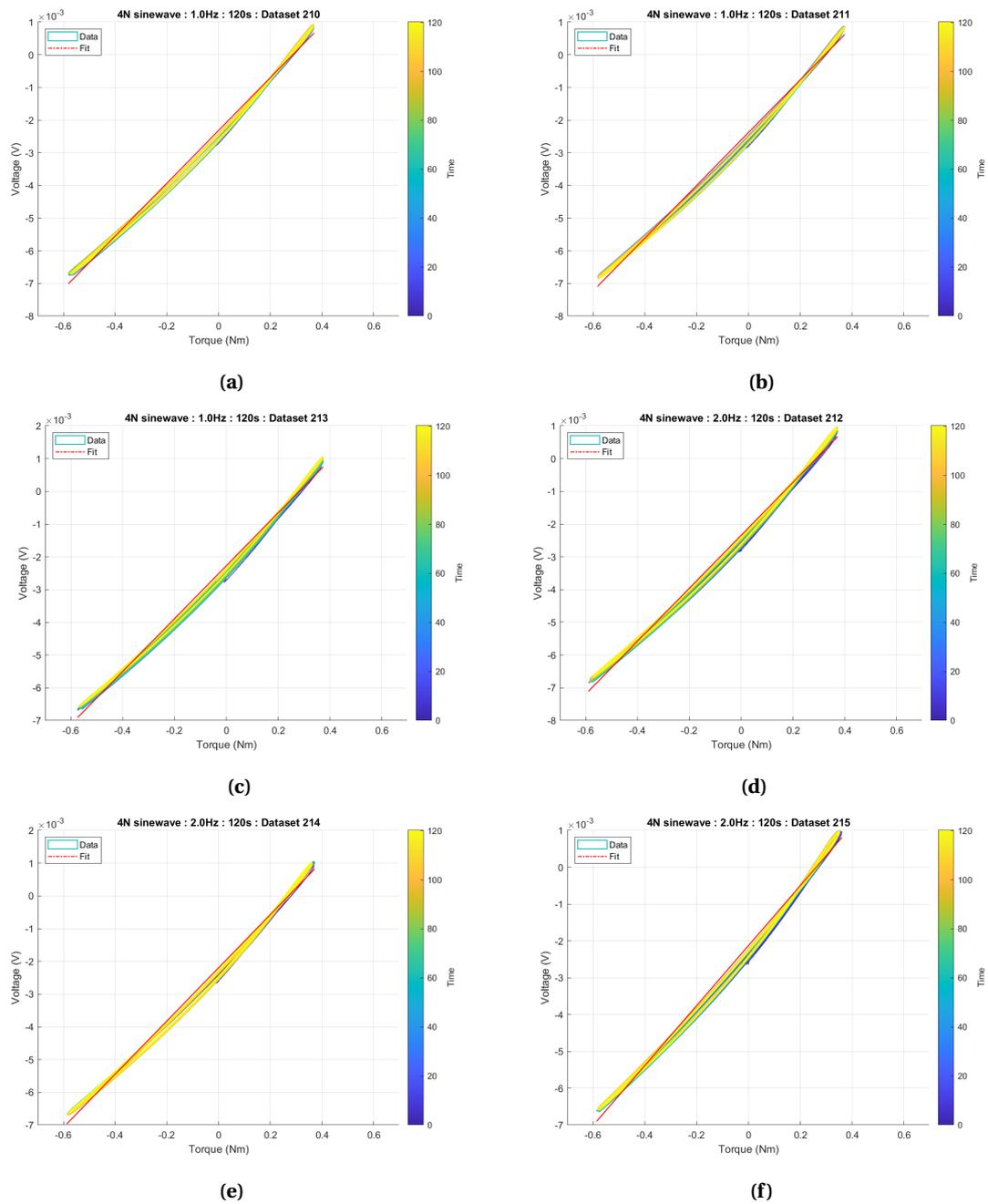
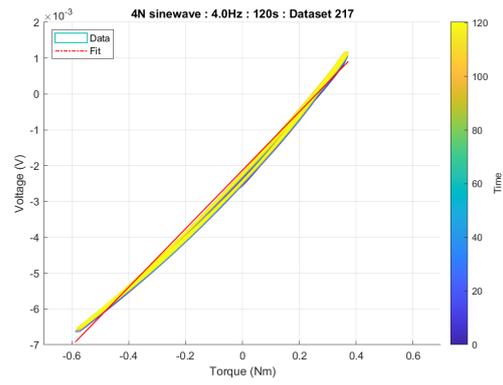
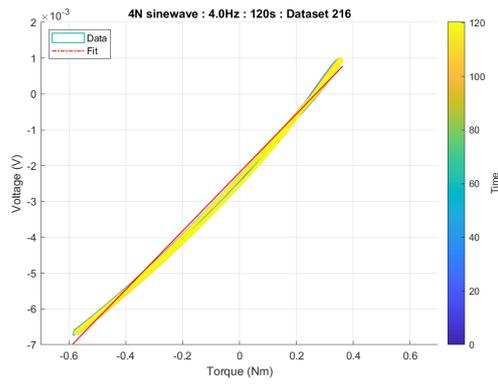


Figure F.4: Measurement results 4N sine wave, 0.2 & 0.5 Hz.

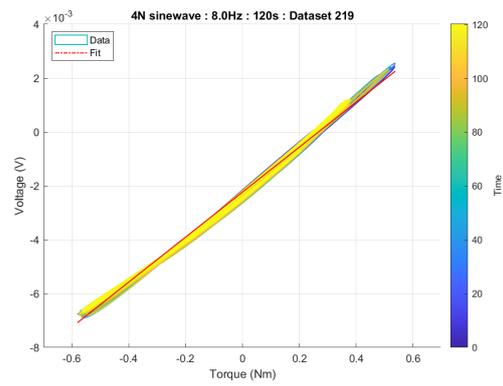
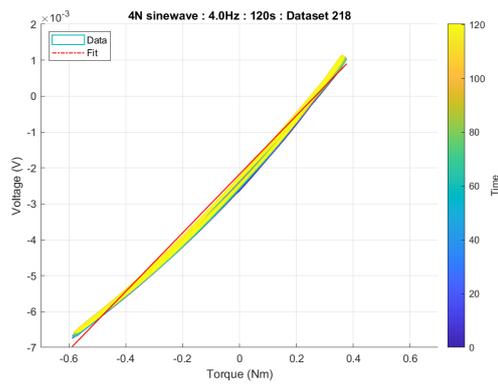


**Figure E.5:** Measurement results 4N sine wave, 1 & 2 Hz.



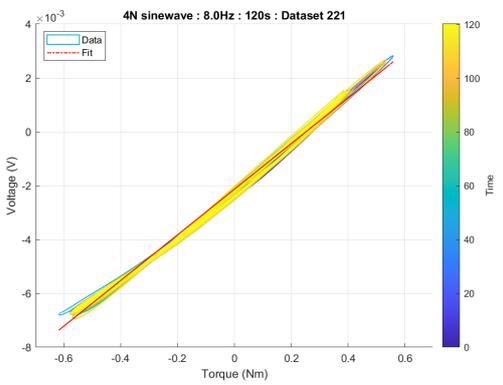
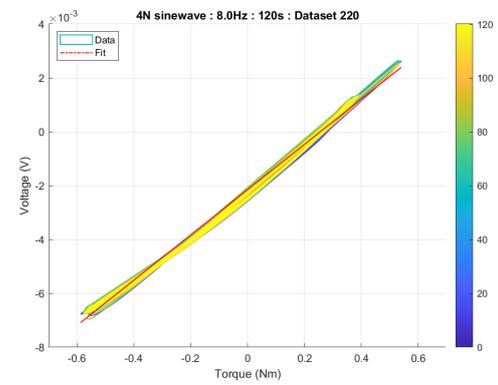
(a)

(b)



(c)

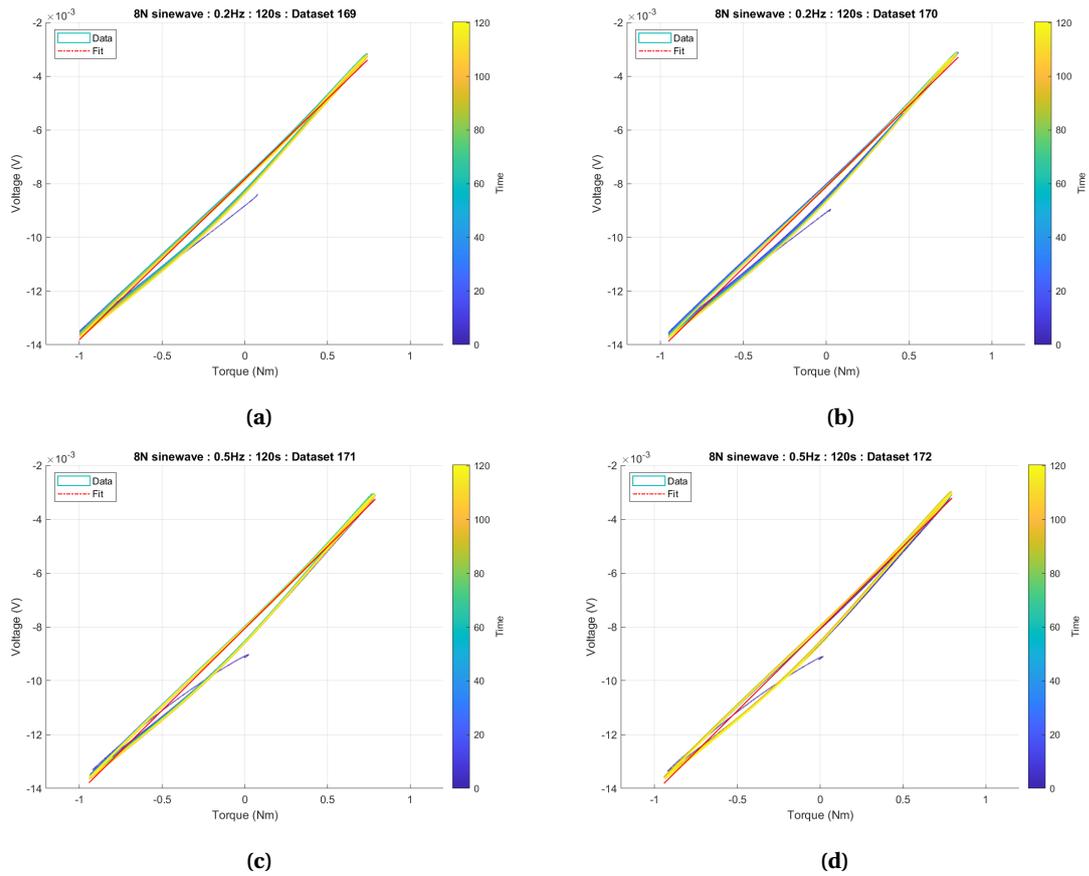
(d)



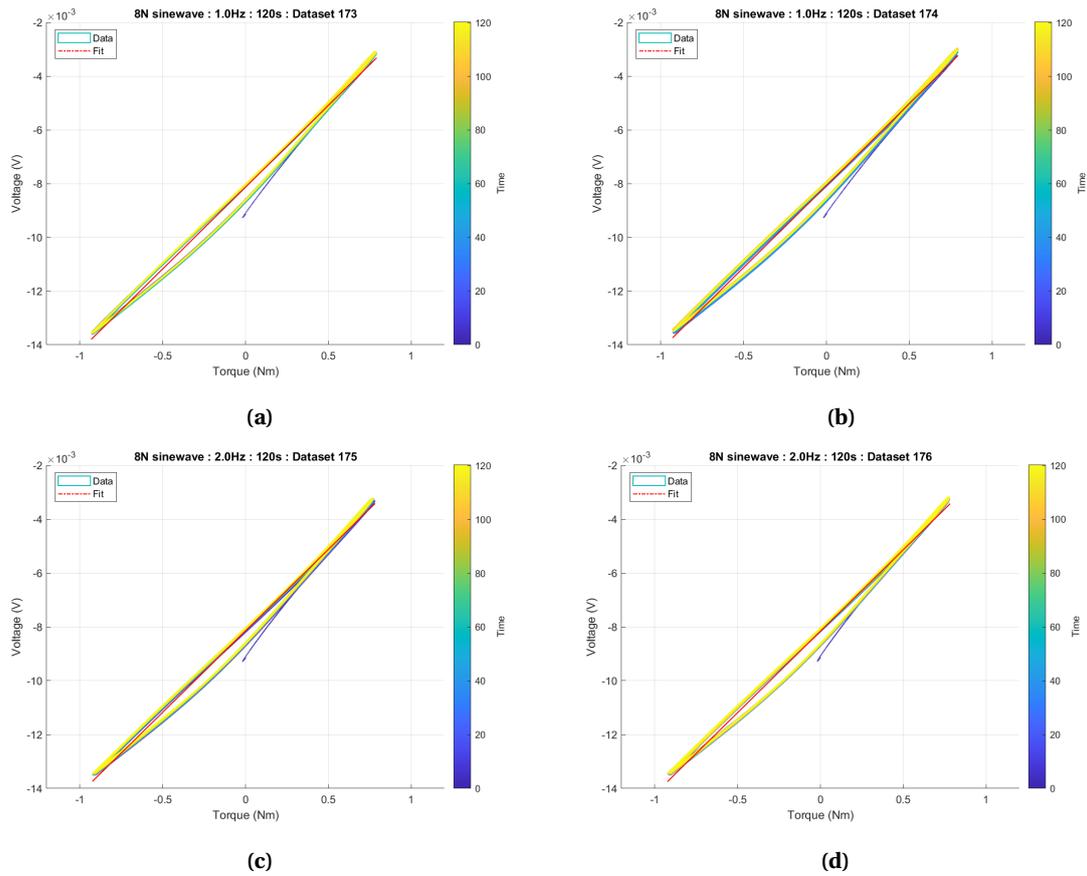
(e)

(f)

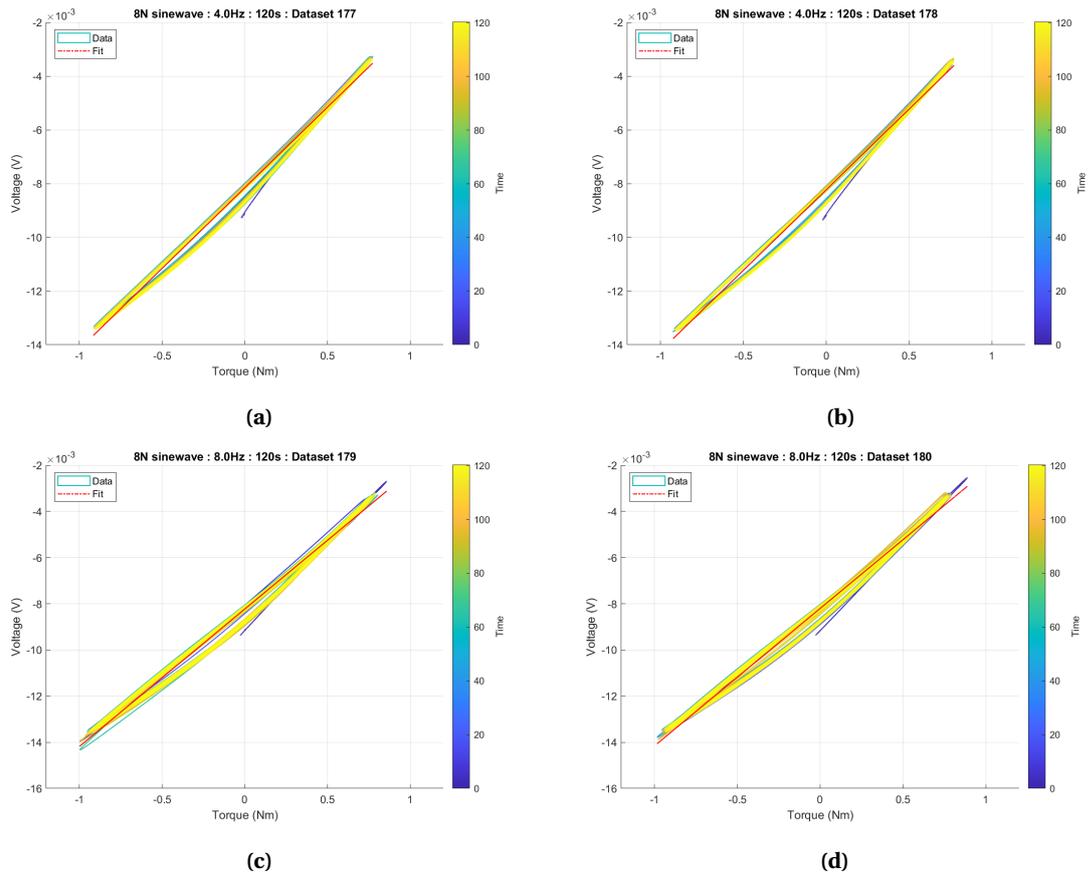
**Figure F6:** Measurement results 4N sine wave, 4 & 8 Hz.



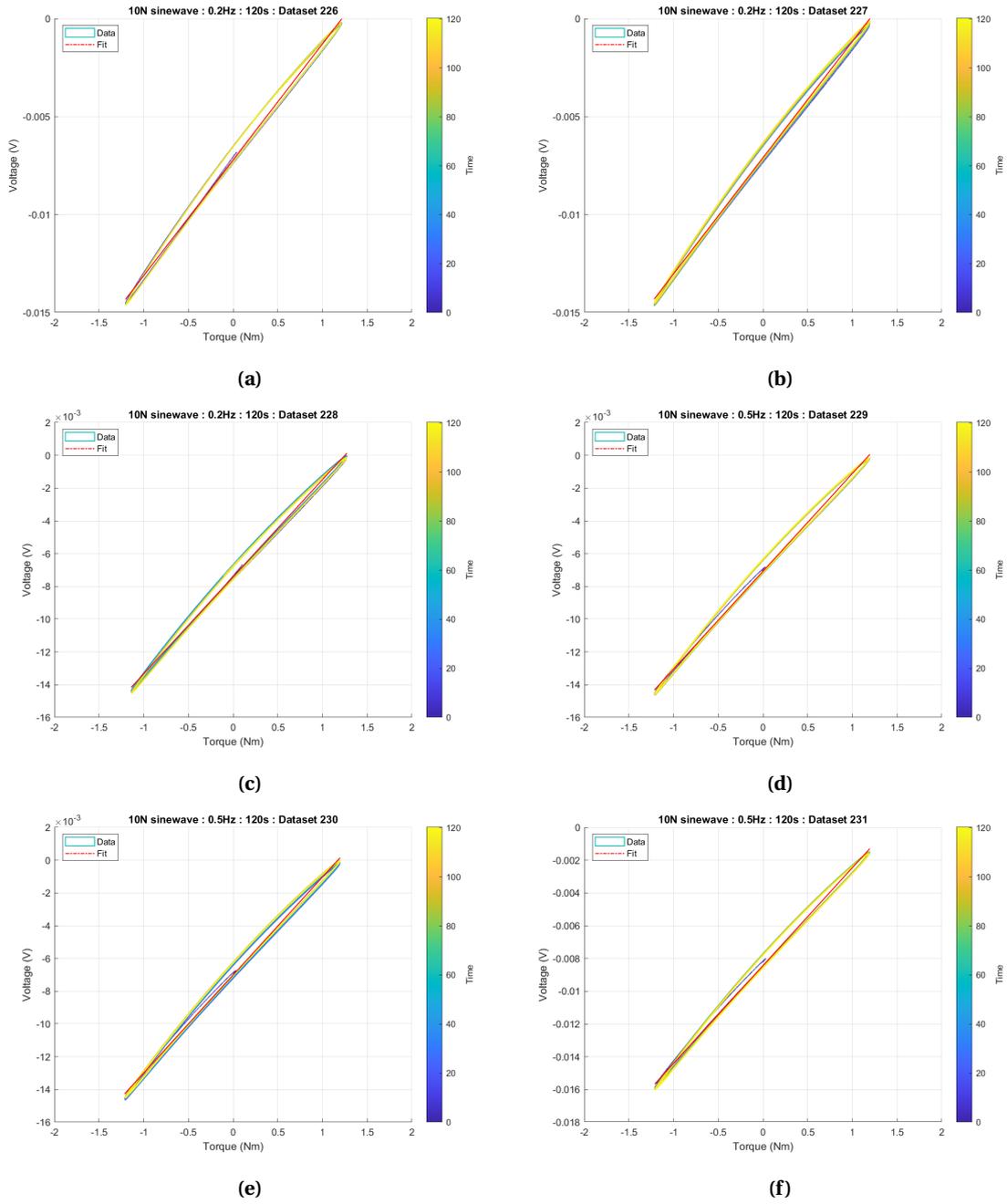
**Figure E.7:** Measurement results 8N sine wave, 0.2 & 0.5 Hz.



**Figure F.8:** Measurement results 8N sine wave, 1 & 2 Hz.



**Figure E.9:** Measurement results 8N sine wave, 4 & 8 Hz.



**Figure F.10:** Measurement results 10N sine wave, 0.2 & 0.5 Hz.

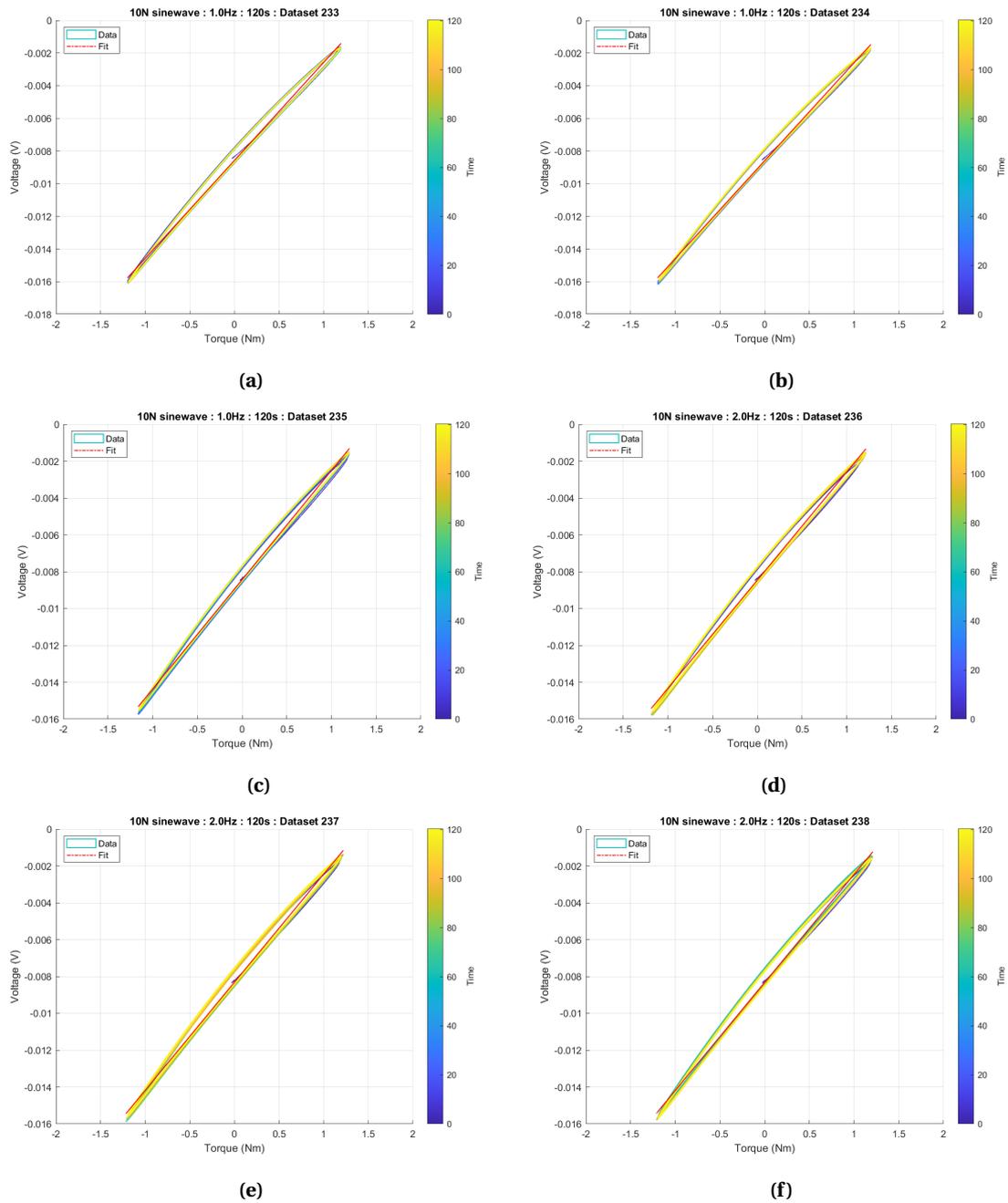


Figure F.11: Measurement results 10N sine wave, 1 & 2 Hz.

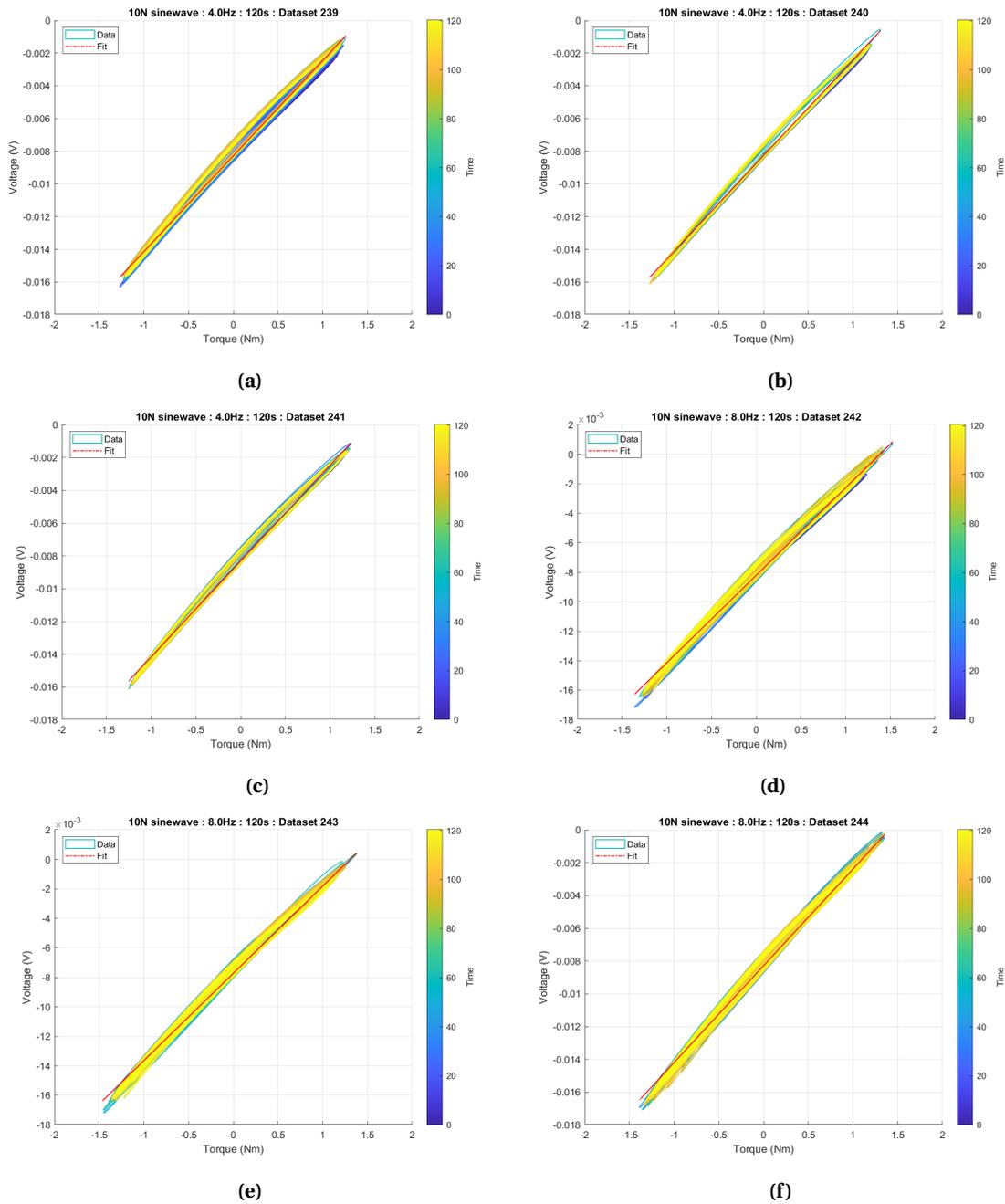


Figure F.12: Measurement results 10N sine wave, 4 & 8 Hz.

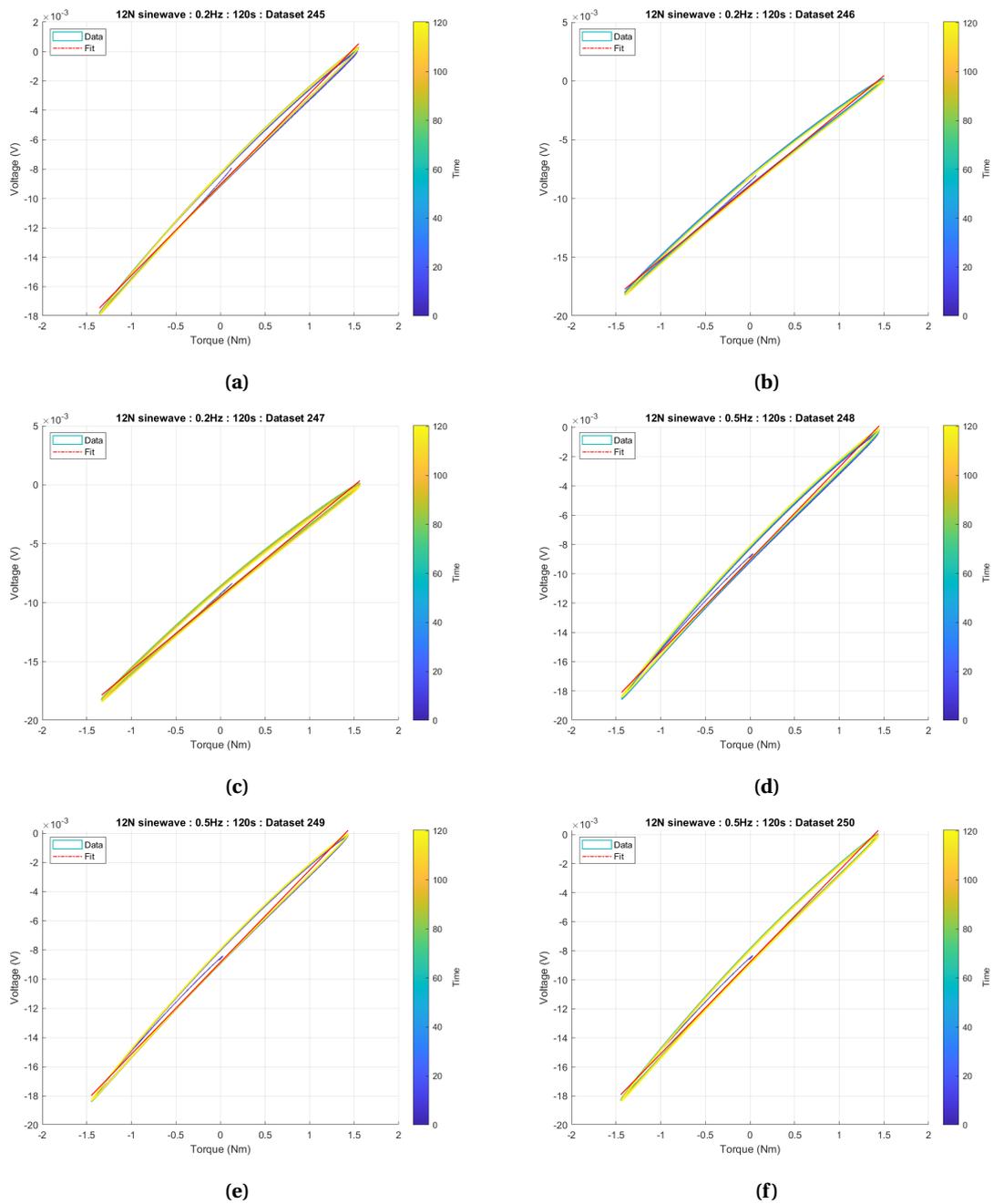


Figure F.13: Measurement results 12N sine wave, 0.2 & 0.5 Hz.

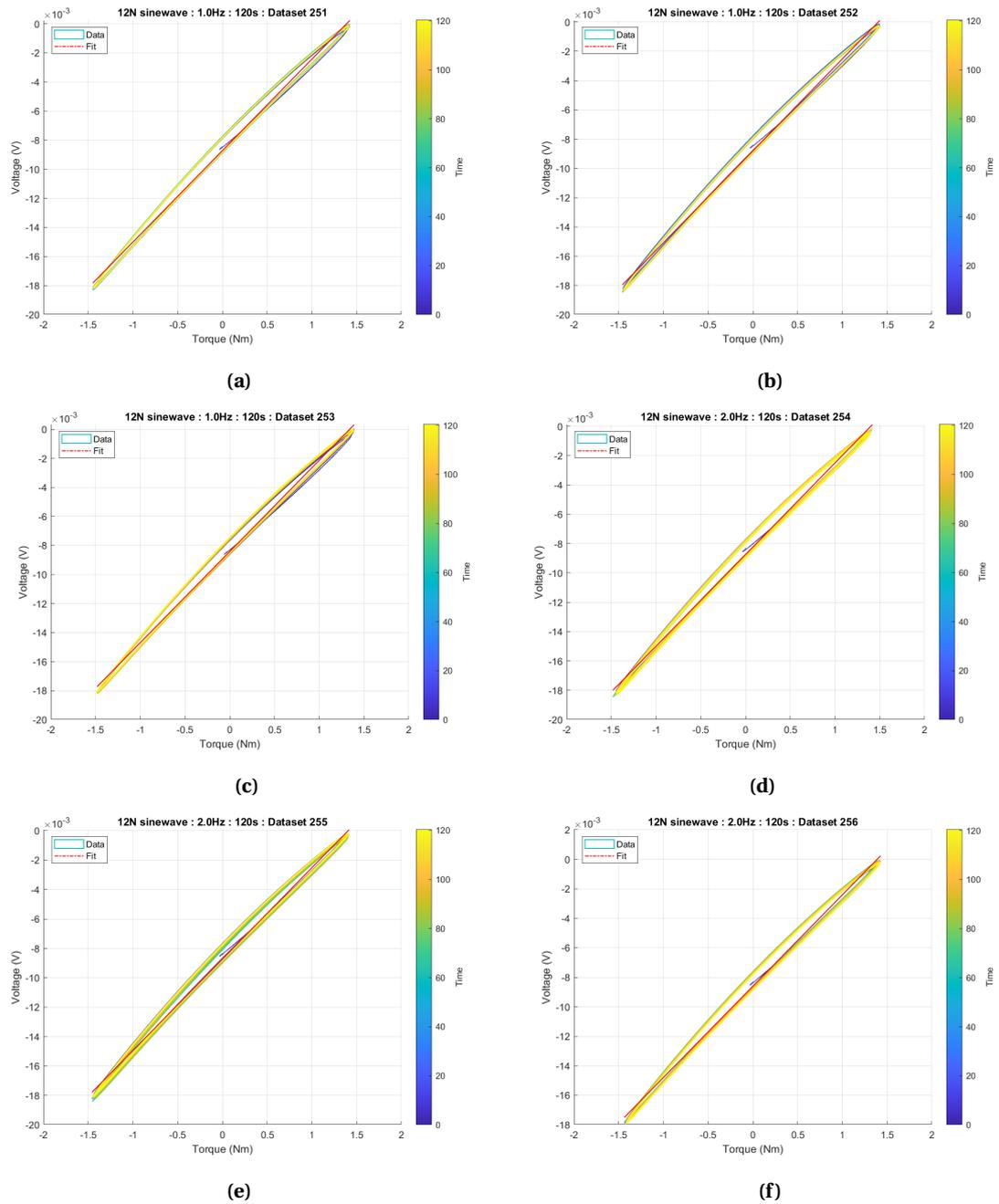


Figure F.14: Measurement results 12N sine wave, 1 & 2 Hz.

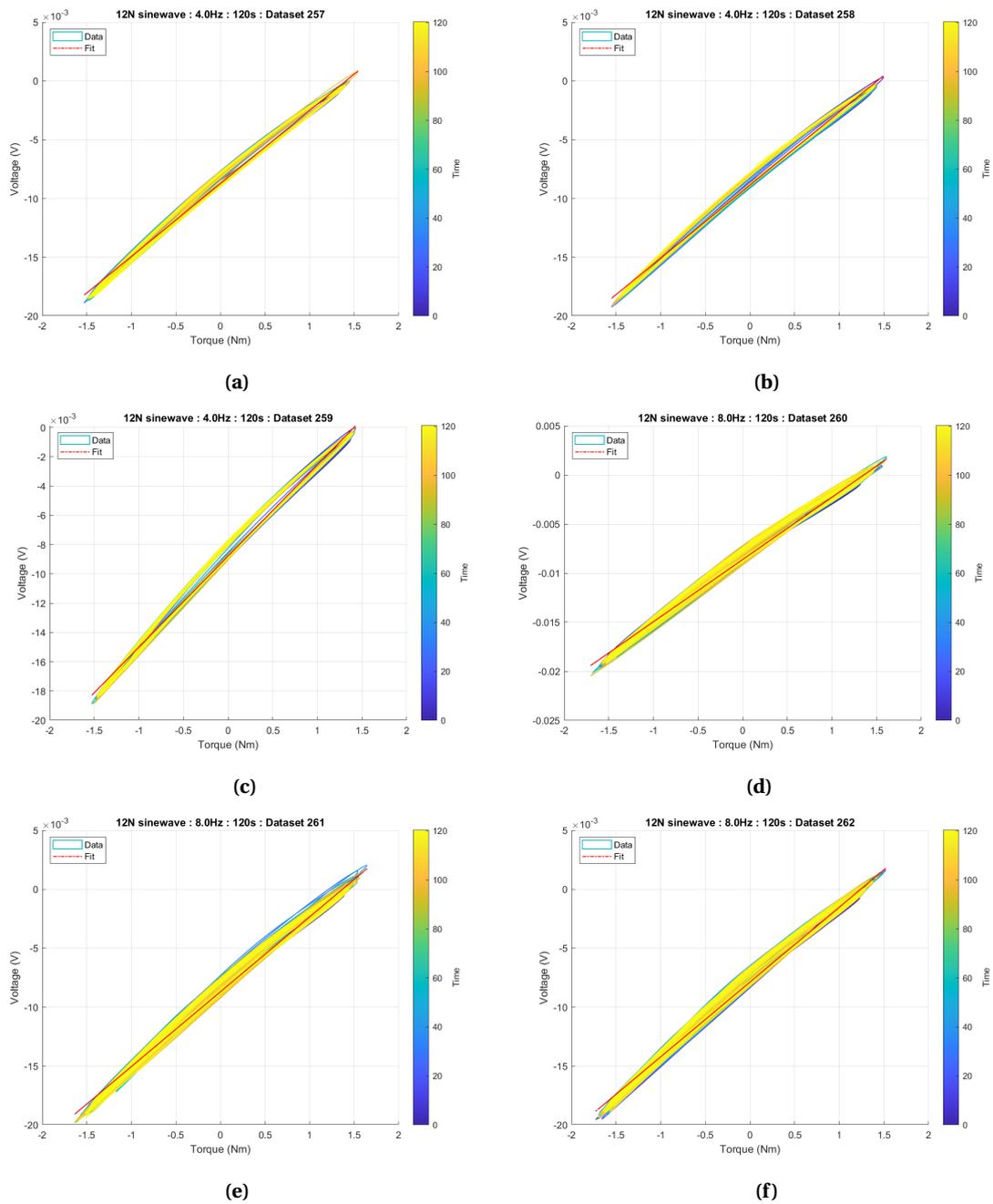
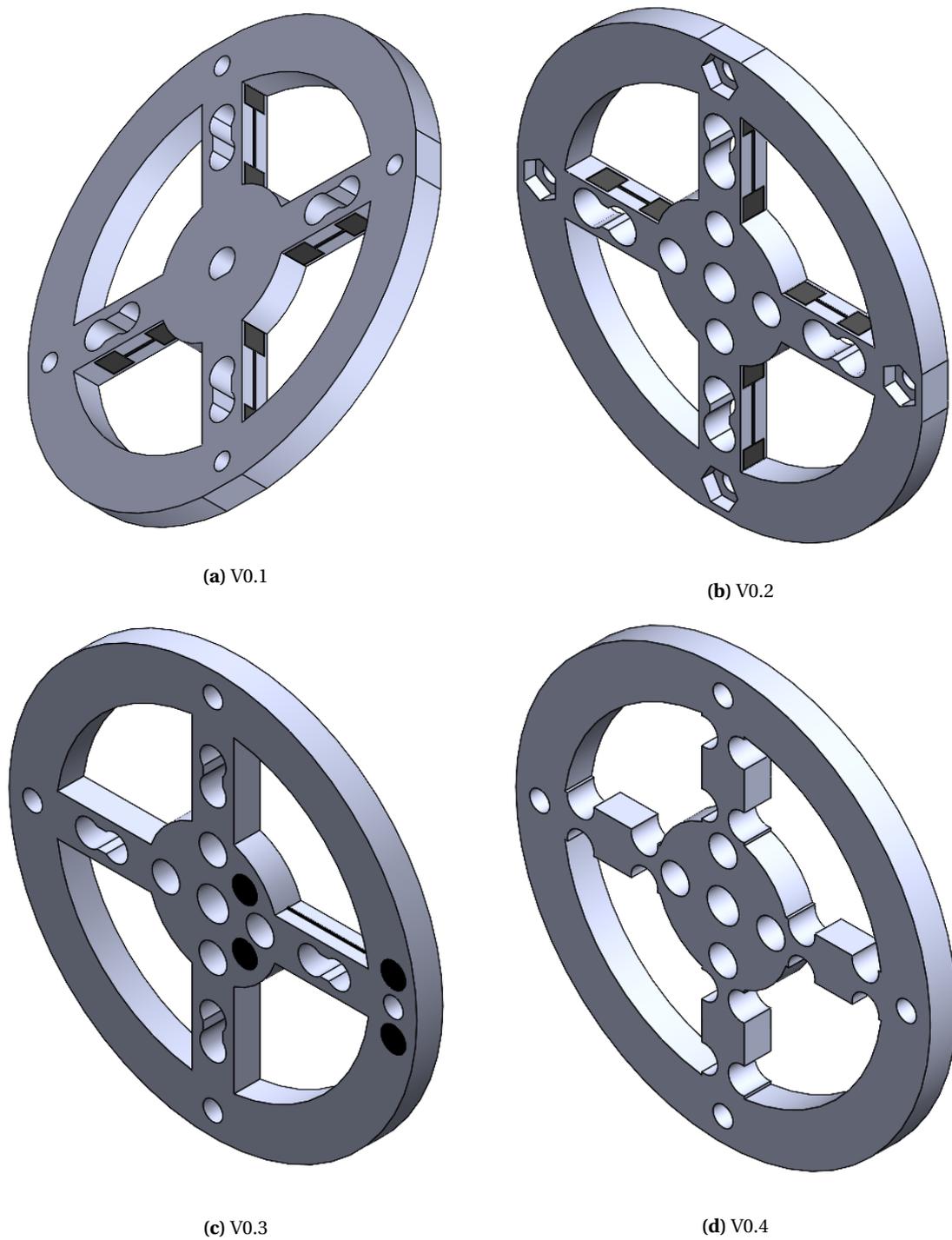
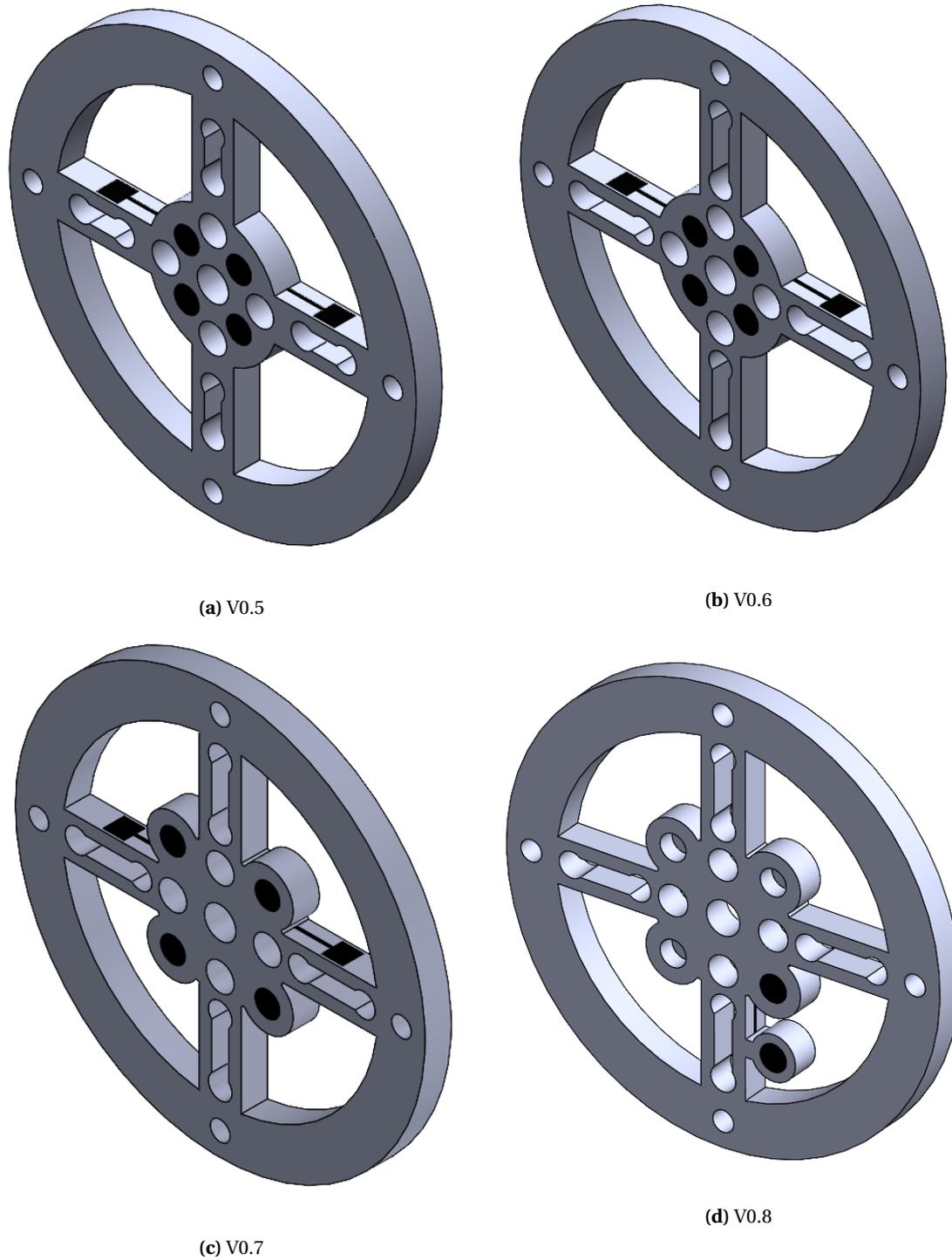


Figure F.15: Measurement results 12N sine wave, 4 & 8 Hz.

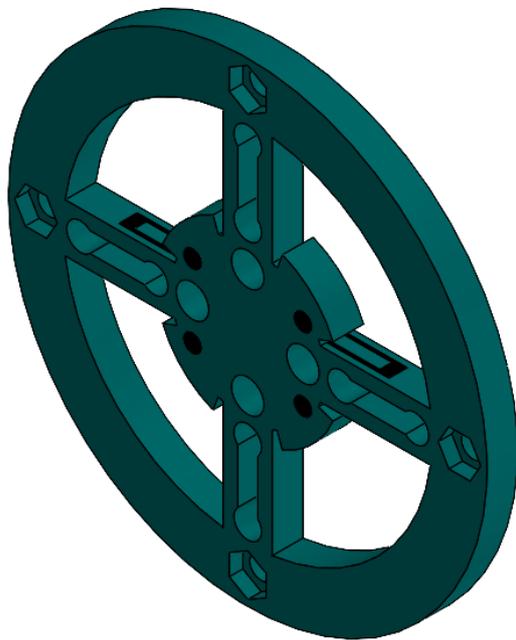
## G Design Iterations



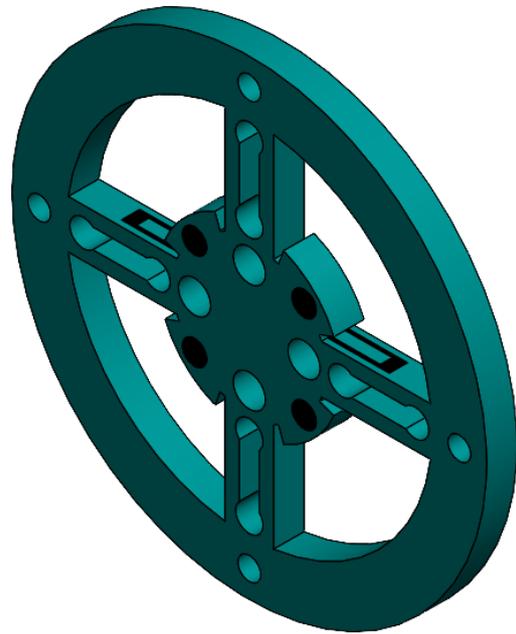
**Figure G.1:** Design iterations of the sensor V0.1 to V0.4.



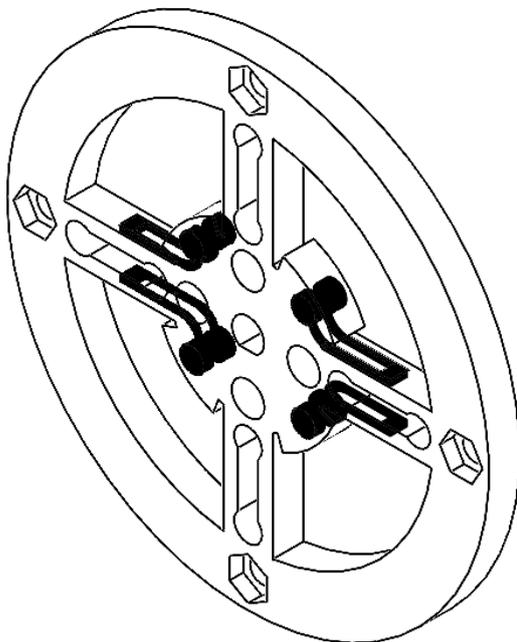
**Figure G.2:** Design iterations of the sensor V0.5 to V0.8.



(a) V0.9



(b) V0.10



(c) V0.11



(d) V0.12

**Figure G.3:** Design iterations of the sensor V0.9 to V0.12.

# H NEMA17 Datasheet

Figure H.1 views the datasheet of the motor.

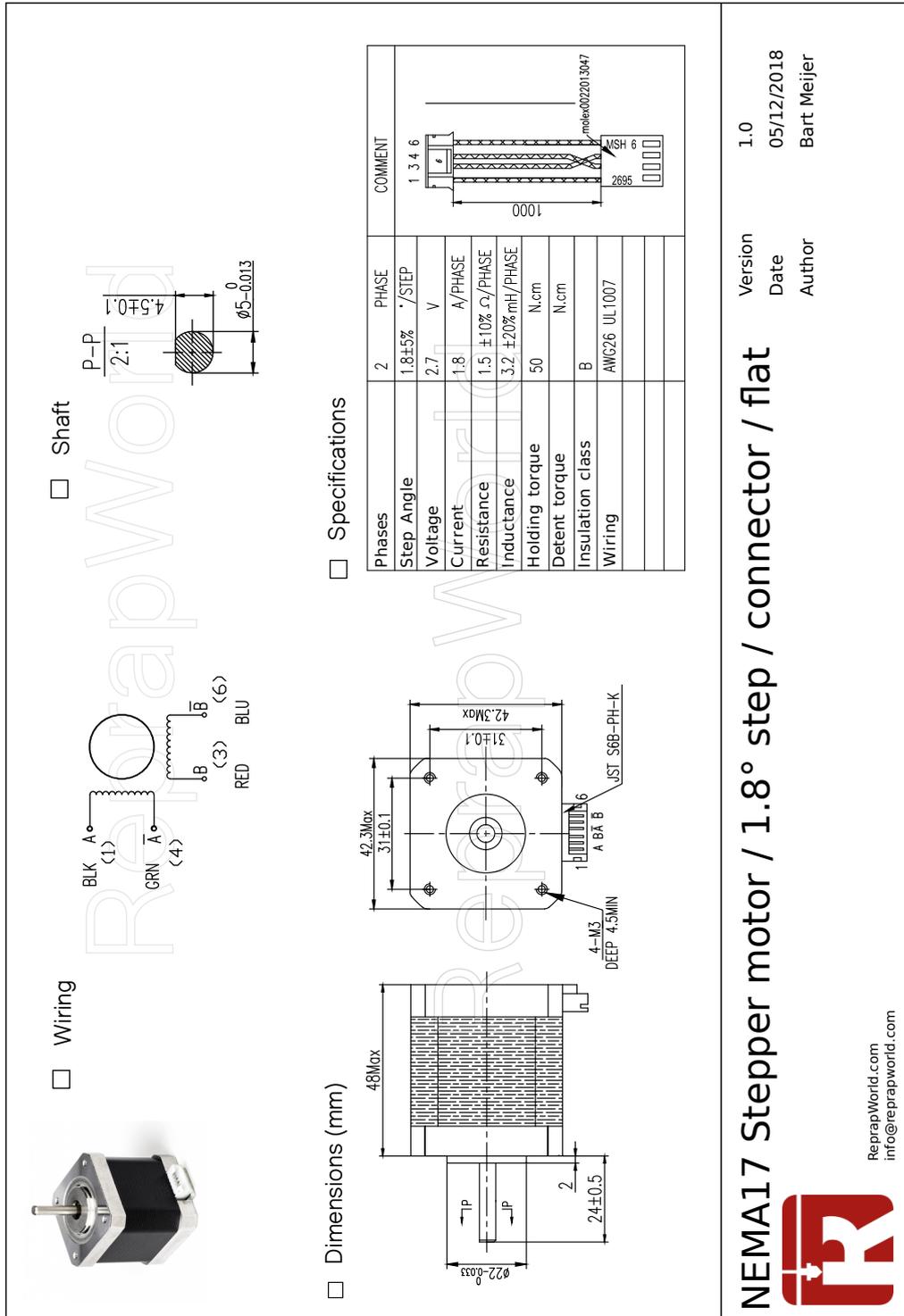


Figure H.1: Datasheet of the NEMA17. This motor is used in the setup.

## I SMAC Datasheet

Table I.1 contains the datasheet of the SMAC LCA25-050-15-6 that was used in the characterization of this work.

**Table I.1:** Datasheet of the SMAC LCA25-050-15-6 [13].

No	DETAILS	UNIT	LCA25-050-15-6
1	LINEAR STROKE	mm	50
2	PEAK FORCE	N	15
3	FORCE CONSTANT	N/A	9.5
4	MAXIMUM CURRENT	Amp	1.6
5	COIL RESISTANCE	Ohm	15
6	MOVING MASS	kg	0.082
7	TOTAL MASS	kg	<i>TBD</i>
8	DIMENSION (L)	mm	155
9	DIMENSION (A)	mm	135
10	DIMENSION (B)	mm	TBD
11	(N) # MOUNTING HOLES	#	8

## Bibliography

- [1] Electronics-Tutorials, "Introduction to capacitors," 2022.
- [2] M. M. I. M. Hasnan, M. F. M. Sabri, S. M. Said, and N. N. N. Ghazali, "Modeling of a high force density fishbone shaped electrostatic comb drive microactuator," *Scientific World Journal*, vol. 2014, 2014.
- [3] N. Innovations, "Measuring strain with strain gages," 2022.
- [4] Zureks, "Fdm schematics," 2022.
- [5] M. Schouten, G. Wolterink, A. Dijkshoorn, D. Kosmas, S. Stramigioli, and G. Krijnen, "A review of extrusion-based 3d printing for the fabrication of electro- and biomechanical sensors," *SPECIAL ISSUE ON 20 YEARS IEEE SENSORS JOURNAL*, 2020.
- [6] G. Ellis, *Nonlinear Behavior and Time Variation*. Elsevier, 2012.
- [7] Y. Lou, J. Wei, and S. Song, "Design and optimization of a joint torque sensor for robot collision detection," *IEEE Sensors Journal*, vol. 19, pp. 6618–6627, 8 2019.
- [8] natuurkunde.nl, "Gewicht berekenen wanneer 2 punten samenkomen," 2022.
- [9] DEWEsoft, "Dewe-43a usb data acquisition system," 2022.
- [10] MIT, "Massachusetts institute of technology department of mechanical engineering 2.14 analysis and design of feedback control systems understanding poles and zeros 1 system poles and zeros."
- [11] Kyowa, "Move into the future with reliable measurements," 2022.
- [12] R. Roy and A. Mukhopadhyay, "Tribological studies of 3d printed abs and pla plastic parts," *Materials Today: Proceedings*, vol. 41, pp. 856–862, 2021. Advancements and Futuristic Trends in Mechanical and Materials Engineering.
- [13] Willburger, "Lca25-xxx-yy-z," 2022.
- [14] Formlabs, "Guide to stereolithography (sla) 3d printing," 2023.
- [15] J. Patents, "Apparatus for production of three-dimensional objects by stereolithography," 1984.
- [16] "United states patent (19) crump (54) apparatus and method for creating three-dimensional objects," 1988.
- [17] A. Chapman, "History of 3d printing," 2022.
- [18] M. Vaezi, S. Chianrabuttra, B. Mellor, and S. Yang, "Multiple material additive manufacturing - part 1: A review: This review paper covers a decade of research on multiple material additive manufacturing technologies which can produce complex geometry parts with different materials," *Virtual and Physical Prototyping*, vol. 8, pp. 19–50, 3 2013.
- [19] Y. Xu, X. Wu, X. Guo, B. Kong, M. Zhang, X. Qian, S. Mi, and W. Sun, "The boom in 3d-printed sensor technology," *Sensors (Basel, Switzerland)*, vol. 17, 5 2017. Piezoresistive sensors<br/>.
- [20] B. Prakken, "3d printed flow sensor," Master's thesis, University of Twente, 2019.

- [21] B. Eijking, "Tactile whisker sensor using flexible 3d printed transducers," Master's thesis, University of Twente, 2017.
- [22] M. Schouten, B. Prakken, R. Sanders, and G. Krijnen, "Linearisation of a 3d printed flexible tactile sensor based on piezoresistive sensing," in *2019 IEEE SENSORS*, pp. 1–4, Oct 2019.
- [23] M. Schouten, C. Spaan, D. Kosmas, R. Sanders, and G. Krijnen, "3d printed capacitive shear and normal force sensor using a highly flexible dielectric," *IEEE Sensors Applications Symposium (SAS)*, pp. 1–6, 2021.
- [24] J. Oprel, G. Wolterink, and dr JP ir Schilder, "Modeling, design and characterization of a 3d printed capacitive shear stress sensor," Master's thesis, University of Twente, 2021.
- [25] M. Ross, "Pairwise-comparison-white-paper," 2020.
- [26] M. Schouten, R. Sanders, and G. Krijnen, "3d printed flexible capacitive force sensor with a simple micro-controller based readout," in *2017 IEEE SENSORS*, pp. 1–3, 2017.
- [27] M. Saari, B. Xia, B. Cox, P. S. Krueger, A. L. Cohen, and E. Richer, "Fabrication and analysis of a composite 3d printed capacitive force sensor," *3D Printing and Additive Manufacturing*, vol. 3, pp. 137–141, 9 2016.
- [28] F. Ahmad, A. Baig, J. O. Dennis, N. H. B. Hamid, and M. H. B. M. Khir, "Characterization of mems comb capacitor," *Microsystem Technologies*, vol. 26, 2020.
- [29] D. Hah, "Analytical design of mems variable capacitors based on shaped-finger comb-drives," *Microsystem Technologies*, vol. 28, 2018.
- [30] J. Shintake, Y. Piskarev, H. Jeong, D. Floreano, J. Shintake, S. H. Jeong, D. Floreano, and Y. Piskarev, "Full paper 1700284 (1 of 8) ultrastretchable strain sensors using carbon black-filled elastomer composites and comparison of capacitive versus resistive sensors," *Advanced Materials Technologies*, 2017.
- [31] D. Alveringh, R. A. Brookhuis, R. J. Wiegerink, G. J. M. Krijnen, M. J. D. Boer, and P. E. Veltink, "A large range multi-axis capacitive force/torque sensor realized in a single soi wafer," *University of Twente*, 2014.
- [32] J. Yin, V. J. Santos, and J. D. Posner, "Bioinspired flexible microfluidic shear force sensor skin," *Sensors and Actuators, A: Physical*, vol. 264, pp. 289–297, 9 2017.
- [33] ProtoPlant, "Protoplant, makers of proto-pasta," 2022.
- [34] H. Jonkers, "3d printed electronics," Master's thesis, University of Twente, 2021.
- [35] AnalogDevices, "24-bit capacitance-to-digital converter with temperature sensor," 2022.
- [36] A. Devices, "Analog devices ahead of whats possible," 2022.
- [37] S. P. One Technology Way, "An-829 application note environmental compensation on the ad7142: The effects of temperature and humidity on capacitance sensors," *Analog.com*, 2022.
- [38] G. Wright, "electromagnetic interference (emi)," 2022.
- [39] A. Dijkshoorn, J. Cui, S. Stramigioli, and G. Krijnen, "First results of a soft, 3d-printed, resistive cantilever flow sensor," *FLEPS 2021 - IEEE International Conference on Flexible and Printable Sensors and Systems*, 2021.

- [40] D. Kosmas, "Model-based hysteresis compensation and control with 3d printed lousy sensors," Master's thesis, University of Twente, 2020.
- [41] S. Roodink and S. Roodink, "Controlling deformations of a 3d-printed vibrating beam using stiffness modulation," Master's thesis, University of Twente, 2020.
- [42] K. Kim, J. Park, J. hoon Suh, M. Kim, Y. Jeong, and I. Park, "3d printing of multiaxial force sensors using carbon nanotube (cnt)/thermoplastic polyurethane (tpu) filaments," *Sensors and Actuators, A: Physical*, vol. 263, pp. 493–500, 8 2017.
- [43] J. F. Christ, N. Aliheidari, A. Ameli, and P. Pötschke, "3d printed highly elastic strain sensors of multiwalled carbon nanotube/thermoplastic polyurethane nanocomposites," *Materials & Design*, vol. 131, pp. 394–401, 10 2017.
- [44] M. Schouten, G. Krijnen, B. Prakken, and R. Sanders, "Linearisation of a 3d printed flexible tactile sensor based on piezoresistive sensing," *University of Twente*, 2019.
- [45] N. Kashiri, J. Malzahn, and N. G. Tsagarakis, "On the sensor design of torque controlled actuators: A comparison study of strain gauge and encoder-based principles," *IEEE Robotics and Automation Letters*, vol. 2, pp. 1186–1194, 4 2017.
- [46] T. Techniques, "Quarter bridge, half bridge and full wheatstone brdige strain gauge load cell configurations.," 2022.
- [47] D. Engineering, "Desktop cnc machines: Diabase engineering: Longmont," 2019.
- [48] V. E. Kuznetsov, A. N. Solonin, O. D. Urzhumtsev, R. Schilling, and A. G. Tavitov, "Strength of pla components fabricated with fused deposition technology using a desktop 3d printer as a function of geometrical parameters of the process," *Polymers*, vol. 10, 3 2018.
- [49] S. Vouyiouka and C. Papaspyrides, "4.34 - mechanistic aspects of solid-state polycondensation," in *Polymer Science: A Comprehensive Reference* (K. Matyjaszewski and M. Möller, eds.), pp. 857–874, Amsterdam: Elsevier, 2012.
- [50] S. Warke and V. Puranik, "Comparison of energy consumption of abs and pla while 3 d printing with fused deposition modeling process," *Materials Today: Proceedings*, vol. 66, pp. 2098–2103, 2022. 2022 International Conference on Recent Advances in Engineering Materials.
- [51] Fenner Drives, "Ninjatek.com." url: <https://ninjatek.com>, Accessed: 22-09-2022.
- [52] A. Okamura, "Lecture 5: Robot dynamics and simulation," 2019.
- [53] A. Heins, "Implementing lagrangian mechanics three ways," 2022.
- [54] K. Rafael and L. A, *Control of robot manipulators in joint space*. Springer, 2005.
- [55] S. Echeandia and P. M. Wensing, "Numerical methods to compute the coriolis matrix and christoffel symbols for rigid-body systems," *University of Twente*, 10 2020.
- [56] N. Cross, *Engineering design methods : strategies for product design*. John Wiley & Sons, 2008.
- [57] Farnell, "silver conductive epoxy adhesive, moderate cure / high ,farnell," 2022.
- [58] D. Systèmes, "Solidworks," 2023.

- 
- [59] A. Dijkshoorn, M. Schouten, G. Wolterink, R. Sanders, S. Stramigioli, and G. Krijnen, "Characterizing the electrical properties of anisotropic, 3d-printed conductive sheets for sensor applications," *IEEE Sensors Journal*, vol. 20, no. 23, pp. 14218–14227, 2020.
- [60] Ultimaker B.V., "Cura," 2021.
- [61] DEWESoft, "Dewesoft hardware manuals," 2022.
- [62] INGENIA, "Embedded motion control library," 2022.
- [63] R. Nave, "Parallel axis theorem," 2022.
- [64] G. Ellis, *The Frequency Domain*. Elsevier, 2012.
- [65] D. Matt, "Small-angle approximation," 2022.
- [66] C. J. Bett, "9 - gain-scheduled controllers," in *The Electrical Engineering Handbook* (W.-K. CHEN, ed.), pp. 1107–1114, Burlington: Academic Press, 2005.
- [67] C. Tutorials, "Introduction: Pid controller design," 2022.
- [68] M. I. of Technology, "6.01 review of linear systems," 2008.
- [69] T. Labs, "Mechaduino hardware versions 0.1, 0.2," 2023.
- [70] T. Labs, "Mechaduino 0.1 manual," 2023.
- [71] R. Keim, "s-domain circuit analysis," 2023.