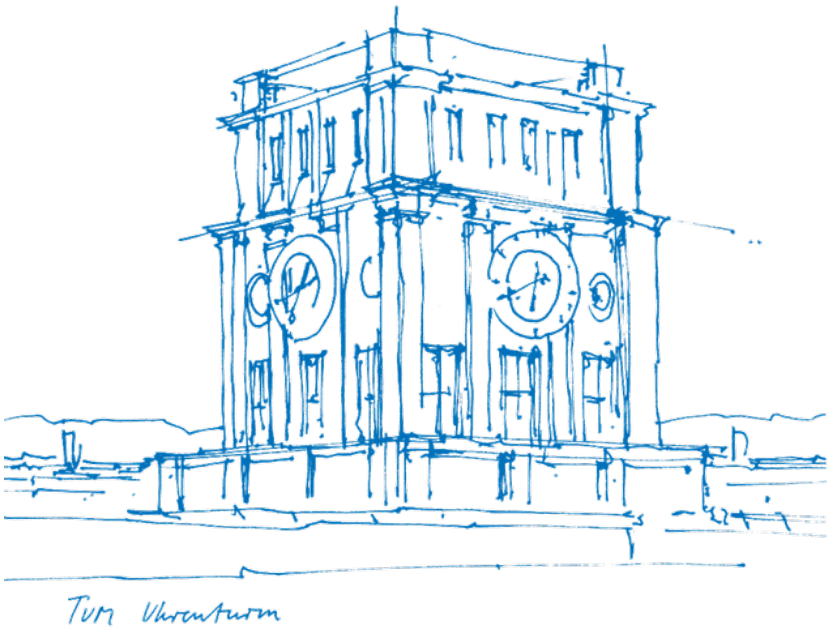
Munich Institute of Robotics and Machine Intelligence
Technical University of Munich

Investigating Force and Moment Scaling in Robotic Force-Torque Sensors

Forschungspraxis Report

Yunus Emre Danabaş



Investigating Force and Moment Scaling in Robotic Force-Torque Sensors

Forschungspraxis Report

Yunus Emre Danabaş

October 8, 2024

Chapter 0: Contents

Chapter 0: Investigating Force and Moment Scaling in Robotic Force-Torque Sensors

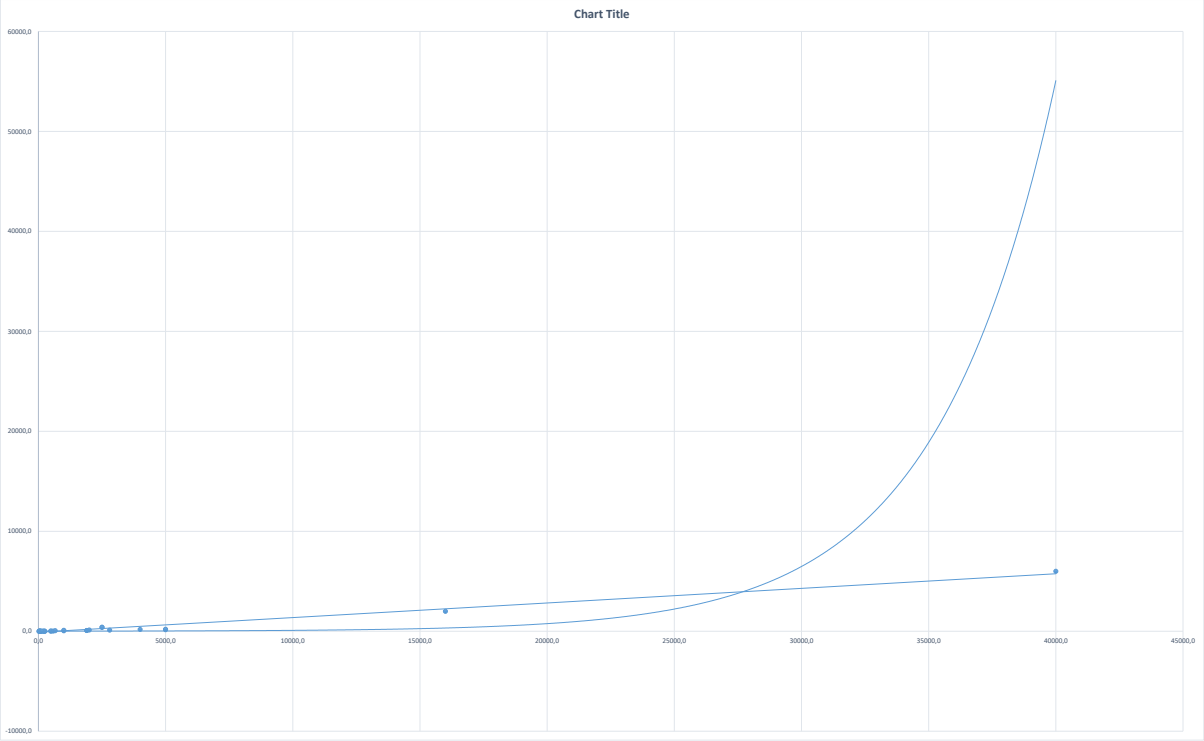
In advanced robotics and automation systems, accurate force and torque measurements are critical to ensuring robotic arms and manipulators' precise and safe operation. Force-torque (FT) sensors, particularly those with six degrees of freedom (6-DoF), are commonly used for this purpose, enabling robots to measure forces and moments in all three axes (Fx, Fy, Fz, Mx, My, Mz). However, existing 6-DoF sensors face inherent limitations, particularly when measuring higher moment values. This problem becomes more pronounced as applications demand high-resolution measurements and the ability to withstand large forces and moments.

A central issue in the current market of 6-DoF FT sensors is the trade-off between their force and moment capacities. Data from commercially available sensors show a clear trend: as the maximum moment capabilities (Mx, My, Mz) increase, the corresponding force capacities (Fx, Fy, Fz) also rise exponentially. This scaling means that to achieve a higher moment range, the sensor must also handle substantially higher forces, even if those forces are not required for the specific application. This makes the sensor bulkier and heavier and reduces its resolution as the increased structural demands make it harder to detect smaller variations in strain. Consequently, the trade-off between sensitivity and load capacity becomes a major limitation in many high-performance robotic applications.

Additionally, while many sensors can handle the maximum forces exerted by robotic arms, they often struggle with the maximum moments generated during operation. For instance, the Franka Emika Panda can exert horizontal forces of up to 300 N and vertical forces up to 410 N at its mounting base, values that fall within the measurement

Figure 1

Graph showing the relationship between maximum force and moment capacities of commercial 6-DoF force-torque sensors. As the maximum moment increases, the required force capacity rises exponentially.



range of most commercially available 6-DoF force-torque sensors. These force values are manageable for sensors designed for high-precision robotic applications. However, the torques exerted during rotational or tilting tasks present a more significant challenge. The Franka Emika Panda is capable of generating tilting torques up to 280 Nm and rotational torques around its base axis up to 190 Nm. These high torque values are often beyond the upper limits of standard force-torque sensors, which struggle to accurately measure such moments while maintaining the same precision for lower forces. This limitation is particularly evident in dynamic tasks where the robot manipulates off-center loads or performs intricate rotational movements.

Similarly, the ABB YuMi (IRB 14000) also faces challenges with torque measurement. During normal operation, it can handle torques up to ±101 Nm along the x-axis and ±61 Nm along the z-axis, which are manageable for many sensors. However,

in emergency stop conditions, these values increase significantly, with torques reaching up to ±202 Nm along the x-axis and ±122 Nm along the z-axis. While YuMi's force capacities are within the sensor limits, the elevated torque demands during critical operations exceed the capabilities of conventional sensors. This discrepancy highlights the limitations in measuring high torques accurately while maintaining force measurement precision.

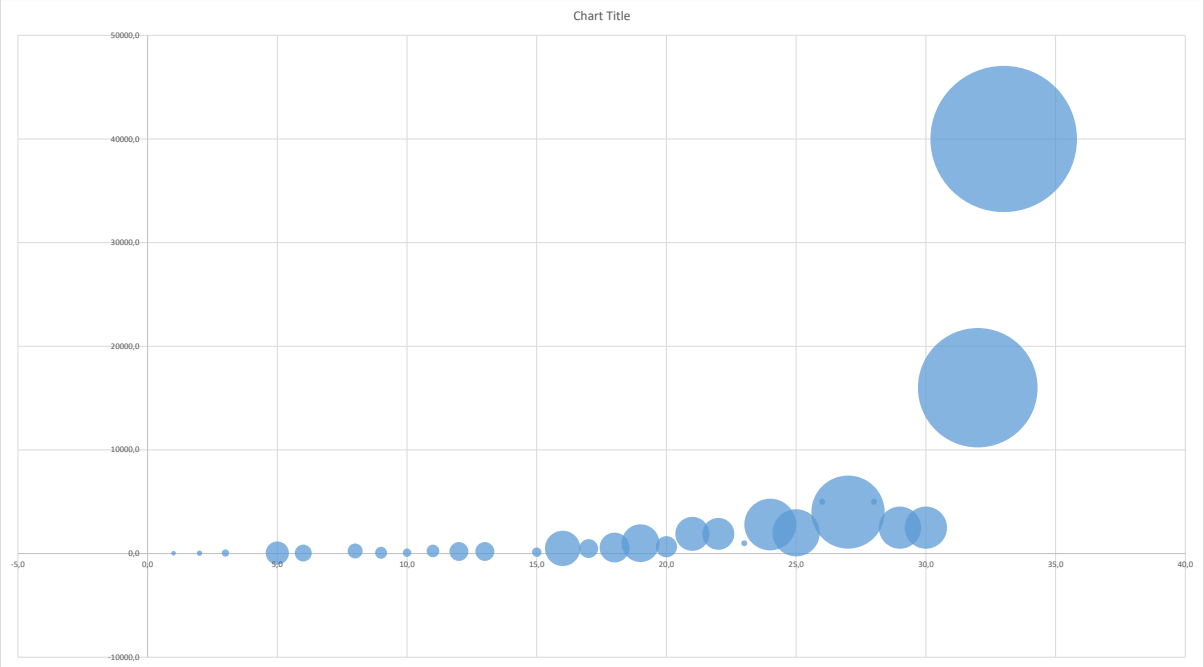
This discrepancy between force and moment measurement capabilities hinders the full utilization of robotic systems, especially in tasks requiring precise force and torque control. This challenge is particularly relevant in collaborative robots (cobots) and industrial manipulators, where accurate torque measurement is vital for tasks like assembly, welding, and material handling. The inability of current sensors to handle high moments without sacrificing force measurement accuracy creates a performance gap, limiting the robots' operational range. Therefore, a design approach that balances the demands of force and moment measurement while maintaining high sensitivity and resolution is necessary to fully leverage the capabilities of modern robotic systems.

Visualized data further demonstrates the shortcomings of current FT sensors. The plots of maximum moment versus maximum force show an exponential relationship, where sensors designed for higher moments require significantly greater force capacities. The scatter plot shows that larger sensor load capacities correspond to reduced resolution. This highlights the trade-off between strength and sensitivity, where improvements in one area come at the expense of the other. As a result, relying on current sensor designs to achieve both high force and moment capacities while retaining precision proves increasingly impractical.

One promising solution to these challenges is the integration of flexure hinges into the sensor structure. Flexure hinges are mechanical components that enable precise,

Figure 2

Scatter plot illustrating the relationship between sensor moment capacity, force capacity, and resolution. Larger data points indicate sensors with lower resolution (i.e., higher minimum measurable force). As force and moment capacities increase, sensor resolution decreases.



repeatable motion through the elastic deformation of material rather than relying on traditional mechanical joints with moving parts. This allows for high precision with minimal friction, wear, or backlash, making them ideal for applications that require fine measurement and control.

In the context of FT sensors, incorporating flexure hinges can significantly improve both moment and force measurement accuracy. Flexure hinges allow for smoother, more predictable deformations when forces and moments are applied, enabling strain to be distributed more uniformly across the sensor. This reduces stress concentrations that are common in traditional designs, where strain is localized in small areas, reducing resolution. By using flexure hinges, sensors can maintain high sensitivity while handling larger loads, enhancing the overall precision of force and moment measurements.

Additionally, flexure hinges reduce the need for reinforcement, as they introduce fewer moving parts than traditional mechanical joints or segmented structures. This

results in more compact, lightweight, and durable sensors, which is crucial for high-performance robotic applications where both precision and durability are essential.

Beyond flexure hinges, another complementary design strategy is using prismatic sensor bodies. By distributing strain across a larger surface area, prismatic structures can enhance moment sensitivity without sacrificing resolution. However, it is important to note that this report does not propose any specific design solutions. Instead, it aims to examine and investigate these concepts through data collection and experimentation to better understand the limitations and opportunities for improvement in current sensor technologies.

Chapter 0: Part 1: Flexure Hinges

Introduction

Flexural pivots, or cross-spring pivots, are mechanical devices characterized by their high compliance in the "in-plane" rotational degree of freedom (θ) while maintaining significant stiffness in all other degrees of freedom. These pivots typically feature a bi-symmetrical design with two equally dimensioned leaf springs intersecting at their midpoints, forming an angle of 2α, which is generally set to $\frac{\pi}{2}$ for optimal stability, stiffness, and ease of construction Zelenika and Bona, 2002.

In our case, we initially created a flexure hinge according to Gunnink's optimal three flexure cross hinge geometries for stiffness Gunnink et al., 2013, which requires $2\alpha = 62°$ and $2\alpha = \frac{\pi}{2}$ without any optimization process.

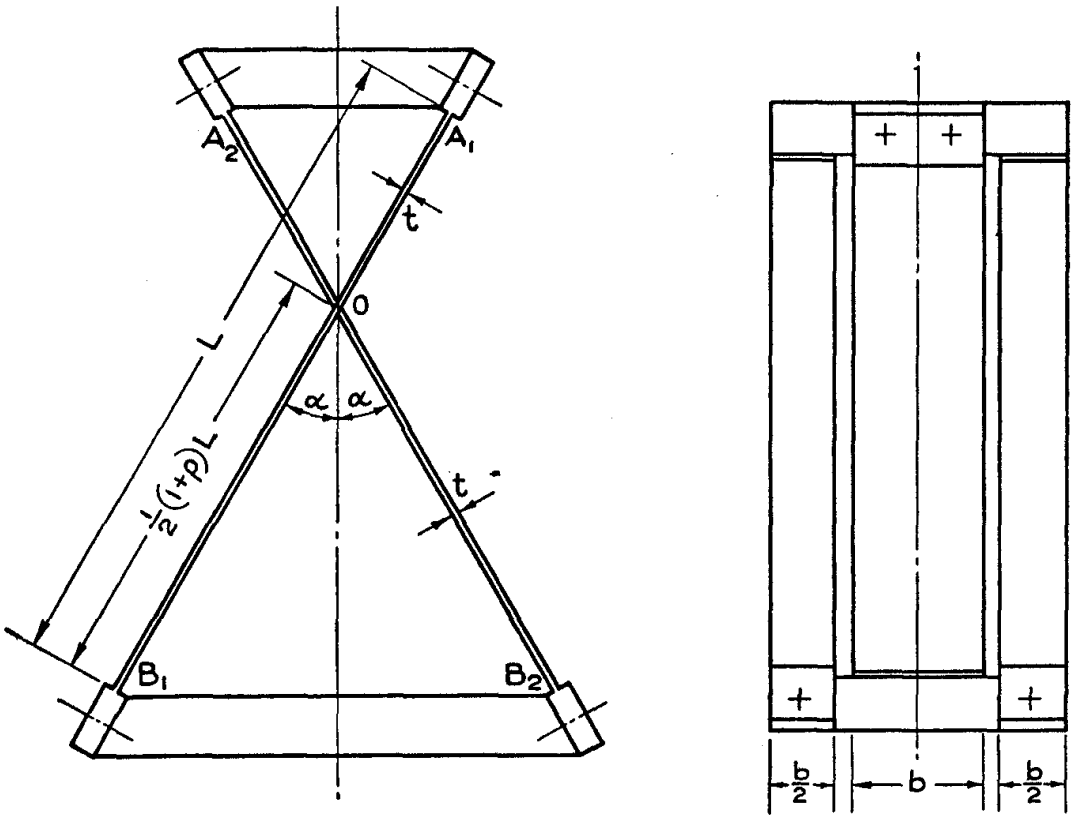
Analytical Model

The approach can evaluate the parasitic motion of the pivot suggested in Wittrick's paper Wittrick, 1951. This paper primarily investigates the effect of changing the point at which the strips cross. It is shown that the variation of rotational stiffness with applied force can be significantly altered by adopting this simple modification. For instance, when the strips cross at one end, the pivot remains stable with the strips in tension, although substantial variations in stiffness with applied force still occur.

Additionally, the linearity of the torque-rotation curve is examined, revealing that deviations from linearity are more pronounced with a side force acting on the pivot than without.

As illustrated in Figure 3, the parameter P defines the point at which the undeflected strips cross. The strips are arranged symmetrically about both vertical center

Figure 3
Witteick Crossed Flexure Pivot.

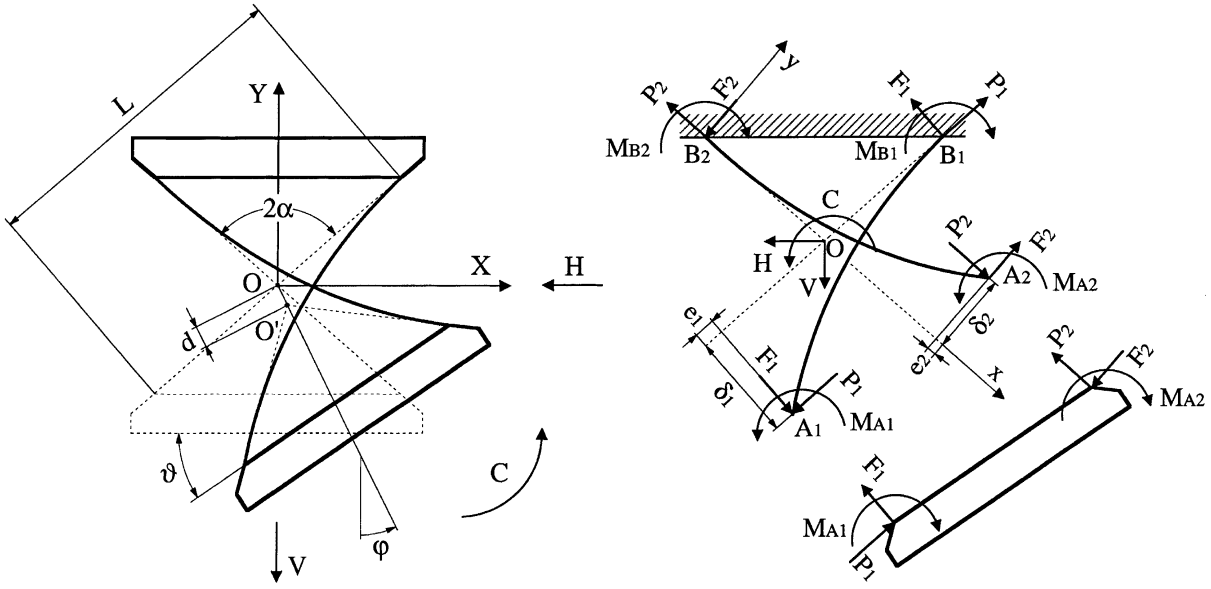


(a) *Front View* **(b)** *Side View*

lines, inclined at an angle α to the vertical, and cross at point O, located at a distance of $\frac{1+P}{2}L$ from the moving ends. When $P = 0$, the strips cross at their midpoints; when $P = \pm 1$, they cross at the fixed or moving ends, respectively. For the purpose of this analysis, we assume $P = 0$, so the strips cross at their midpoints, as shown in Figure 4 Zelenika and Bona, 2002.

In this context, the primary focus of the analysis was to determine the device's stiffness near the undeflected position ($\vartheta = 0$). Thus, terms smaller than ϑ^2 were omitted. This decision was made to avoid complicating the problem with strong non-linearity, which could have made the analysis more complex and less tractable.

Figure 4
Bent Hinge Figures



(a) *Flexural pivot* **(b)** *Displacements and reactions*

When analyzing the equilibrium conditions of a pivot where the undeflected spring strips intersect at their midpoints (Figs. 4), 11 variables are necessary to describe its behavior: e_1, e_2 (shortening of a strip due to change of slope), δ_1, δ_2 (lateral deflection of the moving end of a strip), P_1, P_2, F_1, F_2, M_{B1}, M_{B2}, and ϑ, consequently, 11 equations are needed to solve this system.

Consider a scenario where the moving end of the pivot is subjected to a force system with vertical and horizontal components V and H at point O. These forces represent the external loads acting on the pivot. Additionally, a couple C around this point is applied, representing a moment that tends to rotate the pivot. This scenario is illustrated in Fig. 4.

The following equations can describe the equilibrium conditions on the center of the hinge:

$$V = (P_1 + P_2)\cos\alpha + (F_1 - F_2)\sin\alpha \tag{0.1}$$

$$H = (P_1 - P_2)\sin\alpha - (F_1 + F_2)\cos\alpha \tag{0.2}$$

$$C = M_{B1} - F_1 L\frac{1}{2} + M_{B2} - F_2 L\frac{1}{2} \tag{0.3}$$

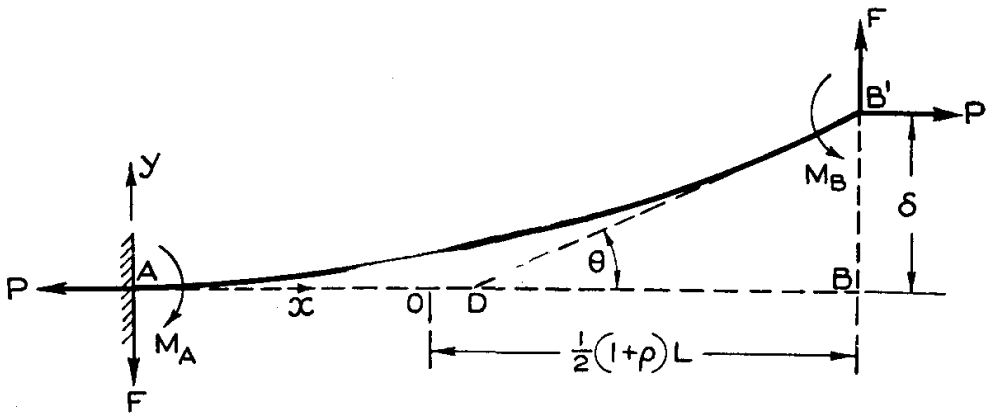
$$= M_{A1} + F_1\left(L\frac{1}{2} - e_1\right) - P_1\delta_1 + M_{A2} + F_2\left(L\frac{1}{2} - e_2\right) - P_2\delta_2 \tag{0.4}$$

Additionally, the deflection of the bottom part can be expressed in terms of δ, e, and α:

$$(\delta_1 - \delta_2)\cos\alpha + (e_1 + e_2)\sin\alpha = L\sin\alpha(1 - \cos\theta) \tag{0.5}$$

$$(\delta_1 + \delta_2)\sin\alpha - (e_1 - e_2)\cos\alpha = L\sin\alpha\sin\theta \tag{0.6}$$

Figure 5
The Bending of a Single Strip



The remaining equations are derived by examining the balance of the single-leaf springs (Figs. 5) using various analytical methods with differing precision. To

make the problem easier to handle and more general, these equations are written in a dimensionless form. This approach helps to simplify the calculations while still considering the mechanical properties of the springs.

$$\lambda_{1(2)} = \frac{e_{1(2)}}{\vartheta L} \tag{0.7}$$

$$\xi_{1(2)} = \frac{2\delta_{1(2)}}{\vartheta L} - 1 \tag{0.8}$$

$$f_{1(2)} = \frac{F_{1(2)} L^2}{EI} \tag{0.9}$$

$$4\beta^2_{1(2)} = \frac{P_{1(2)} L^2}{EI} \tag{0.10}$$

$$h = \frac{HL^2}{EI} \csc\alpha \tag{0.11}$$

$$v = \frac{VL^2}{EI} \sec\alpha \tag{0.12}$$

$$m_{B1(2)} = \frac{M_{B1(2)} L}{EI} \tag{0.13}$$

The parameters $\lambda_{1(2)}$ and $\xi_{1(2)}$ can be written as functions of β_1 and β_2. By substituting these expressions into Eqs. (0.1) and (0.3), and using Eq. (0.9) while assuming $2\alpha = \frac{\pi}{2}$, we can derive the following equations:

$$\begin{aligned}
&4\beta_{1(2)}(4+\vartheta^2)(\beta_{2(1)}\cosh\beta_{2(1)} - \sinh\beta_{2(1)}) \\
&\quad\times\left[\frac{1}{2}(v \pm h) - 4\beta_{2(1)}\right] \pm \beta_{2(2)}\beta_{1(2)}\cosh\beta_{2(1)} \\
&\quad\times\left[\beta_{2(1)} \mp 32 + 16\vartheta + 16\cos\vartheta(\pm 2 - \vartheta) + 16\sin\vartheta(2 \pm \vartheta) + \vartheta^2\right. \\
&\quad\times\left(\mp 4 + 2\vartheta \pm 4\coth(\beta_{1(2)}\beta_{1(2)}) \mp 2\csc^2(\beta_{1(2)})(1+\cosh 2\beta_{1(2)})\right. \\
&\quad\left.\left.+\vartheta\csc^2(\beta_{2(1)})(1+\cosh 2\beta_{2(1)})\right)\right] \\
&\quad - 2\vartheta\left[4\tanh(\beta_{2(1)})(4+\vartheta^2) + \vartheta^2\coth(\beta_{2(1)})\right] = 0
\end{aligned} \tag{0.14}$$

These equations assume that the leaf springs are under tension. For compressive forces, β becomes imaginary, so we substitute $\beta = i\omega$ in Eq. (0.14), where $\omega = -\frac{PL^2}{4EI}$.

To solve this non-linear system with β_1 and β_2 (or ω_1 and ω_2 for compression) as unknowns, we use the Newton-Raphson method iteratively. Once V and H are known, we can find v and h, and for a given ϑ, we solve Eq. (0.14) to get β_1 and β_2. Then, using

Eqs. (0.9), (0.13), and (0.3), we determine λ_1, λ_2, ξ_1, ξ_2, e_1, e_2, δ_1, δ_2, f_1, f_2, m_{B1}, and m_{B2}.

Finally, using simple geometric considerations, we can calculate $\frac{dX}{L}$ and $\frac{dY}{L}$, which represent the displacements in the X and Y directions relative to the pivot's center in terms of ϑ, λ_1, ξ_1, and α.

$$\frac{dX}{L} = \frac{\vartheta(\xi_1 + 1)}{2}\cos\alpha + \lambda_1\vartheta^2\sin\alpha + \frac{\sin(\alpha - \vartheta) - \sin\alpha}{2} \tag{0.15}$$

$$\frac{dY}{L} = \frac{\vartheta(\xi_1 + 1)}{2}\sin\alpha - \lambda_1\vartheta^2\cos\alpha - \frac{\cos(\alpha - \vartheta) - \cos\alpha}{2} \tag{0.16}$$

MATLAB Simulation

Newton-Raphson Method in Flexural Pivots

In the analysis of flexural pivots, the Newton-Raphson method is utilized to solve the non-linear system of equations derived from the system's equilibrium conditions and deflection equations. This method iteratively computes variables such as β_1 and β_2 (or ω_1 and ω_2 for compression scenarios). The iterative approach involves:

1. Estimating initial values for deflection parameters and forces.
2. Iteratively update these estimates by solving the linearized equations until convergence is achieved.

Eccentricities, Deflections, and Dimensionless Parameters

Eccentricities (e_1, e_2) and deflections (δ_1, δ_2) are critical in determining the pivot's parasitic motions and center shift. They are computed from the leaf springs' equilibrium equations and compatibility conditions, which are essential for evaluating the pivot's performance under load. Additionally, dimensionless parameters such as λ_1, λ_2, ξ_1, and ξ_2 are derived from β_1 and β_2. These dimensionless parameters normalize

the system equations, simplifying the analysis and making it independent of the pivot's actual dimensions.

Simulation Setup

MATLAB simulations were conducted to analyze the behavior of the leaf spring system and validate derived equations. The system's material and geometric properties are defined as follows:

- **Young's modulus, E:** 200 GPa
- **Leaf spring length, L:** 98 mm
- **Leaf spring width, b:** 13 mm
- **Leaf spring thickness, t:** 0.5 mm
- **Moment of inertia, I:** $\frac{bt^3}{12}$
- **Vertical load, V:** 100 N
- **Horizontal load, H:** 100 N
- **Eccentricity 2, e_2:** 15 mm
- **Deflection 2, δ_2:** 20 mm
- **Load orientation angle, α:** 45°
- **Characteristic angle, θ:** 30°

Using the function `calculate_delta1_e1`, initial deflection and eccentricity values (δ_1, e_1) are computed.

Numerical Solution: Newton-Raphson Method

The Newton-Raphson method is employed to solve for dimensionless parameters β_1 and β_2. Initial guesses and convergence criteria are:

- **Initial guesses:** $\beta_{1_{\text{init}}} = 1.0$, $\beta_{2_{\text{init}}} = 1.0$
- **Tolerance:** tol = 10^{-6}
- **Maximum iterations:** 100,000

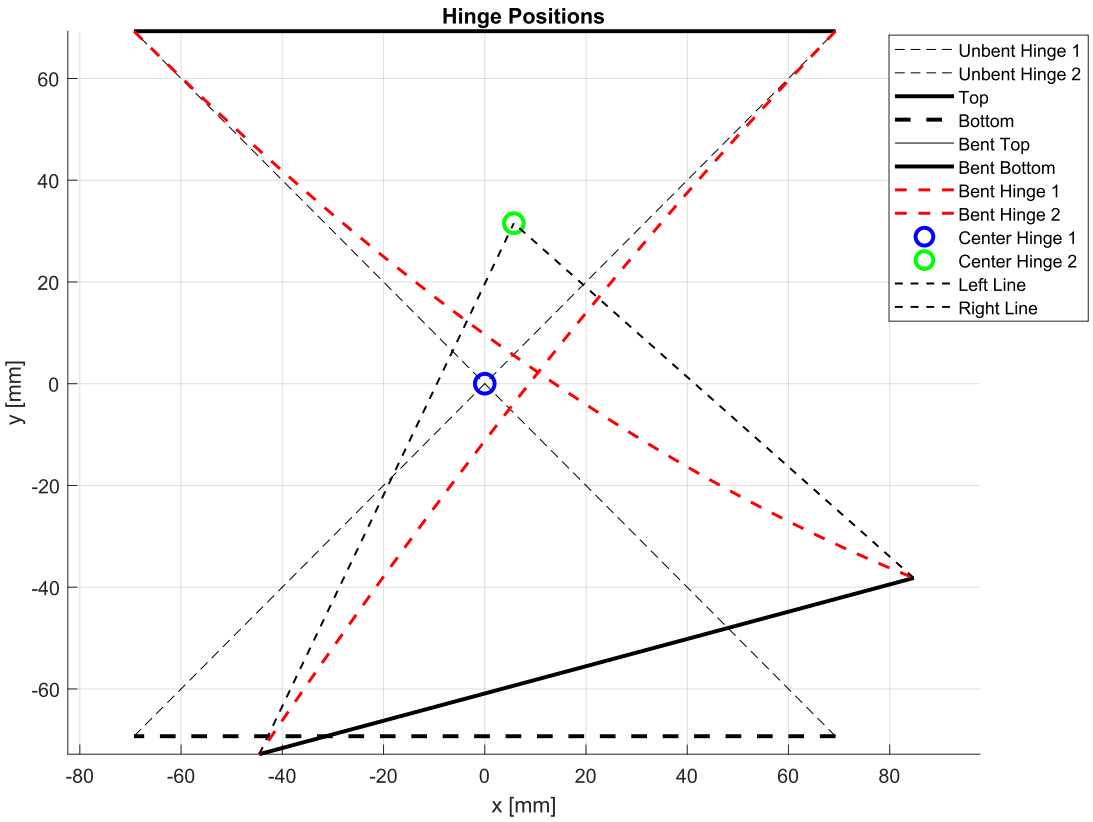
The iterative process iterates until the values of β_1 and β_2 converge.

Results and Visualization

Upon solving the nonlinear equations, dimensionless deflections (ξ_1, ξ_2) and eccentricities (λ_1, λ_2) are computed. Displacements (dX, dY) relative to the pivot center and total displacement (dD) are calculated in millimeters. Additional parameters such as forces (f_1, f_2) and moments (m_{B1}, m_{B2}) are also determined and displayed.

A plot illustrating hinge positions is generated using `plot_hinge_positions`, showing both original and bent positions of the hinges, along with calculated displacements and deformations. The figure caption should describe the content of the plot clearly.

Figure 6
Illustration of hinge positions, displacements, and deformations



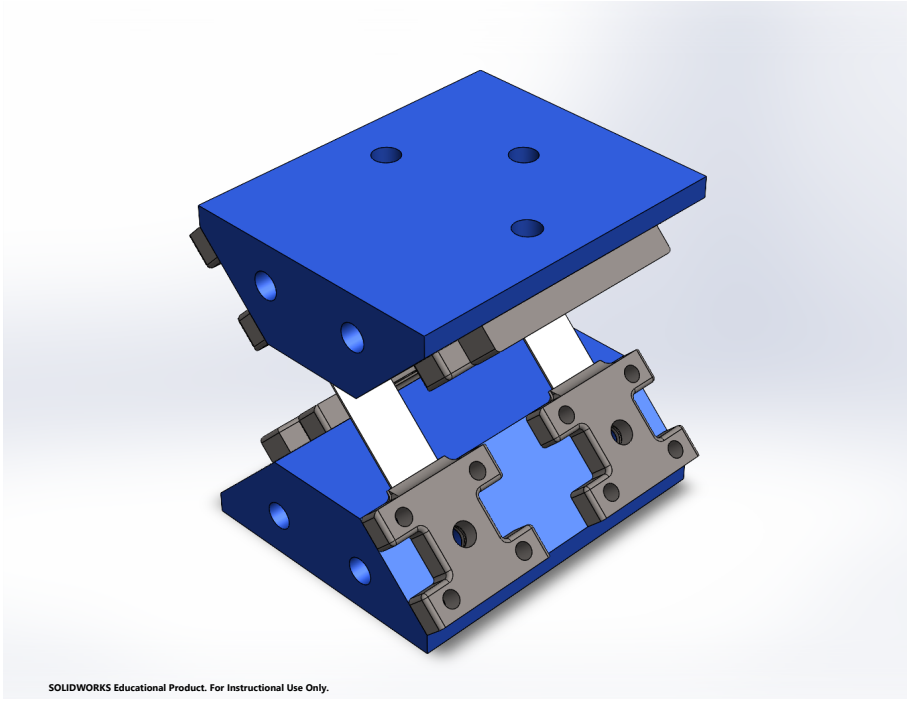
All MATLAB code utilized for simulations and calculations in this section is provided in the appendix for reference.

CAD and Prototype

CAD Design

The design process for the flexural pivot began with creating a detailed 3D model using SolidWorks. The assembly comprises several parts designed to integrate seamlessly, ensuring functionality and ease of assembly. The key components of the design include:

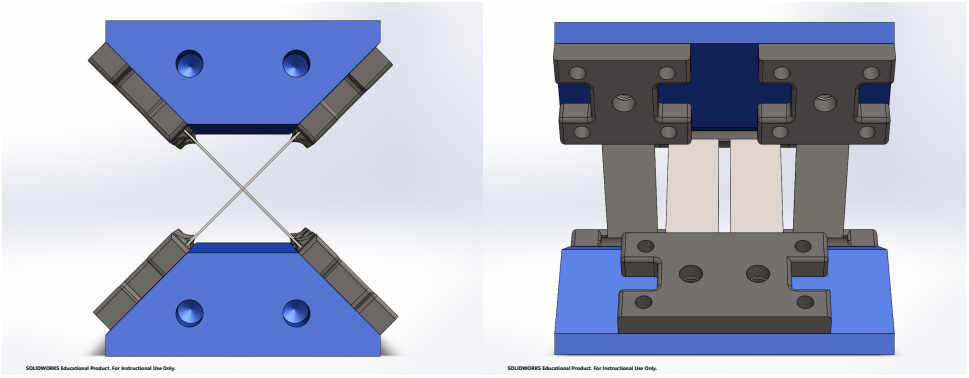
Figure 7
SolidWorks assembly of the flexure hinge.



- **Base Plate:** Serves as the foundation for the pivot, providing stability and a mounting surface for the hinges.
- **Hinge Holders:** These components hold the steel strips in place, allowing them to flex and create the desired pivot action.

- **Flexure Strips:** Made from steel, these strips are the primary elements that bend to provide a rotational degree of freedom.

Figures 7 show the overall assembly and detailed views of critical components.



(a) *Front View* **(b)** *Right View*

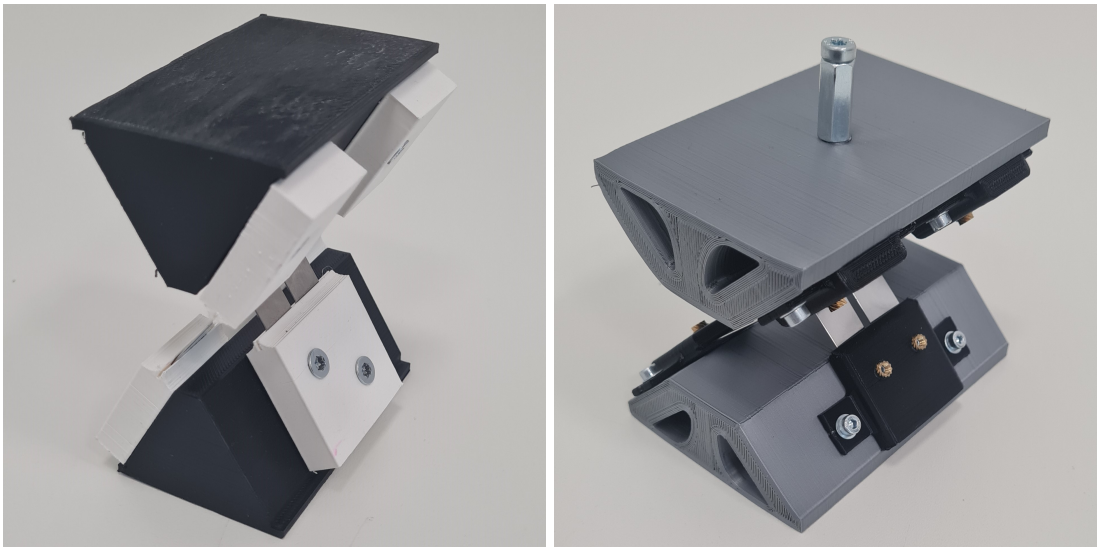
Prototype Fabrication

The prototype was fabricated using a combination of 3D printing and metalworking techniques. The primary steps involved in the fabrication process are as follows:

1. **3D Printing:** The main structural components were printed using PLA filament on a 3D printer.
2. **Steel Strips:** The flexural elements were made from steel strips cut to length and fitted into the hinge holders.
3. **Assembly:** The printed parts and steel strips were assembled using bolts and inserts to ensure proper alignment and secure fitting according to the CAD design.

I created two draft versions, iterating and updating various features to improve performance and functionality. After refining the design, the final prototype was achieved.

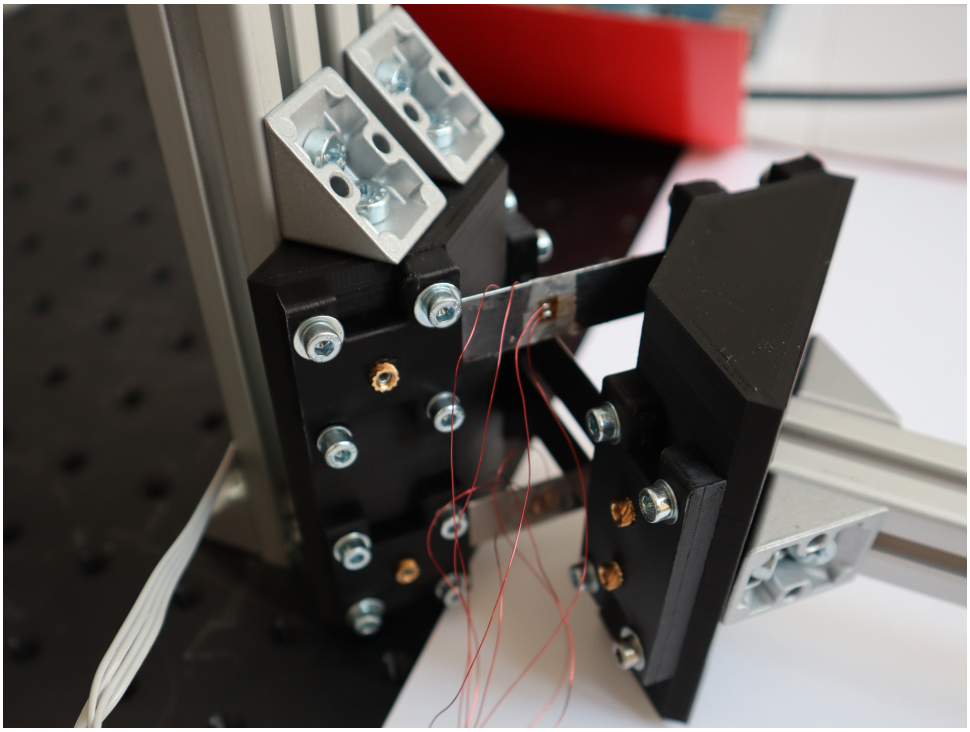
Figure 9
Draft versions of the flexural pivot prototype.



(a) *Draft Version 1*

(b) *Draft Version 2*

Figure 10
Final prototype of the flexural pivot.



Analysis of Flexure Hinge Bending Using Computer Vision Techniques

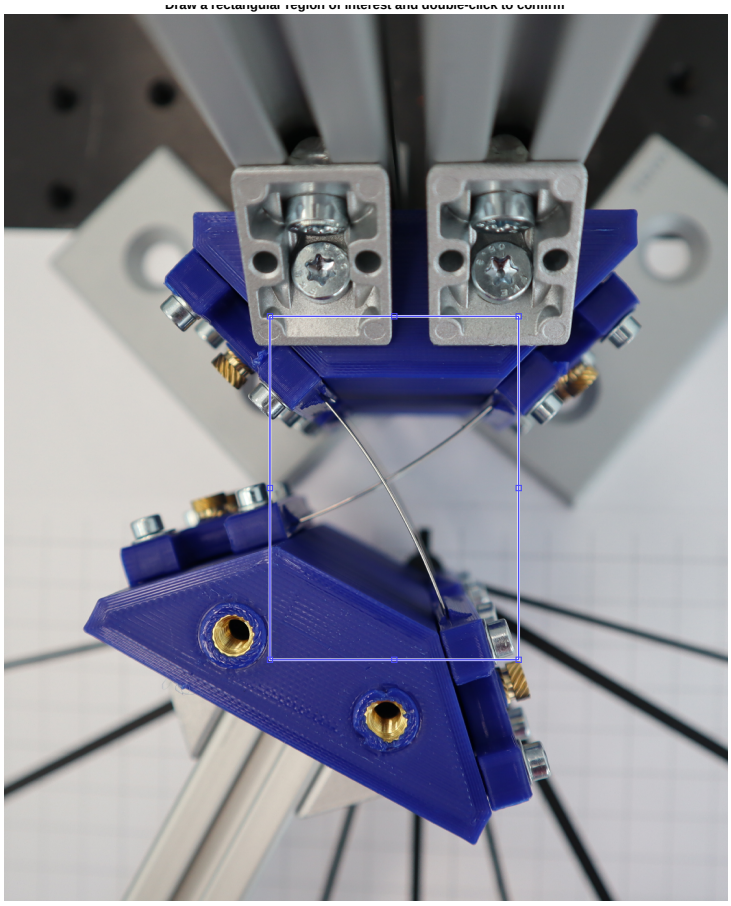
This section outlines the methodology employed to examine the bending of a flexure hinge through image analysis, utilizing MATLAB's Computer Vision Toolbox.

Methodology

- **Image Acquisition and Region of Interest Selection:** An image of the flexure hinge is read into MATLAB. The user selects a rectangular region of interest (ROI) in the image. A mask for the ROI is created and applied to the grayscale version of the image.

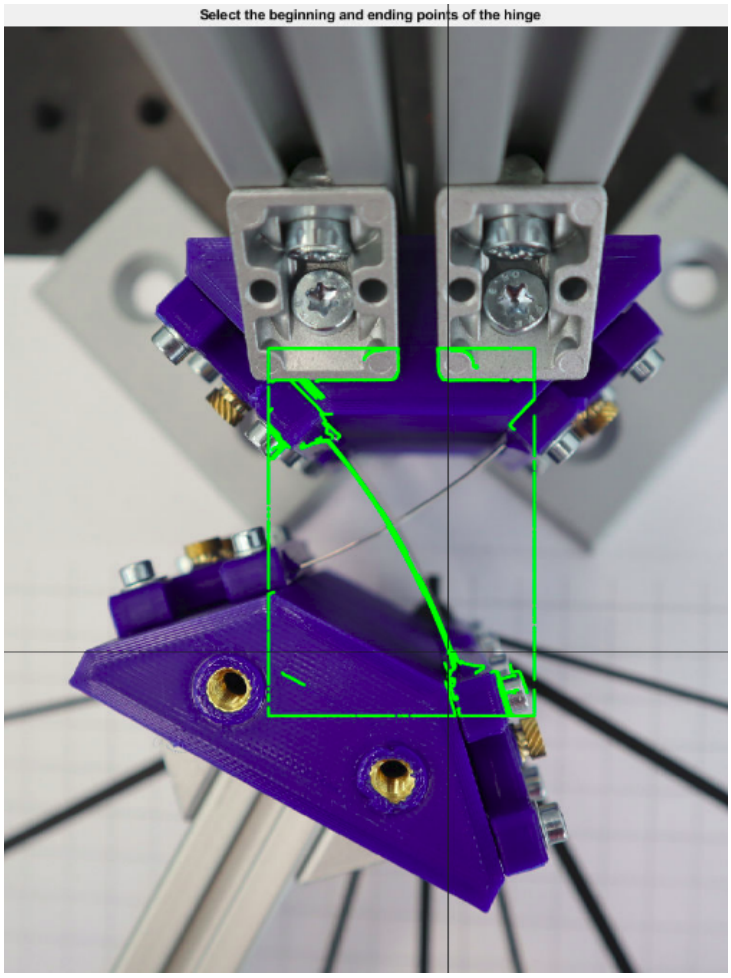
Figure 11

The original image with the selected ROI highlighted. This screenshot shows the user-defined rectangular region on the hinge, illustrating where the analysis is focused.



- **Image Preprocessing:** Noise reduction and contrast enhancement techniques are applied. Morphological operations are used to enhance edges within the image.
- **Edge Detection and Refinement:** Edge detection is performed using the Canny method. Small objects are removed, and holes are filled to refine the detected edges.

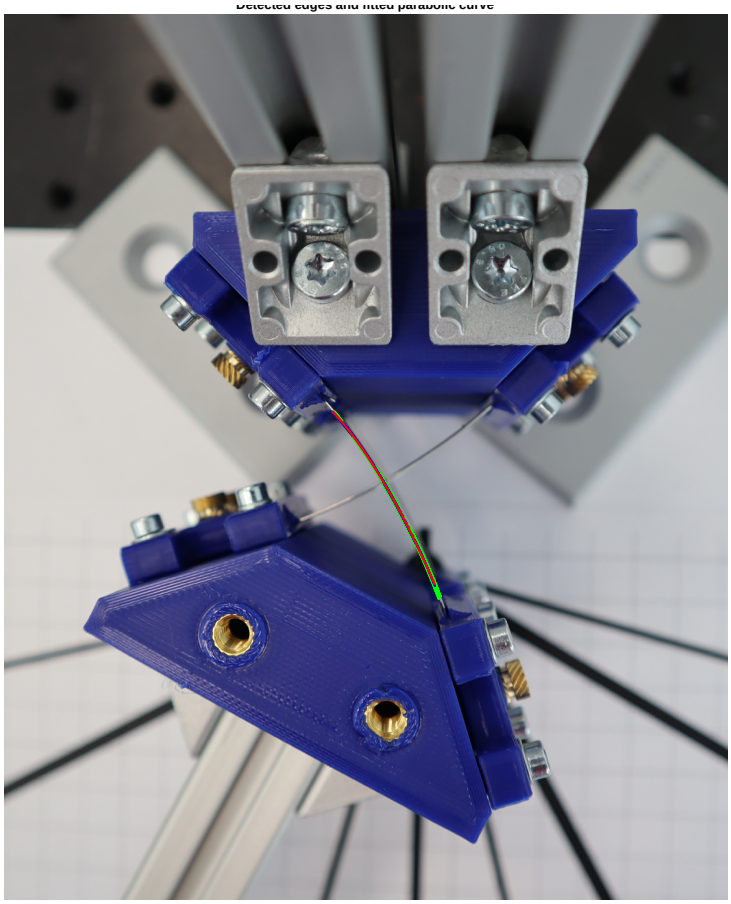
Figure 12
The image showing detected edges after applying the Canny edge detection algorithm. Highlighted edges are visible, with any small, irrelevant objects removed and holes filled.



- **Parabolic Curve Fitting:** Edge points within the user-defined region are selected. A second-degree polynomial (parabolic) fit is performed on the selected edge

coordinates. The inliers of the fit are identified using RANSAC, and the fitted curve points are generated.

Figure 13
The image displaying the detected edge points and the fitted parabolic curve. The screenshot includes the original image with detected edges and the parabolic curve overlaid in red, demonstrating the fit.



- **Bending Parameters Calculation:** Coefficients of the fitted parabola are calculated and displayed. The bending angle at the midpoint of the selected segment is calculated and displayed in degrees.

Figure 14

MATLAB output showing the coefficients of the fitted parabola and the calculated bending angle. This screenshot presents the MATLAB command window or figure displaying these results.

```
Select the beginning point of the hinge
Select the ending point of the hinge
Coefficients of the fitted parabola:
a: 28.3299
b: 269.1177
c: 2200.6967
Calculated Bending Angle: 89.9997 degrees
>>
```

Application of Results

The results obtained from the analysis can be used to validate design specifications, assess mechanical performance, optimize hinge design, ensure quality control, and document performance characteristics in technical reports and project documentation.

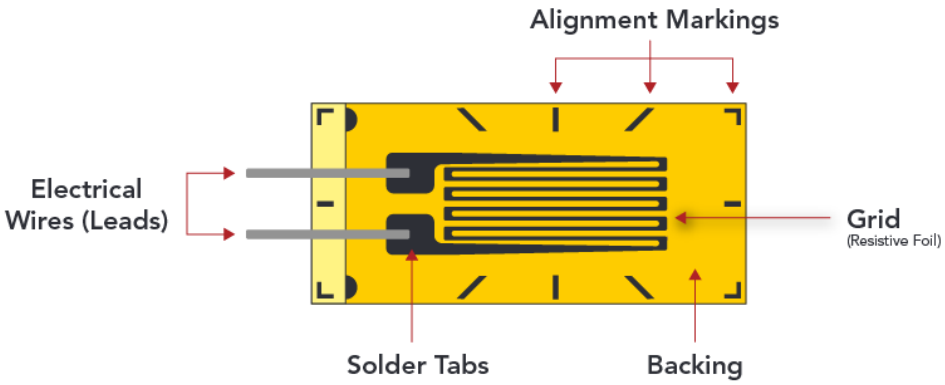
Chapter 0: Strain Gauges in Flexure Hinge Project

Introduction to Strain Gauges

Definition of Strain Gauges

Strain gauges are sensors that measure strain (deformation) in objects by changing their electrical resistance when deformed. They are typically made of fine wire or foil arranged in a grid pattern and bonded to the object's surface. When the object deforms, the strain gauge's resistance changes, which can be measured using a Wheatstone bridge circuit to quantify the strain.

Figure 15
Example of a strain gauge.



Applications of Strain Gauges

Strain gauges are used across various industries for their accuracy in measuring strain:

- **Aerospace:** Monitor the structural integrity of aircraft components, detecting stress concentrations and potential failures.
- **Civil Engineering:** Measure strain in structures like bridges and buildings to assess their health and guide maintenance.
- **Mechanical Engineering:** Test materials and components under load, helping design more robust products.

Strain gauges are also used in load cells, torque sensors, and pressure transducers.

Relevance to Flexure Hinge Project

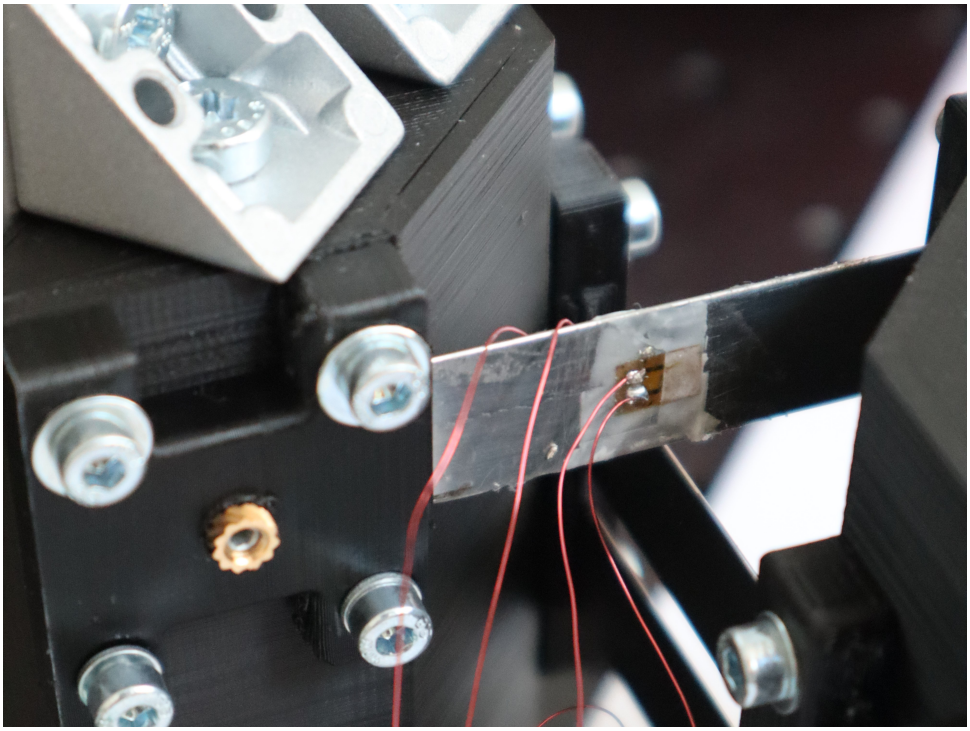
Importance in Measuring Deformation

In the flexure hinge project, strain gauges are crucial for understanding the hinge's deformation and stress characteristics. They provide accurate, real-time data on the strain distribution within the hinge, helping validate the design and ensure it meets specifications. Stress areas and potential failure points can be identified by measuring strain, leading to optimized design and improved material selection.

Integration with Project

Strain gauges are attached to specific locations on the hinge to collect deformation data under various loads. This data is essential for analyzing hinge performance and validating theoretical models. Comparing recorded strain measurements to expected values helps refine the design. Strain gauge data also provides insights into the hinge's behavior under repeated bending cycles, aiding in material selection and design improvements. This integration ensures precise monitoring and contributes to the project's success.

Figure 16
Strain gauge installed on the flexure hinge.



Chapter 0: Experimental Testing of the Flexure Hinge and its Potential Application in Force-Torque Sensors

Introduction

This chapter presents the results of experimental tests conducted on the flexure hinge to validate the theoretical model discussed in previous chapters. The primary focus is on evaluating the hinge's performance under 15 and 30-degree bending angles and comparing MATLAB model predictions with real-life deformation data captured through computer vision analysis. Additionally, strain gauge data is analyzed separately to observe stress responses under these conditions.

The main sections of this chapter are structured as follows:

- Presentation of results for 15-degree and 30-degree bending without commentary.
- A comprehensive comparison of the MATLAB model and real-life computer vision data using visual plots and mathematical values.
- Analysis of strain gauge data in relation to the hinge deformation.

Experimental Setup

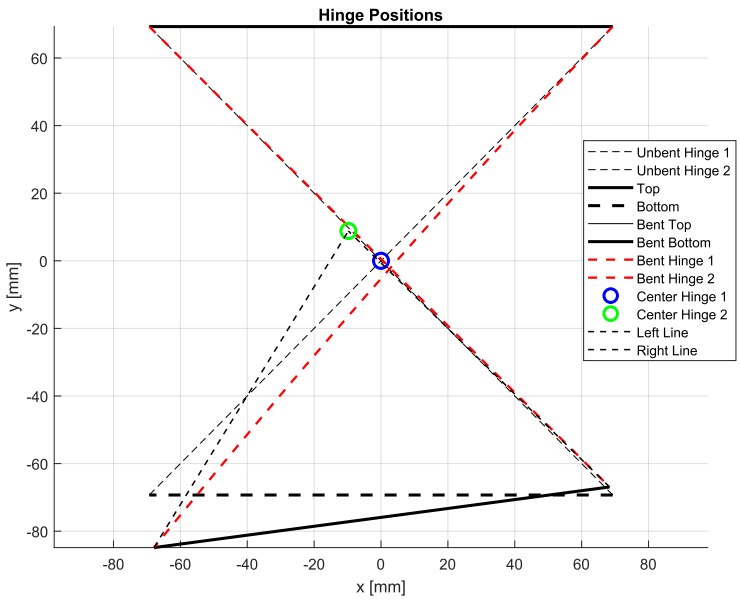
The flexure hinge was subjected to controlled bending at two angles, 15 and 30 degrees. Strain gauges were installed at critical points to capture stress data during the bending. MATLAB simulations were carried out to model the hinge's theoretical deformation, while real-world images were analyzed through computer vision algorithms to detect and quantify the actual deformation.

Case 1: 15-Degree Bending

MATLAB Model Plot

The theoretical results of the flexure hinge under a 15-degree bend, as obtained from the MATLAB simulation, are shown in Figure 17.

Figure 17
MATLAB Model Plot of Flexure Hinge under 15-Degree Bending



Real-Life Computer Vision Plot

The real-life deformation of the flexure hinge at 15 degrees, as captured by computer vision analysis, is shown in Figure 18.

Case 2: 30-Degree Bending

MATLAB Model Plot

The theoretical results of the flexure hinge under a 30-degree bend, as obtained from the MATLAB simulation, are presented in Figure 19.

Figure 18

Real-Life Deformation of Flexure Hinge under 15-Degree Bending (Computer Vision)

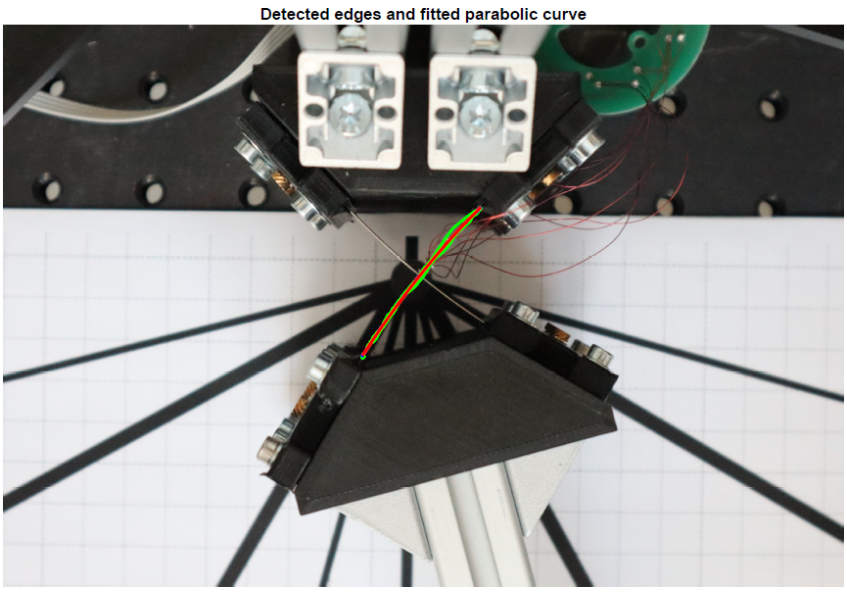
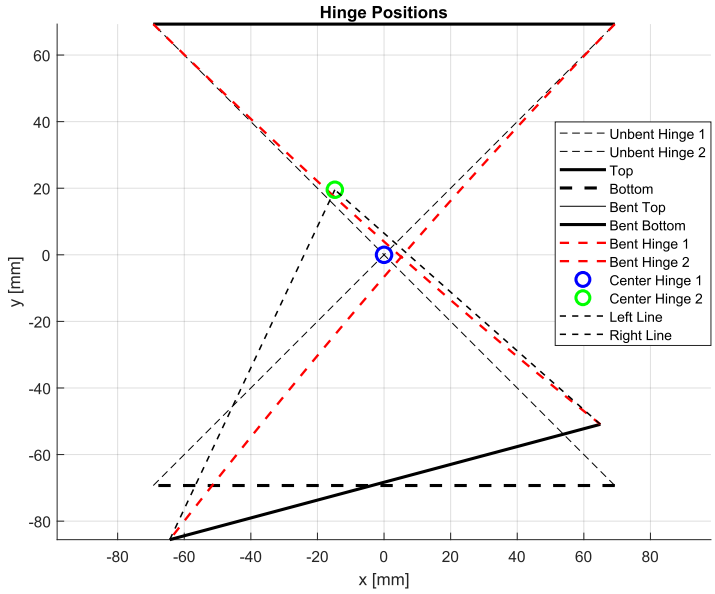


Figure 19

MATLAB Model Plot of Flexure Hinge under 30-Degree Bending

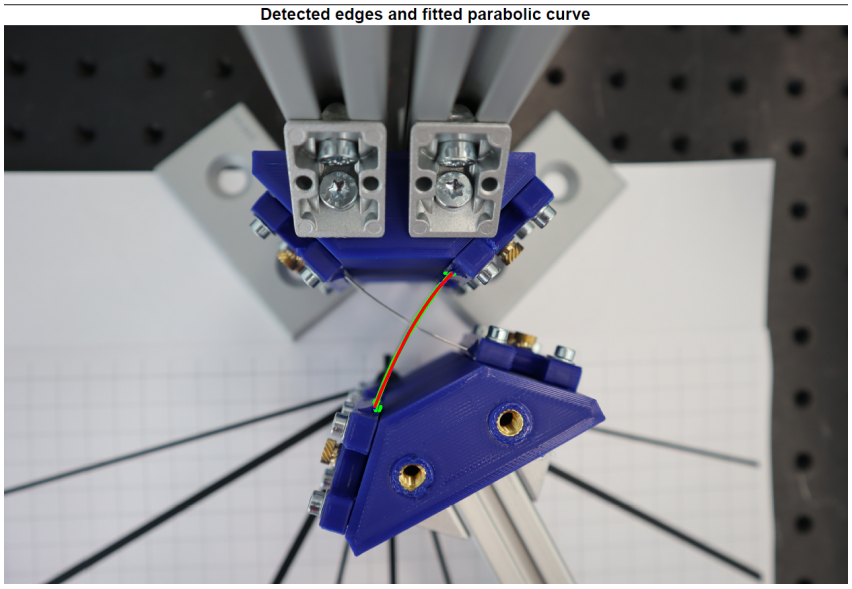


Real-Life Computer Vision Plot

The real-life deformation of the flexure hinge at 30 degrees, as captured by computer vision analysis, is shown in Figure 20.

Figure 20

Real-Life Deformation of Flexure Hinge under 30-Degree Bending (Computer Vision)



Comparison and Discussion

When comparing the MATLAB model outputs with the real-world data obtained from the computer vision analysis for 15-degree- and 30-degree bending cases, several insights emerge, along with some expected discrepancies due to the inherent differences between theoretical simulations and real-life conditions.

The visual comparison of the MATLAB model and computer vision data shows that both methods capture the overall behavior of the flexure hinge under different bending angles. However, when comparing the parabola coefficients, significant differences arise. In the 15-degree case, the MATLAB model's coefficients ($a_2 = -0.000700$, $b_2 = 1.124701$, $c_2 = -5.280370$) differ substantially from the computer vision-derived coefficients ($a = 19.1247$, $b = -262.1467$, $c = 1880.9937$). Similarly, in the 30-degree case, the MATLAB model gives $a_2 = -0.000981$, $b_2 = 1.163584$, and $c_2 = -6.626902$, while the computer vision data provides $a = 31.0035$, $b = -265.3983$, and $c = 2164.05$.

These differences in parabola coefficients can be attributed to several factors. First, while the MATLAB model is based on fundamental principles and assumes ideal

conditions, real-world scenarios introduce imperfections such as material irregularities and external factors that are not fully accounted for in the simulation. Furthermore, the iterative Newton-Raphson method used in the model to converge on a solution, though effective, adds computational complexity, which may lead to small deviations when compared with real-life behavior.

On the other hand, the computer vision method relies on detecting the edges of the bent hinge, and this process is sensitive to image quality, lighting conditions, and the precision of the curve-fitting algorithm. Given that perfect edge detection is challenging in practice, the coefficients from the real-life data are expected to not perfectly match those from the theoretical model. Despite these differences in numerical values, the general shape and behavior of the bent hinge are similar across both methods, indicating that the overall deformation pattern is captured by both the MATLAB simulation and the computer vision approach.

Moreover, the MATLAB model provides additional insights into the forces and moments acting on the hinge, such as m_{B1}, m_{B2}, f_1, and f_2, which cannot be easily obtained through visual data alone. These forces and moments give a deeper understanding of the stresses and mechanical behavior of the flexure hinge, particularly in relation to its potential application in force-torque sensors.

Despite the discrepancies in the parabola coefficients, the overall comparison shows that the MATLAB model and the real-world computer vision data provide valuable, complementary insights. With its analytical approach, the MATLAB model offers detailed information about the underlying physics of the hinge's deformation, while the computer vision analysis validates these predictions through real-life observation. The slight differences between the two can be attributed to the simplifications inherent in the model and the practical challenges of real-world data collection.

In summary, despite the differences in coefficients, the visual and mathematical consistency between the two methods suggests that the flexure hinge model is robust and captures the key characteristics of the hinge's deformation. The additional data the MATLAB model provides, such as forces, moments, and stress distributions, further enhance the understanding of the flexure hinge's performance under bending, making it a valuable tool for applications in force-torque sensors.

Chapter 0: Robot Bases

Introduction

This chapter presents the design and evaluation of a robotic base for the Franka Emika Panda arm, incorporating a 6 DoF FTE-Omega-160-IP60 SI 1000-120 force-torque sensor and Siemens SIMATIC SCADA Systems. The goal is to analyze the base's mechanical behavior under various conditions, particularly focusing on the measurement of both force and moment data. While this chapter outlines the design process and experimental setup, the actual execution of both simulated and real-world experiments was not completed due to technical difficulties and time constraints during the internship.

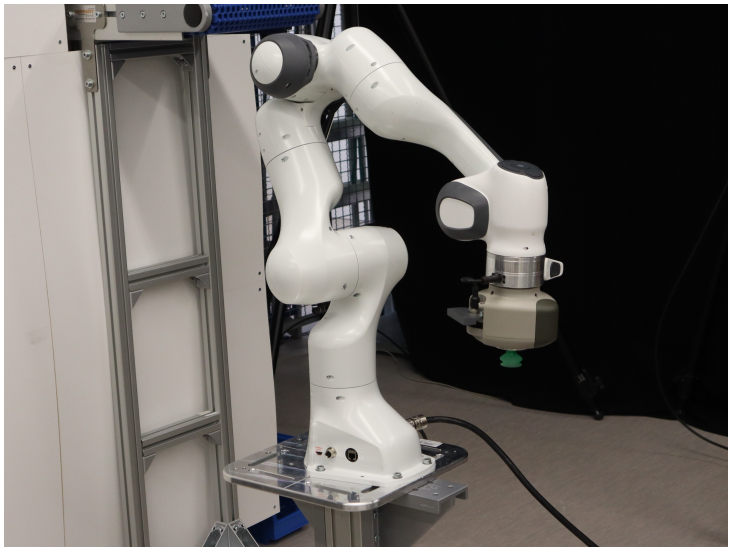
Despite these setbacks, the necessary components and equipment are in place, and the simulations were partially set up using a digital twin in Gazebo. However, technical issues prevented the full completion of these simulations. Likewise, the real-life experiments could not be initiated due to a lack of time. This chapter will describe the intended experiments and explain the challenges encountered.

Equipment Overview

The experimental setup includes several critical components:

- **Franka Emika Panda**: A highly dexterous 7-DoF robotic arm designed for various manipulation tasks. This arm is part of the planned experiments to apply forces and moments to the base.

Figure 21
Franka Emika Panda robotic arm.



- **6 DoF FTE-Omega-160-IP60 SI 1000-120 Force Torque Sensor**: A precision sensor designed to measure forces and torques in six degrees of freedom, which is central to the experiment's goal of measuring interactions between the robot arm and the base.

Figure 22
6 DoF FTE-Omega-160-IP60 SI 1000-120 force torque sensor.

- **Robotnik Kairos**: A mobile robotic platform designed to support robotic arms such as the Franka Emika Panda. It was planned to be used for dynamic testing, but the mobile-based experiments were not initiated.

Figure 23
Robotnik Kairos mobile base.

- **Siemens SIMATIC SCADA Systems**: Integrated with the Franka Emika Panda to perform additional acceleration measurements, this system provides real-time data on the dynamics of the robotic arm during operations.

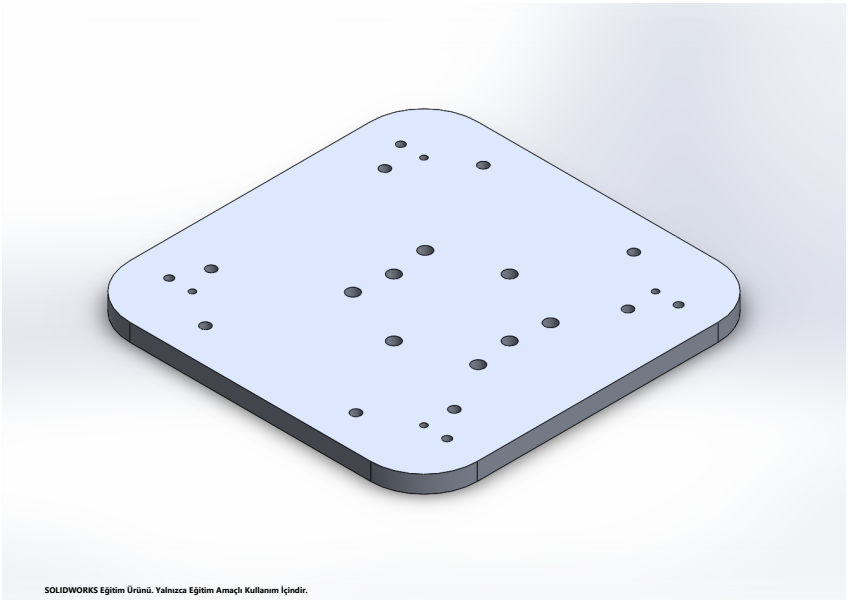
Figure 24
Siemens SIMATIC SCADA Systems.

Necessity of an Adapter Plate

A custom adapter plate was designed to integrate these components into a cohesive experimental setup. The adapter plate interfaces the force-torque sensor, the profile base, and the Franka Emika Panda arm. The design of the plate ensures a secure and precise connection, enabling accurate force and moment measurements, although the experiments were not conducted.

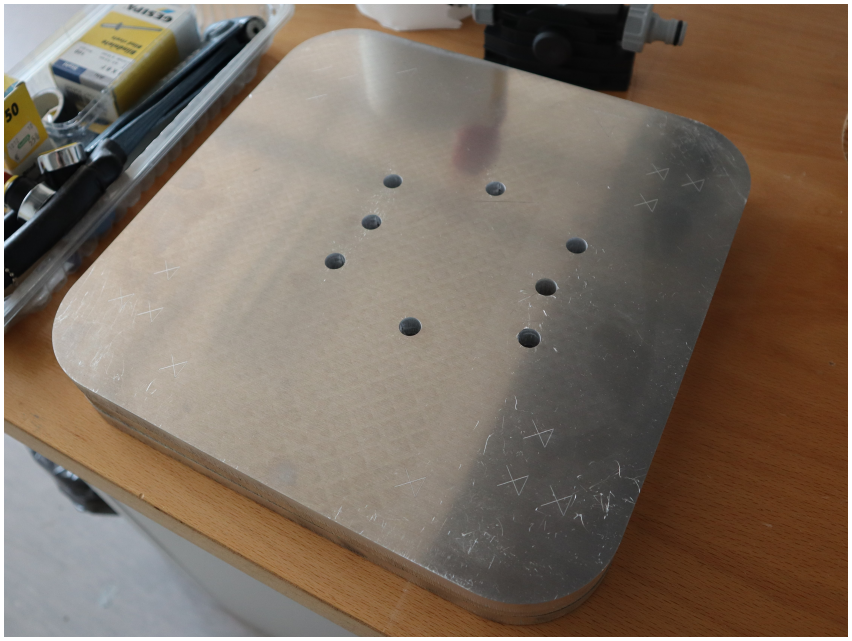
- **Adapter Plate Design**: The plate was designed using CAD software to meet the specific requirements of the setup. It accommodates the sensor's dimensions and provides robust support for the Franka Emika Panda.

Figure 25
CAD model of the custom-designed adapter plate.



- **Manufacturing of the Adapter Plate**: The plate was manufactured from steel to withstand the expected forces and moments. The plate was ready for installation, but real-world experiments could not be initiated.

Figure 26
Manufactured adapter plate, ready for installation.



Simulation with Gazebo

Creation of the Digital Twin

To prepare for the simulation experiments, a digital twin of the experimental setup was created. This twin includes the base, the force-torque sensor, and the Franka Emika Panda arm. While the digital twin was successfully developed using CAD software, the simulations could not be fully executed due to technical difficulties with the simulation environment.

URDF File Development

The next step was to convert the digital twin into a Universal Robot Description Format (URDF) file for use in Gazebo. The URDF file was successfully generated, and preliminary tests in Gazebo showed that the model loaded correctly. However, further technical issues prevented the completion of the simulation experiments.

Figure 27
CAD model of the digital twin, showing the entire experimental setup.

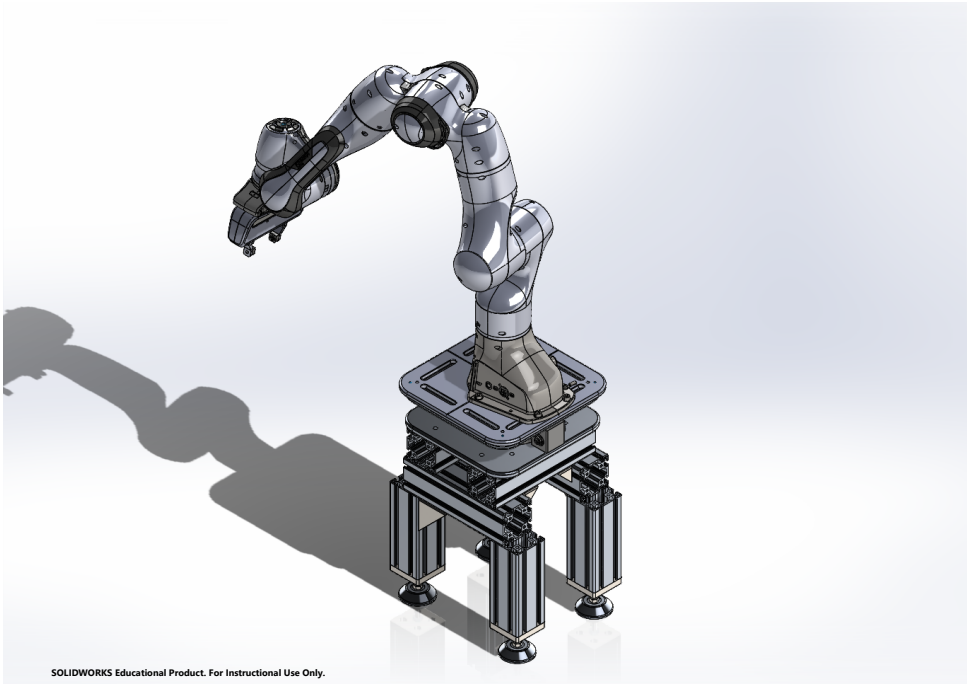
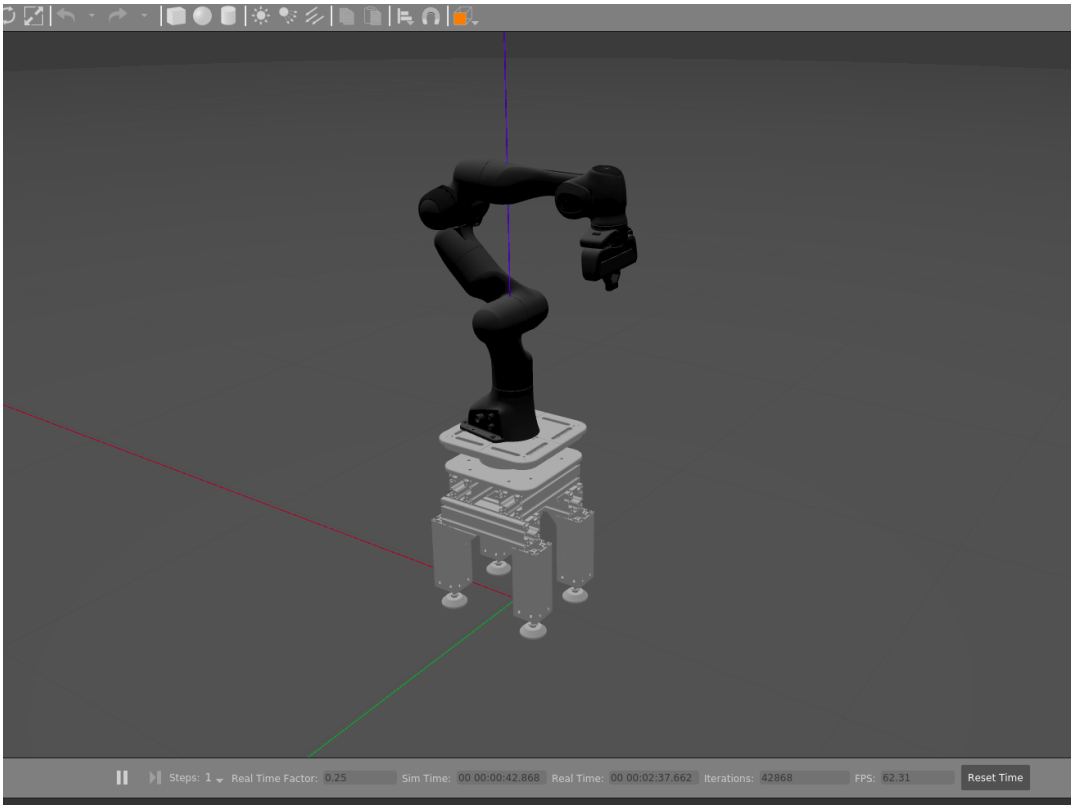


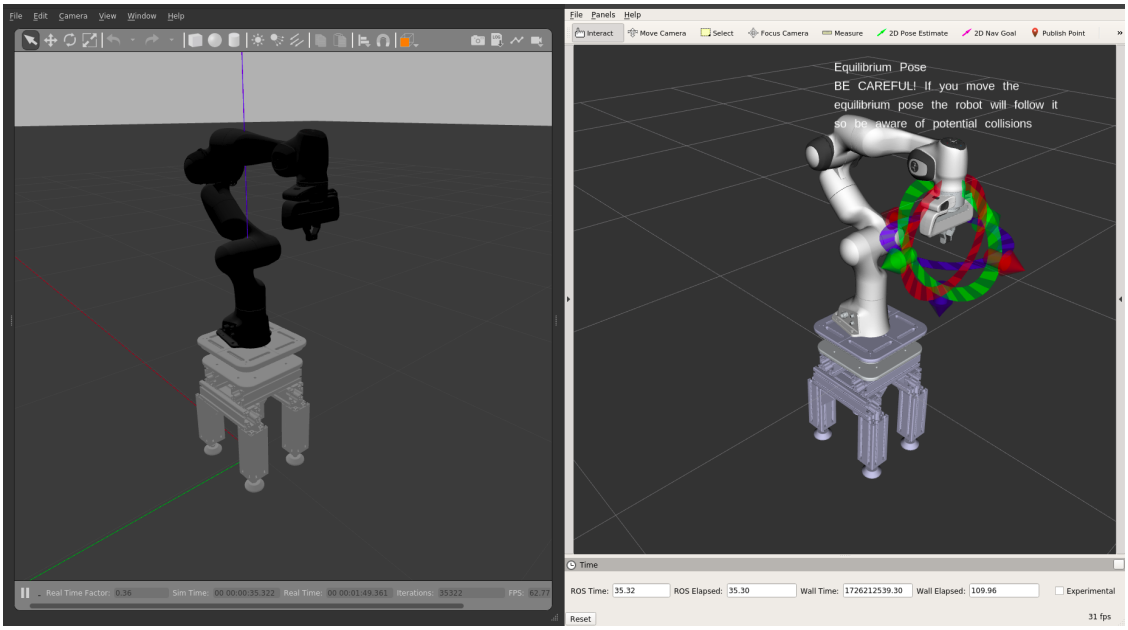
Figure 28
URDF model of the experimental setup in Gazebo.



Simulation Experiments

Although the simulations were prepared, technical issues with Gazebo prevented the full completion of the experiments. The intended simulations aimed to evaluate the stability and dynamic response of the base under different load conditions and trajectories, as shown in the preliminary tests below.

Figure 29
Simulation showing the base and arm in motion.



Real-World Experiments

Static Base Setup

In the real-world experiments, the static base, built from Item profiles, was intended to mount the 6 DoF force-torque sensor and the Franka Emika Panda arm.

Unfortunately, due to time constraints during the internship, the static experiments could not be initiated. The setup was prepared, but testing was not performed.

Mobile Base Setup with Robotnik Kairos

Plans for a mobile base setup using the Robotnik Kairos platform were in place. The mobile base was prepared for experiments, but due to time limitations, real-world experiments involving the mobile platform were not carried out.

Conclusion

While the full set of experiments could not be completed during the internship, the necessary groundwork has been laid for both simulated and real-world testing. The components, equipment, and setup are ready for future experiments to evaluate the performance of the robotic base. The design of the adapter plate and digital twin development provide a solid foundation for future work. Completing the experiments in both static and dynamic setups will be crucial to understanding the force and moment behavior of the system. Overcoming the technical difficulties and time constraints will allow for a more comprehensive evaluation of the robotic base's capabilities.

Chapter 0: Discussion

Chapter 0: Conclusion

Strain Gauge Installation Manual

Required Materials

Before you begin the installation, gather all necessary materials. This will ensure a smooth process without interruptions.

Materials Needed:

- Strain gauges
- Special adhesive for strain gauges
- Sandpaper
- Isopropyl alcohol
- Clean cloth or wipes
- Tape
- Soldering iron and solder
- Connecting wires
- Weights or clamps for applying pressure during curing
- Gloves
- Optional: 3D printed jig for precise positioning

Surface Preparation

Proper surface preparation is crucial for a successful strain gauge installation. This step ensures that the adhesive bonds well to the surface.

Surface Sanding. Start by sanding the surface where the strain gauge will be applied. Use progressively finer grits of sandpaper until the surface is smooth and free of any imperfections. Ensure the sanding is done evenly to avoid uneven surfaces.

Figure 30
All necessary materials laid out.

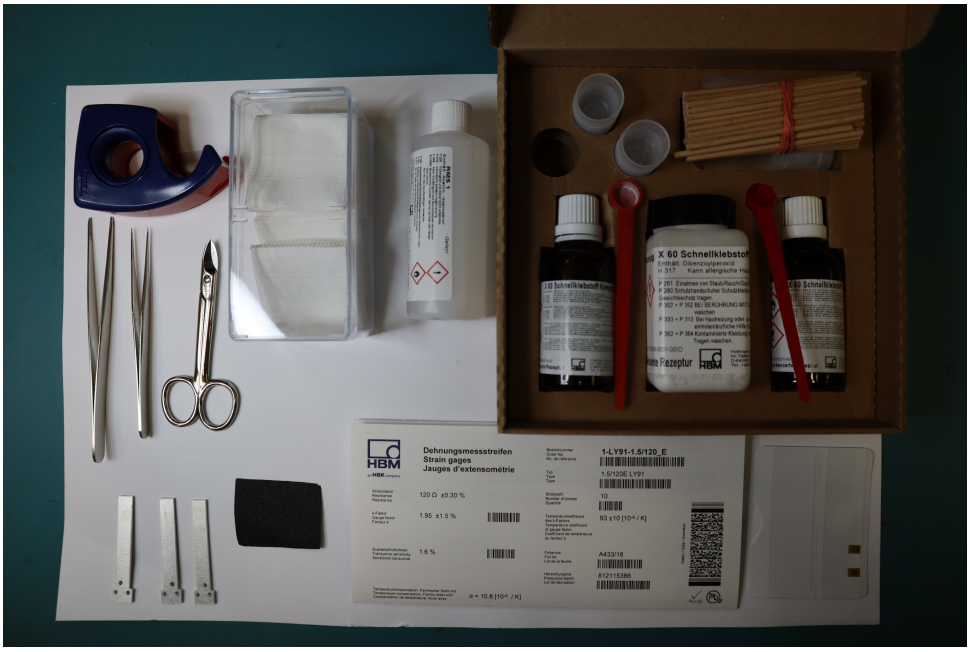


Figure 31
Sanding the surface.

Cleaning with Alcohol. After sanding, clean the surface thoroughly with isopropyl alcohol. This will remove any dust, oils, or other contaminants. Use a clean cloth or wipe to apply the alcohol and ensure the surface is completely dry before proceeding.

Strain Gauge Positioning

Accurate positioning of the strain gauge is critical for precise measurements.

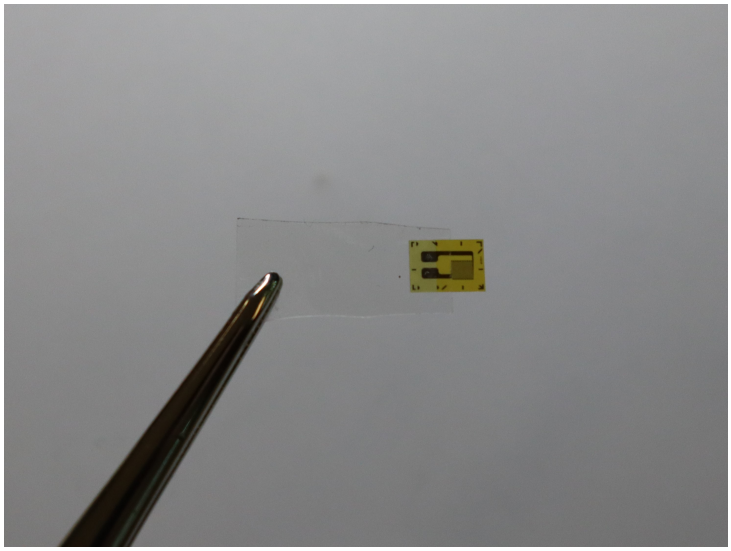
Using Tape.

Figure 32
Cleaning the surface with alcohol.

- Place the strain gauge on the prepared surface. Use a piece of tape to hold it in position. Ensure the gauge is aligned correctly with the measurement axis.
- **Warning:** Cover the surface outside an area twice the length of the strain gauge with tape to prevent the adhesive from spreading over the entire surface.
- Smooth out the tape to eliminate any air bubbles.
- **Warning:** When securing the strain gauge with tape, ensure the tape covers the soldering terminals to prevent the adhesive from covering these critical areas.

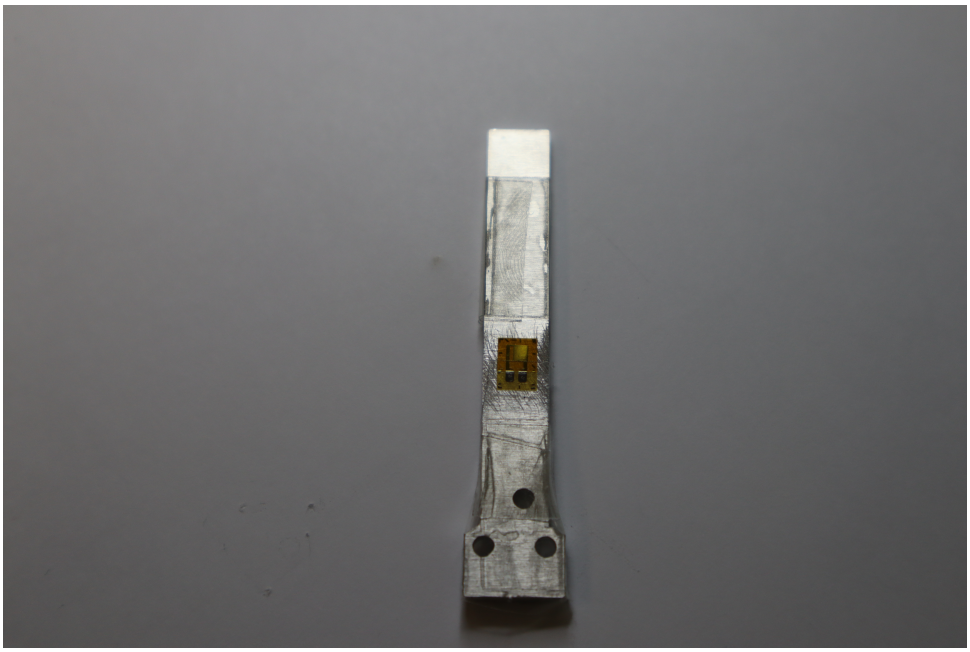
Figure 33
Covering beam with tape

Figure 34
Positioning the strain gauge with tape.



Using a 3D Printed Jig (Optional). If available, use a 3D-printed jig to position the strain gauge. The jig will help you place the gauge accurately and consistently. Secure the gauge in the jig and align it with the desired position on the surface.

Figure 35
Positioning the strain gauge (with a 3D printed jig if applicable).



Adhesive Preparation

Follow these steps to prepare the adhesive:

- Take one teaspoon of the dry component.
- Add 20 drops of the liquid component.
- Mix thoroughly with a wooden stick for 30 seconds to 1 minute until the adhesive is uniform in color and consistency.

Figure 36
Adhesive Kit.

Figure 37
Mixing the adhesive components.

Application of Adhesive and Curing

Now that the adhesive is prepared follow these steps:

- Apply a thin, even adhesive layer to the back of the strain gauge or directly onto the prepared surface.
- Carefully place the strain gauge onto the surface, ensuring it is aligned correctly.
- Apply another thin layer of adhesive to the top of the strain gauge after positioning it on the surface.
- Carefully place a thin plastic film over the strain gauge to protect it and ensure even pressure distribution.
- Apply pressure using weights or clamps to hold the gauge during curing. Follow the adhesive manufacturer's recommended curing time and conditions.
- **Note:** To check if the adhesive has fully cured, observe the excess mixture left on the lid—if it's dry, the adhesive on the strain gauge is likely cured as well.

Figure 38
Applying adhesive to bottom

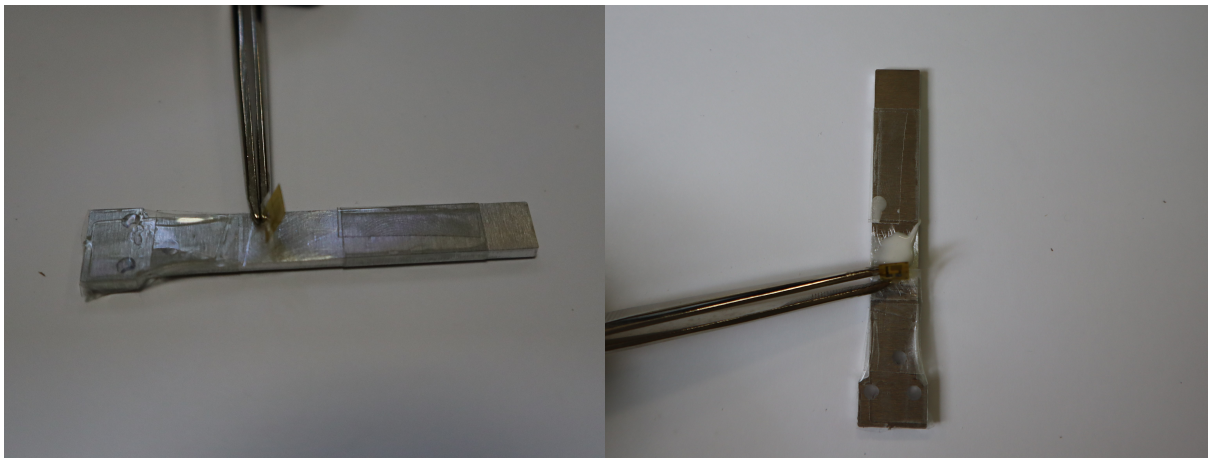


Figure 39
Applying adhesive to top and place plastic film

Figure 40
Applying pressure

After applying pressure, allow the adhesive to cure according to the manufacturer's instructions. Once the adhesive has fully cured, carefully remove the tape and plastic film.

The strain gauge should now be securely attached to the surface. Ensure the gauge is firmly bonded and no adhesive has seeped into unwanted areas.

Soldering the Strain Gauges

Once the adhesive has fully cured, proceed with soldering:

Figure 41
Strain gauge after curing and tape removal.

- Carefully solder the connecting wires to the strain gauge terminals. Ensure the solder joints are clean and strong to avoid signal loss.
- Use a minimal amount of solder to avoid damaging the gauge or affecting its sensitivity.

Conclusion

Your strain gauge installation is now complete! Before starting your measurements, testing the setup to ensure everything functions correctly is important. You can do this by measuring the resistance between the nodes on the PCB or the wires connected to the strain gauge.

Each strain gauge typically has a resistance of approximately 120 ohms. The strain gauges are arranged in a Wheatstone bridge configuration when connected to the PCB. Measure the resistance at different points to ensure the values align with this setup. Make sure to check both the PCB and the wires for correct connections and continuity.

If the resistance values are correct, your strain gauge installation is ready for use.

Chapter 0: Bibliography

Gunnink, K., Aarts, R., & Brouwer, D. (2013). Performance optimization of large stroke flexure hinges for high stiffness and eigenfrequency. *Precision Engineering*, *37*(4), 903–910. https://doi.org/10.1016/j.precisioneng.2013.04.007

Wittrick, W. H. (1951). The properties of crossed flexure pivots and the influence of the point at which the strips cross. *Aeronautical Quarterly*, *2*, 272–292.

Zelenika, S., & Bona, F. D. (2002). Analytical and experimental characterisation of high-precision flexural pivots subjected to lateral loads. *Precision Engineering*, *26*(2), 381–388.