Designing a Large Range of Motion and Variable Stiffness Infant Neck Surrogate for Shaking Trauma Studies

Yaro Marsille
4658264
Department of Biomedical Engineering, Delft University of Technology

Supervisors: Arjo Loeve & Kim Hutchinson

TU Delft

Designing a Large Range of Motion and Variable Stiffness Infant Neck Surrogate for Shaking Trauma Studies

By

Y. Marsille

In partial fulfilment of the requirements for the degree of:

Master of Science
in Biomedical Engineering

at the Delft University of Technology,
to be defended publicly on Wednesday June 11, 2025 at 4:00 PM.

Supervisor:	dr. ir. A. Loeve (chair),	TU Delft
Thesis committee:	ir. K. Hutchinson,	TU Delft
	Prof. dr. J. Dankelman,	TU Delft

Abstract

Every year, 14 to 41 cases per 100,000 infants under 1 year old are diagnosed with inflicted head injury (IHI), primarily resulting from shaking trauma (IHI-ST) or blunt force trauma. Without reliable structural bending stiffness data of the infant's neck, injury predictions using physical infant surrogates in shaking simulations remain highly uncertain, undermining both forensic and preventive studies. Due to ethical constraints, biomechanical properties of infant necks are scarce, limiting the biofidelity and validation possibilities for existing infant surrogates, such as anthropomorphic test dummies (ATDs). Currently, infant neck surrogates suffer from inadequate biofidelity concerning stiffness and validated range of motion, necessary to accurately simulate shaking trauma simulations.

This thesis aims to address the gap by exploring the design of a durable, adjustable stiffness surrogate neck, improving the accuracy of shaking experiments. Experimental stiffness values obtained from functional spine units (FSU) by Luck et al., extrapolated by Sullivan et al. were used as target values, suggesting stiffness ranges of 0.2 Nm/rad in flexion and 0.4 Nm/rad in extension for a 1.5-month-old infant. The design aims for a 90-degree ROM in flexion and extension, essential for accurate simulation of chin-to-chest and occiput-to-back contacts, both critical for assessing injury mechanisms.

A compliant monolithic hinge mechanism from Fowler et al. was proposed as the core mechanism, able to achieve large angular displacements through flexures. Finite element analysis was performed to optimize material and geometric parameters. Parametric modeling in ABAQUS identified the relationships between stiffness, stress, material properties, and hinge geometry. Based on these relationships, feasible prototype geometries were extracted, varying in flexure thickness, length, and width. Manufacturing was done using 3D printing with polylactic acid (PLA) and carbon fiber-reinforced polyethylene terephthalate glycol (PETG-CF).

Static experimental validation demonstrated achievable stiffness values at the lower boundary of target ranges while the upper bound was not reached, primarily due to anisotropy from 3D printing and material limitations. Despite these limitations, prototypes successfully reached the targeted ROM. Future research should incorporate dynamic testing to validate durability and head kinematics and should consider multi-degree-of-freedom designs to fully replicate infant neck biomechanics. Further challenges remain in replicating infant neck viscoelasticity and obtaining experimental infant data for validation.

Table Of Contents

1 Introduction **5**
1.1 Program of Requirements 7

2 Methods and Materials **7**
2.1 Concept Phase 7
2.2 Monolithic Hinge Optimization Phase 8
2.3 Experiment Phase 9

3 Results **11**
3.1 ABAQUS Model Validation 11
3.2 Design Space 11
3.3 Material and geometry selection 12
3.4 Experimental results 13

4 Discussion **15**
4.1 Biofidelity 15
4.2 ABAQUS Model Validation 16
4.3 Material selection 16
4.4 Comparison computational and experimental results 16
4.5 Data Analysis 17
4.6 Future Research and Recommendations 17

5 Conclusion **18**

References **19**

Appendix **22**
A. Concept Selection 22
B. Abaqus Code 24
C. Mesh validation 29
D. Simulation results 30
E. Python code Maximum stress regression 36
F. Python code Stiffness regression 40
G. Data analysis 42
H. Investigated materials before manufacturing method 45
I. Investigated materials after manufacturing method 45
J. Material and geometry selection charts 46
K. ABAQUS results prototypes 51
L. Moment calculations 53
M. Experimental results for flexion/extension (FE) and lateral bending (LB) tests 55

1 Introduction

Every year, 14 to 41 cases per 100.000 infants under 1 year old are diagnosed with inflicted head injury (IHI), also known as abusive head trauma or non-accidental head injury. [1-5] The main causes for IHI are shaking trauma or blunt force trauma. Shaking trauma is caused by violently shaking the infant, causing accelerations and decelerations of the head. [6] Symptoms of inflicted head injury by shaking trauma (IHI-ST) include subdural hematomas, diffuse axonal injury, retinal hemorrhage and neck injuries. [7,8] Understanding the head-neck dynamics of infants is important in researching injuries associated with violently shaking an infant, as increased accelerations can be observed during shaking, with maximum accelerations occurring at chin-to-chest or occiput-to-back contact. [9]

Compared to adults, infants have a larger head-to-body ratio and weaker neck muscles, which leads to lower resistance to violent movements. Furthermore, the vertebrae in infants are mainly cartilaginous, combined with weaker ligaments and smaller articular facets compared to adults, overall leading to a more flexible neck structure compared to adults. [10] As biomechanical tests on infants are limited due to ethical considerations, simulation models are used to research the consequences of violently shaking an infant. These models consist of physical models using mechanical surrogates, mathematical models using finite element methods or rigid body models, and animal models. These models often rely on scaled adult material properties, to which it is unknown how comparable these scaled properties are to infant properties. [9]

Movements of the neck can be described by flexion/extension (FE), the movement in the sagittal plane, lateral bending (LB), the movement in the frontal plane and axial rotation (AR), the rotation of the cervical spine along its axis, as seen in Figure 1, plus axial compression/extension caused by antero-superior loading. [11] The cervical spine consists of the vertebrae C1-C7 and the additional musculature and ligamentous structures. [12] To describe the full range of motion (ROM) of the neck in flexion/extension, both active and passive ROM needs to be addressed. Passive ROM is the ROM obtained from an external force, while active ROM is the ROM obtained by muscle movements. [14] Looking into stiffness, a distinguishing can be made in active and passive bending stiffness, where active bending stiffness is the stiffness where the infant exerts a muscle force, while the passive bending stiffness is the structural bending stiffness of the neck with minimal muscle activation.
As IHI-ST occurs mainly by shaking in the sagittal plane resulting from an external force and the premature neck musculature cannot actively influence the head movements, as well as loss of consciousness occurring [14], the passive ROM and passive stiffness values in the sagittal plane are the most relevant to obtain to improve shaking simulations in infants.

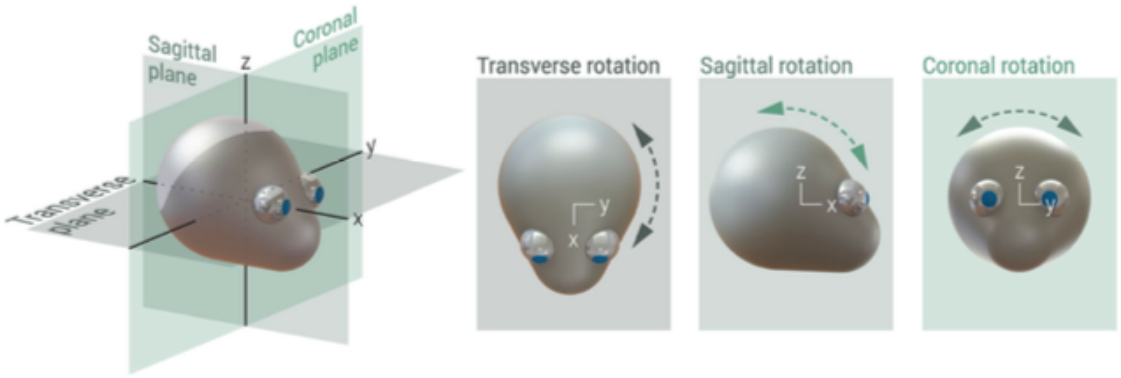

Figure 1: Planes of the head with motions, adopted from Vester et al (2019) [11]

Designing a biofidelic surrogate neck requires experimental data on infant biomechanics, especially for the range of motion and passive stiffness of the neck. A systematic literature study was performed to find and assess experimental methods for obtaining range of motion and stiffness, infant neck biomechanical data and infant neck surrogates. [15] Biomechanics of the whole cervical spine (WCS), as well as head-neck complexes as individual motion segments indicated as functional spine units (FSU) were found in this review.

Key findings were that no experimental data was found for the structural bending stiffness for the WCS in living infants, and passive stiffness in flexion and extension was only measured in one study for cadaveric FSUs [16]. As these stiffnesses represent separate motion segments, separate stiffnesses were extrapolated by Sullivan et al. [17] to obtain an estimate of the structural bending stiffness for the WCS, resulting in estimated neck bending stiffness values ranging from 2.6 to 4.3 mNm/° (0.15 to 0.25 Nm/rad) in flexion and 4.4 to 7.2 mNm/° (0.25 to 0.41 Nm/rad) in extension for a 1.5-month-old infant's cervical spine. Studies investigating the ROM were scarce, with two studies measuring the lateral bending and axial rotation range of motion in living infants [18][19], while one study measured the flexion/extension ROM in FSUs [16]. Ohman et al used protractors to measure the lateral bending and axial rotation ROM [18], while Castle et al. used two- and three-dimensional video methods, both while the researcher applied motion to the infant's head [19]. Lateral bending range of motion values stated are 70 degrees according to Ohman et al., while Castle et al. state a range between 10 and 21 degrees based on the video analysis method. Luck et al. estimate the range of motion in flexion/extension by extrapolating FSU range of motions during tensile and bending tests, using a servo-hydraulic testing system, starting at 175 degrees for an 1.5 month old to 150 degrees at 1 year old [16].

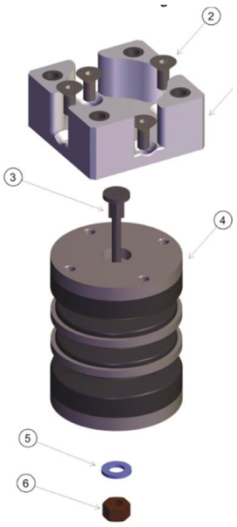

Figure 2: Q0 dummy neck [20]; 1: Accelerometer mount, 2: Screws M4, 3: Tension cable, 4: Molded neck, 5: Washer M5, 6: Hex nut M5

Figure 3: Surrogate neck by Sullivan et al. [17]; 1: Silicone body, 2: Double neoprene rubber bands, 3: Threaded connector, 4: Tygon tubing

When looking into infant surrogates, anthropomorphic test dummies (ATDs) like the Q0 ATD, the Aprica 2.5 ATD and the CRABI ATD are used in current research [14][20][21]. However, as these dummies are commercial car crash-test dummies, the stiffness of the neck is generally experienced to be too large to represent an infant. The structural bending stiffness was only found for the CRABI ATD neck, stating a stiffness of 17 Nm/rad [21]. Furthermore, by using the Q0 dummy in shaking simulations, chin-to-chest and occiput-to-back acceleration data is not accurate, as the neck (see Figure 2) is validated for a range of motion till 73.5 degrees, which is smaller than the contact angle of 90 degrees [22]. As maximum accelerations occur during chin-to-chest (flexion) or occiput-to-back (extension) contact, these accelerations cannot be obtained accurately, which will lead to inaccurate injury assessment. This makes the use of these ATDs in shaking simulations debatable, as these accelerations are believed to contribute the most to injuries [6].

A surrogate neck based on experimental stiffness data by Luck et al. [16] was designed by Sullivan et al. [17]. The neck design of Sullivan et al. can be seen in Figure 3. While this is a step forward regarding biofidelity compared to ATD necks, as this neck is designed based on experimental data instead of scaled adult data, the neck only approximates the neck stiffness while the range of motion is insufficient. Furthermore, this neck makes use of silicon and rubber materials which are unfavorable due to their rapid degradation.

All in all, to improve the accuracy of physical shaking simulations and therefore give a better insight into injuries in infants, the surrogate neck used needs to be improved. As experimental data on the bending stiffness of the infant's neck is uncertain, it is favorable to obtain a surrogate neck capable of varying in stiffness, so it can be updated quickly when data advancements are made, or to match properties at a certain age.

Therefore, the objective of this thesis project was to explore the design of an infant neck that is durable, has a realistic range of motion, and variable stiffness for use in shaking experiments.

1.1 Program of Requirements

To perform improved shaking simulation studies using the Q0 dummy, a neck will be designed based on experimental data. As this data is still uncertain due to the lack of data available, the neck will be designed to be adaptable in stiffness to rapidly adapt if advancements in knowledge are made. Furthermore, by making the stiffness adaptable, the neck can be tuned to mimic an infant's neck at different ages. This is particularly important as the research of Luck et al. [16] show significant changes in properties in the first year of life. To ensure testability using the Q0 dummy, the surrogate must geometrically match the Q0's neck mounting points on both the torso and head.
Research has shown that the shaking motion mostly acts in the sagittal plane [14]. Also, the neck can only be validated against flexion/extension data as lateral bending and axial rotation data is not present. Therefore, the target stiffness for this design will be the extrapolated FSU stiffness and range of motion data obtained by Luck et al and Sullivan et al. [16, 17] as this is the only experimental data available for infants. Additionally, as accelerations during chin-to-chest and occiput-to-back contact contribute significantly to injuries, a sufficiently high range of motion is needed to allow contact. Therefore, the main objective is to design a neck having a validated stiffness over the full range of motion in flexion/extension.
In short, the following design requirements for a surrogate neck have been drafted:

R1) Must match Q0 mounting point geometry to allow for testing using Q0 dummy.
R2) Must be variable in stiffness below 17 Nm/rad, by tuning or by modular parts to enable adaptation to age or data updates
R3) Must consist of materials that do not age or plastically deform under cyclic loading
R4) Must have one rotational degrees of freedom to allow for motion in flexion/extension (FE)
  i) Can be extended to more DOFs while always allowing for FE
R5) Must mimic the characteristics of infant necks by:
  i) Approximating a minimal structural bending stiffness of 0.2 Nm/rad in flexion and 0.4 Nm/rad in extension
  ii) Having a sufficient high range of motion of 180 degrees in flexion/extension to allow for chin-chest contact and occiput-back contact
R6) Must have space for at least 9 sensor cables (1.5 mm diameter) from the torso to the head
R7) Should return to its neutral position when not loaded

2 Methods and Materials

2.1 Concept Phase

Based on these design requirements, several possible concept solutions were identified to address these challenges. These solutions were distinct from each other by either the achievable ROM, the stiffness achievable and the degrees of freedom (DOF) of the surrogate. As the neck must allow for motion in flexion and extension, but can have multiple DOFs, options have been explored for 1 to 3 DOF.

The full concept exploration process description can be found in Appendix A.

Based on manufacturability and stiffness feasibility, 3 working mechanisms were selected for further evaluation:

C1) A neck model consisting of a ball and socket joint (3DOF) supported by torsion springs.
C2) A neck model based on the compliant hinge design from Fowler et al. [23]
C3) A neck model based on the cylindrical compliant pivot design by Dearden et al. [24]

Harris profiles have been constructed for evaluation of the 3 concepts, as seen in Figure 4 [25]. Criteria based on design requirements **R2-R6** are sorted from top to bottom from most important to least important.

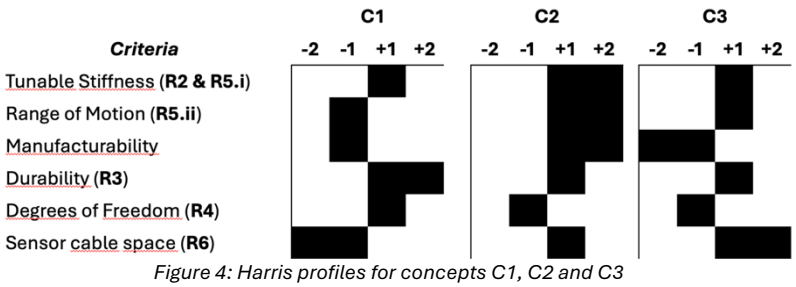

Figure 4: Harris profiles for concepts C1, C2 and C3

Based on the Harris profiles, the working mechanism chosen is based on a monolithic compliant space hinge by Fowler et al. [23]. This hinge is able to have a ROM of 180 degrees in one plane, while still allowing for displacements in every other plane. It consists of 16 flexures in a star-like shape, coupled with rigid bodies. A picture of the generic shape of the hinge is shown in Figure 5. While initially designed to deploy solar panels in space, the research team states that the stiffness of this hinge can be tuned by choosing material and geometric properties. Five prototypes were fabricated from polypropylene (PP-1 & PP-2), titanium (Ti-1) and a carbon nanotube framework (CNT-1 & CNT-2), with varying geometries and therefore stiffnesses. Table 1 in Appendix C shows all geometric and material properties of these 5 prototypes. These prototypes show potential to be used in a neck design by tuning this compliant mechanism to the desired stiffness while having a sufficiently large ROM.

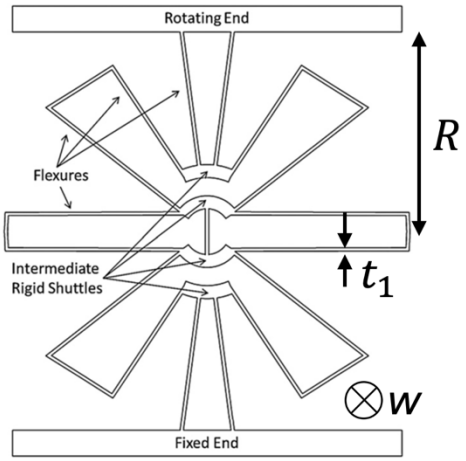

Figure 5: Monolithic compliant hinge shape, retrieved from Fowler et al. [23]

2.2 Monolithic Hinge Optimization Phase

To design this monolithic hinge to fit the purpose as an infant surrogate neck, a finite element method was used to determine bending stiffness and maximum stress (σ_{max}), done using a parametric ABAQUS input script [26]. This parametric approach allows quick variation of geometric parameters; radius R, flexure thickness t_1 and hinge width w, as well as material parameters to observe the influence on stiffness and maximum stresses. The full Python script for this model can be found in Appendix B. The model was validated using the geometries and materials used by Fowler et al. and comparing our results with their simulations. A mesh convergence test was performed for prototypes PP-2 and Ti-1 from Fowler et al. (see Appendix C), to establish a mesh size which acknowledges both accuracy and minimal computational power, after which a mesh consisting of 1000 elements was considered sufficient. To keep a constant element count, mesh size had to be varied according to the geometry of the hinge.

The parametric ABAQUS model was used to calculate the bending stiffness resulting from different material and geometrical properties, as well as to assess the elasticity of the hinge. Simulations were performed for multiple materials, radii, flexure thicknesses and widths, seen in Appendix D. Based on these simulations, relations were identified between the stiffness, material, and geometric properties, as well as maximum stress, material and geometric properties.

A Python-based regression analysis resulted in two expressions: a stiffness model $k = f(E, t_1, R, w)$ and an elastic-limit model expressed as the ratio $\frac{E}{\sigma_{max}} = g(t_1, R)$ with E the Young's modulus, t_1 the thickness of the flexures, R the radius of the hinge and w the width of the hinge. As elastic behavior demands $\sigma_y \geq \sigma_{max}$, the ratio $\frac{E}{\sigma_y}$ for materials in Appendix H & I was calculated and multiplied with safety factors 1.5 and 2. Stiffness contours were plotted in the t_1, R plane for several hinge widths and a set Young's modulus, while corresponding contours of $\frac{E}{\sigma_y}$ including safety factors were plotted on the same grid to show minimum dimensions that prevent yielding. Overlaying the stiffness and stress models reveals intersection dimensions that identify the material and geometry combinations resulting in a target stiffness without exceeding the yield limit.

To manufacture the prototypes, multiple available manufacturing methods and corresponding materials were investigated. Manufacturing methods considered include laser cutting, 3D printing and CNC milling. Furthermore, available materials were investigated on their ratio $\frac{E}{\sigma_y}$ and the possibilities to produce the hinges from the given materials using the manufacturing methods investigated.

To manufacture the hinge, a parametric SolidWorks model was created. By using this approach, geometries can be quickly varied based on simulation results. Prototypes consisting of one or two hinges to make the hinge as wide as possible for maximum stiffness and to make space for cables, placed

between Q0 complying endplates, were manufactured from polylactic acid (PLA) and carbon fiber reinforced polyethylene terephthalate glycol (PETG-CF). Prototypes were manufactured using the Bambu Lab A1 FDM printer, nozzle size 0.4 mm, layer height 0.2 mm, infill 100% [27].

2.3 Experiment Phase

In order to measure to what extent the ABAQUS simulations are true to reality, a quasi-static loading experiment was conducted. Each of the 8 distinct prototypes, described in Table 1, varying in geometry and material, were fixed on one end, while the other end was used for loading. A custom testing endplate (Figure 5, item D) was produced which offered a point to apply a tension force through a cable. The testing endplate could be attached to the neck in two orientations, allowing for testing in flexion/extension and lateral bending. The cable was run through a pulley to make sure the direction of the load was parallel to the top of the endplate, to apply a moment around the center of the hinge. A load cell (LSB200, Futek Advanced Sensor Technology, Irvine, CA, USA) calibrated to 30 N, was attached to this cable. To allow for easy video analysis, a red and a blue marker were placed on the end of the testing endplate and at the contact point with the neck respectively. The red marker was used to analyze the angle, where the blue marker was needed due to the center shift of the hinge. A camera (Apple iPhone 15 Pro, Telelenscamera — 77 mm ƒ2.8, 30 fps) was placed at 2 meters distance with the optical axis aimed at the blue marker to minimize perspective.

Table 1: Geometric and material properties for manufactured prototypes including ABAQUS results based on expected geometries

Prototype	$R\ (mm)$	$t_1\ (mm)$	$w\ (mm)$	$E\ (GPa)$	$\sigma_y\ (MPa)$	$\sigma_{max}\ (MPa)$ (ABAQUS)	$\frac{\sigma_y}{\sigma_{max}}(-)$	Expected Stiffness FE (Nm/rad) (ABAQUS)
PLA1	35	0.8	10	2.75	75	45.1	1.66	0.07
PLA2	30	0.8	2 x 7.5			52.3	1.43	0.12
PLA3	45	1.1	20			47.2	1.59	0.28
PLA4	22	0.8	2 x 5			70.6	1.06	0.11
PETG-CF1	25	1	20	2.89	83	80.3	1.03	0.40
PETG-CF2	30	1.2	2 x 10			78.6	1.05	0.56
PETG-CF3	35	1.2	15			68.1	1.22	0.36
PETG-CF4	15	0.6	2 x 5			82.2	1.01	0.07

The neck was rotated to 90° by pulling the cable by hand, aimed to achieve the maximum rotation after 5 seconds while keeping the pulley wrap angle at 90°. The load cell was monitored by LABView with intervals of 100 ms. The angle of deflection was calculated by video analysis software Tracker [28], which calculated the angle between the red marker (F), the blue marker (E) and the horizontal start position. This was done using a frame rate of 30 fps, and every 3 frames the angle was calculated to obtain a sample time of 100ms. As the hinge is loaded asymmetrical in the start position, the angle data is corrected to start at 0 degrees. This will not influence the bending stiffness as the stiffness is determined by the change in moment and rotation. A picture of this experimental setup is provided in Figure 6, with a schematic overview in Figure 7.

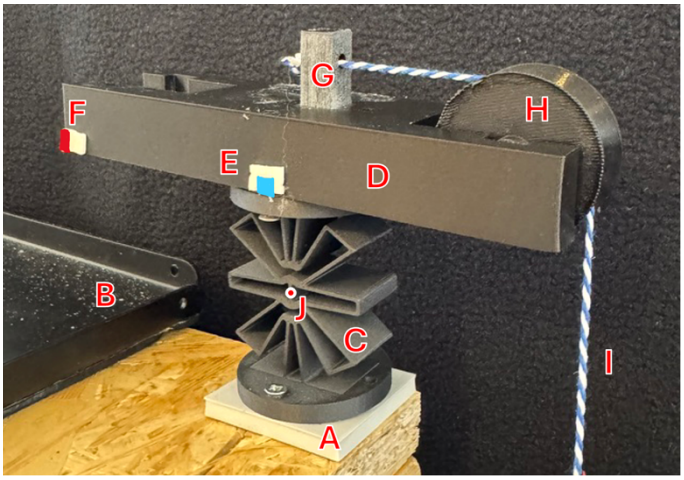

Figure 6: Experimental setup; A: Fixation, B: Counterweight, C: Hinge, D: Testing endplate, E: Blue marker, F: Red marker, G: Force application point, H: Pulley, I: Cable with load cell, J: Geometric center of the hinge

The bending moment about the hinge center, M_J was obtained by using the static equilibrium, seen in Figure 8: $\sum M_J = F_c\ d - F_{p,x}\ e + F_{p,y}\ h$, where F_c is the tensile load obtained from the load cell in N, $F_{p,x}$ and $F_{p,y}$ the horizontal and vertical projections of the pulley reaction force on the endplate in N, d the perpendicular distance from the cable to the geometric center of the hinge in m, measured once per prototype with a digital caliper. Parameters e and h represent the horizontal and vertical distances from the pulley support to the hinge center, with $e = d - 0.012$ and $h = 0.06\ m$. The 90° wrap angle remains constant, leading to a constant direction of the pulley reaction force, hence $F_{p,x} = F_{p,y} = F_c$.

Substituting these equal forces and the expressions for e and h simplifies the moment to $\sum M_J = 0.072F_c$. The complete derivation with additional free body diagrams is provided in Appendix L.

To process this data, angle and moment data were synchronized, with a sample interval of 100 ms. Data was filtered by selecting values from the start of the rotation until the peak rotation. The angle and moment data and were plotted against each other by eliminating the time. The average stiffness of the neck was obtained by calculating the slope of a linear fit on the moment-rotation data. All data processing was performed using Python. This experimental data was compared to the corresponding simulation data, by calculating the relative errors between expected and measured values.

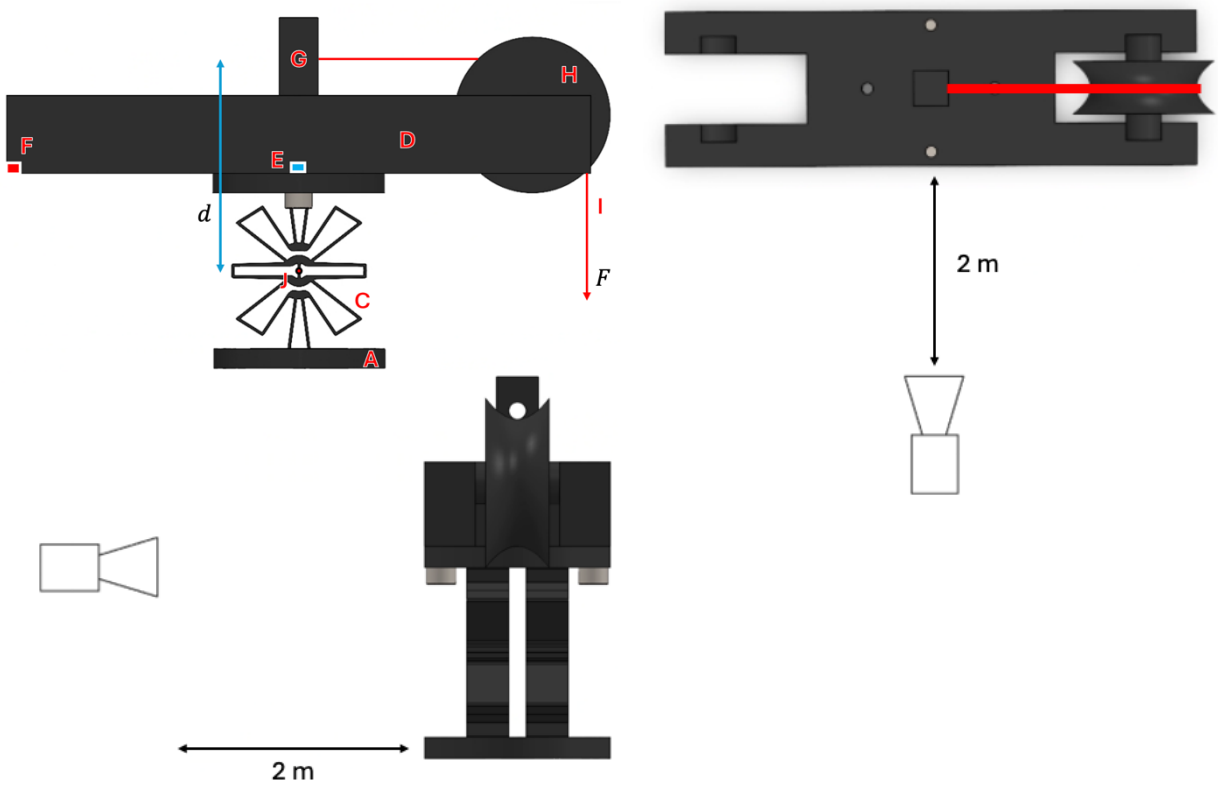

Figure 7: Experimental setup with tension force F, the distance from the cable to the hinge center d, and camera placement. Note that the image is not in scale. A: Fixation, C: Hinge, D: Testing endplate, E: Blue marker, F: Red marker, G: Force application point, H: Pulley, I: Cable with load cell, J: Geometric center of the hinge

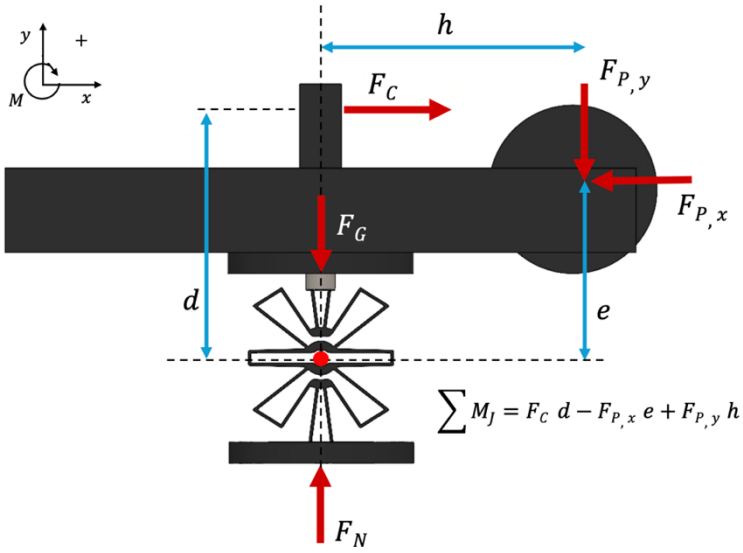

Figure 8: Free body diagram of the hinge in experimental setup. Fc is the tension force in the cable, Fp,x and Fpy the horizontal and vertical projections of the reaction force of the pulley, Fg the weight of the system and Fn the normal force. d, h and e are perpendicular distances to the center of the hinge.

3 Results

3.1 ABAQUS Model Validation

Table 2 shows the comparison between the results created by the ABAQUS code and the results of Fowler et al., including relative error. For the model validation, the maximum stress in the flexures was determined at a rotation of 90 degrees for comparison with results from Fowler et al. [23]. The maximum stress in the hinge is observed at the connection of the flexures with the rigid bodies, as seen in Figure 9. To ensure that the hinge can rotate 90 degrees, the target rotation in the model will be set at 1.7 radians, which will be used in a mesh convergence test. The mesh convergence tests for the ABAQUS input file, by using the prototype models PP-2 and Ti-1 from Fowler et al. can be found in Appendix C.

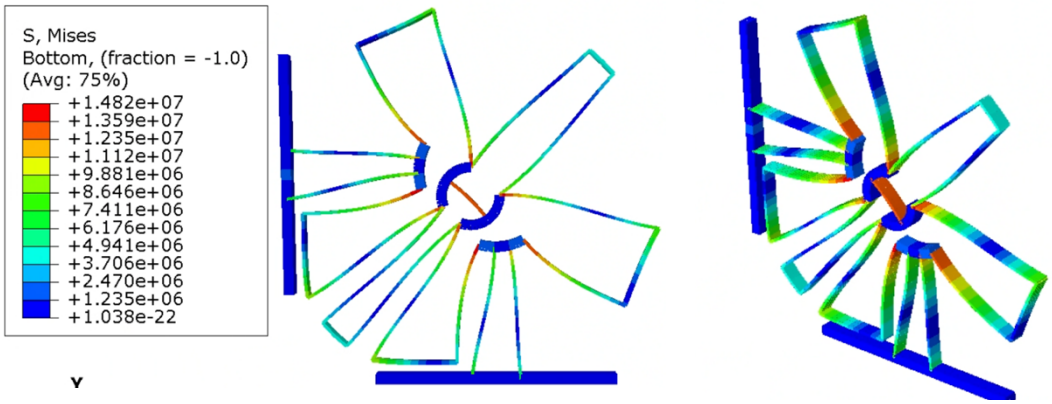

Figure 9: FEM simulation of prototype Ti-1 by Fowler et al. [23], deflected till 90 degrees. The maximum stress occurs at the connection between the flexures and the rigid bodies.

Table 2: Model validation results: comparison results Fowler et al. [23], with geometries in Appendix C, and ABAQUS results

Prototype	σ_{max} (MPa) [21]	σ_{max} (MPa) (ABAQUS)	Error	Bending Stiffness (Nm/rad) [21]	Bending Stiffness (Nm/rad) (ABAQUS)	Error
PP-1	20.6	20.2	1.94%	0.450	0.430	4.44%
PP-2	15.3	14.8	3.38%	0.311	0.310	0.32%
Ti-1	805	808	0.37%	1.27	1.25	1.57%
CNT-1	30.0	30.4	1.33%	1.74	1.70 * 10-6	2.30%
CNT-2	40.1	40.7	1.50%	1.03	1.01 * 10-6	1.94%

3.2 Design Space

All performed simulation results can be found in Appendix D. The shape of the hinge is the same as the Ti-1 prototype by Fowler et al., while the length of the four horizontal flexures is scaled by a factor 0.85 to ensure the hinge does not exceed the geometry of the endplates. Based on these simulations, the following relations were found:

$$Stiffness = f(E, t_1, R, w)$$
$$\sigma_{max} = g(E, t_1, R)$$

With E the Young's modulus, t_1 the thickness of the flexures, R the radius of the hinge and w the width of the hinge.

It was observed that the ratio of $\frac{E}{\sigma_{max}}$ was shape dependent, as this ratio was the same for every simulation using the same t_1 and R. Furthermore, it was observed that the maximum stress was independent of the width of the hinge.
By using a power-law regression, the following equations for bending stiffness and maximum stress were established:

$$Stiffness = 0.000142 \frac{t_1^{2.92}}{R^{0.969}} w E^{1.001} \left(\frac{Nm}{rad}\right) \quad (1)$$
$$\frac{E}{\sigma_{max}} = 1.61 \frac{R^{0.978}}{t_1^{0.921}} (-) \quad (2)$$

With E the Young's modulus in MPa, t_1 the thickness of the flexures in mm, R the radius of the hinge in mm and w the width of the hinge in mm.

As $\frac{\sigma_y}{\sigma_{max}} \geq 1$ for elastic deformations, the following relationship must be true in order for the hinge to not deform plastically:

$$\frac{E}{\sigma_y} \leq 1.61 \frac{R^{0.978}}{t_1^{0.921}} \ (-) \tag{3}$$

The Python scripts used for data analysis can be found in Appendices E and F.

3.3 Material and geometry selection

To satisfy the requirement for the neck to have a length smaller than 50 mm, R should be aimed to have an approximate length of 20 mm, to leave space for endplates connecting the neck to the torso and head. According to Equation 3, the ratio $\frac{E}{\sigma_y}$ of the material should be smaller than 48.3 for this geometry to remain within elastic limits. Most potential was shown by fiber reinforced polymers due to their small $\frac{E}{\sigma_y}$, which were used in further calculations. However, t_1 cannot be smaller than 0.6 mm due to 3D printing manufacturing limits regarding wall thickness. A list of the properties and ratio $\frac{E}{\sigma_y}$ of fiber reinforced composites produced by Bambu Lab can be found in Appendix I.

The contour plots for PLA at w = 10 and for PETG-CF at w = 10, 20 can be found in Figure 10. Plots for more materials by Bambu Lab can be found in Appendix J. The shaded grey area shows for which geometries the hinge would deform plastically, including different safety factors (SF). The colored lines show what the stiffness of the hinge would be at a given geometries. The corresponding t_1 and R would be the geometry where the part has a given stiffness and σ_{max} would equal σ_y. Therefore, feasible parameters sets will lie right of this line. This will be the so-called design space for the set of requirements.

Multiple sets of geometry and materials were chosen for prototyping, either with one hinge between the endplates or two for increased off-axis stiffness, as seen in Table 3.

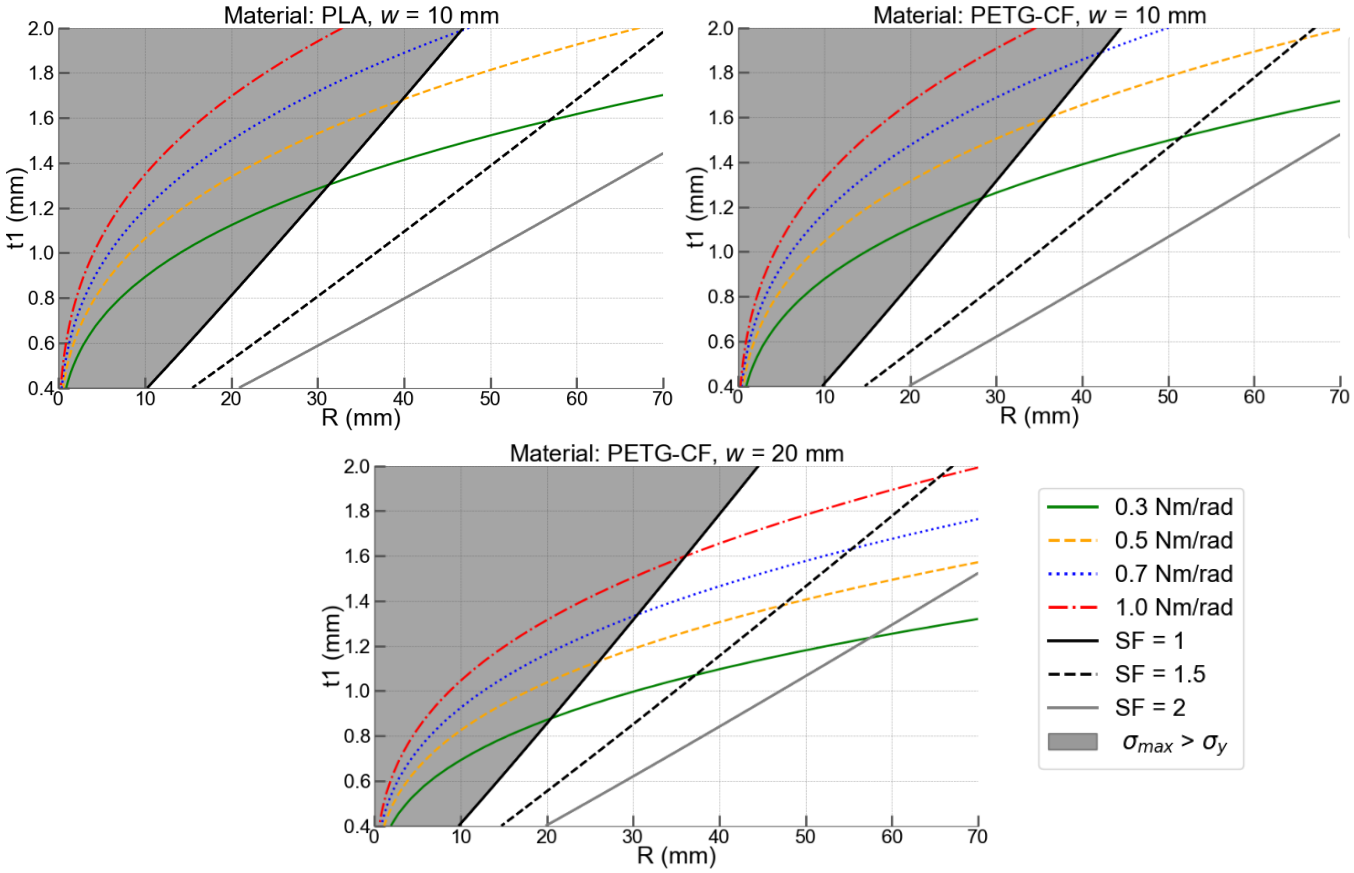

Figure 10: Geometry selection graphs for PETG-CF and PLA for a hinge width of 10 mm and 20 mm. In the shaded grey area, the expected maximum stress in the flexures will be above the yield stress, resulting in plastic deformation. Safety factors (SF) of 1, 1.5 and 2 are shown, alongside target stiffnesses of 0.3, 0.5, 0.7 and 1.0 Nm/rad on the same grid.

PLA and PETG-CF filaments by Bambu Lab [27] were chosen as materials, where the PLA served as model validation prototypes, as they would not fail but did not reach the desired stiffness. PETG-CF was able to reach an expected stiffness of 0.5 Nm/rad, while exceeding the target length of 20 mm. Computed moment-rotation and stress-rotation curves for each prototype can be found in Appendix K. A picture of the CAD model and manufactured prototype of PETG-CF2 can be found in Figure 11 and 12.

Table 3: Geometric and material properties for manufactured prototypes including ABAQUS results based on measured geometries

Prototype	$R\ (mm)$	$t_1\ (mm)$	$w\ (mm)$	$E\ (GPa)$	$\sigma_y\ (MPa)$	$\sigma_{max}\ (MPa)$ (ABAQUS)	$\frac{\sigma_y}{\sigma_{max}}(-)$	Expected Stiffness FE (Nm/rad) (ABAQUS)
PLA1	35	1.0	10	2.75	75	55.4	1.35	0.14
PLA2	30	0.8	2 x 7.5			52.3	1.43	0.12
PLA3	45	1.0	20			43.3	1.73	0.21
PLA4	22	0.8	2 x 5			70.6	1.06	0.11
PETG-CF1	25	1.1	20	2.89	83	86.6	0.96	0.52
PETG-CF2	30	1.1	2 x 10			73.2	1.13	0.44
PETG-CF3	35	1.3	15			72.7	1.14	0.45
PETG-CF4	15	0.6	2 x 5			82.2	1.01	0.07

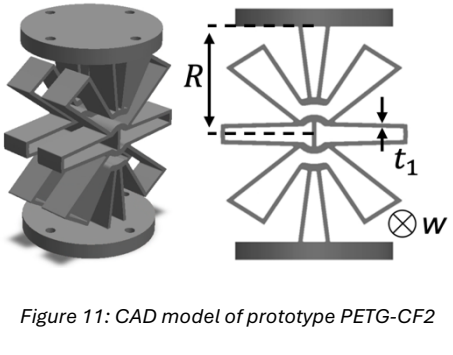

Figure 11: CAD model of prototype PETG-CF2

Figure 12: Pictures of prototype PETG-CF2

3.4 Experimental results

Table 4 shows the measured properties of the dummy necks. The hinge geometry was measured after manufacturing and new stiffness predictions were performed based on the actual, produced geometry.

Table 4: Experimental results, including the distance measured from the center of the hinge to the force application point, and comparison in FE

Prototype	d [mm]	Expected Stiffness FE (Nm/rad)	Measured Stiffness FE (Nm/rad)	Error (-)	Measured Stiffness LB (Nm/rad)
PLA1	73	0.14	Failed	-	Failed
PLA2	68	0.12	0.10	-16.7%	0.23
PLA3	83	0.21	0.15	-28.6%	0.43
PLA4	60	0.11	Failed	-	Failed
PETG-CF1	63	0.52	Failed	-	Failed
PETG-CF2	68	0.44	0.31	-29.5%	0.51
PETG-CF3	73	0.45	Failed	-	Failed
PETG-CF4	53	0.07	0.06	-14.3%	0.68

Figure 13 shows the rotation of PLA2 at 0, 45 and 90 degrees as well as the full rotation path. The moment-angle curves for PLA2 and PETG-CF2 can be found in Figure 14. The full experimental results can be found in Appendix M. Prototypes PLA1, PLA4, PETG-CF1 and PETG-CF3 failed at the connection between a flexure and the rigid shuttle, as shown in Figure 15.

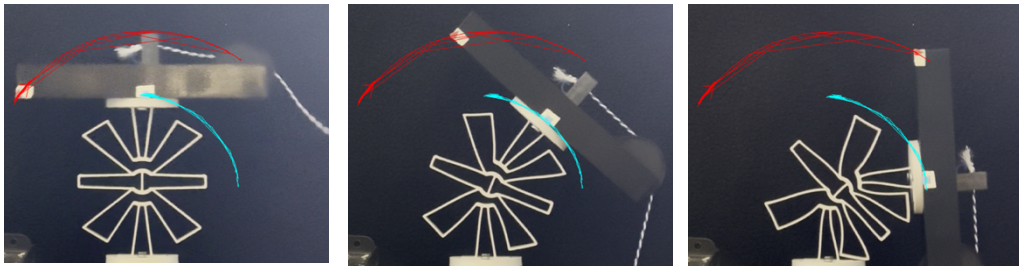

Figure 13: Screenshots of motion analysis software Tracker at 0, 45 and 90 degrees. The red line shows the path of the red marker, the blue line the path of the blue marker.

Based on the simulations and measurements a final optimized prototype was built, of which the geometry can be found in Table 5, as well as an image of the FEM simulation in Figure 16. The ratio $\frac{\sigma_y}{\sigma_{max}}$ was increased to improve durability, as prototypes PLA1, PLA4, PETG-CF1 and PETG-CF3 failed. The moment-angle curve for the final prototype can be seen in Figure 17. Holes in the endplates were made to pass cables from the torso to the head.

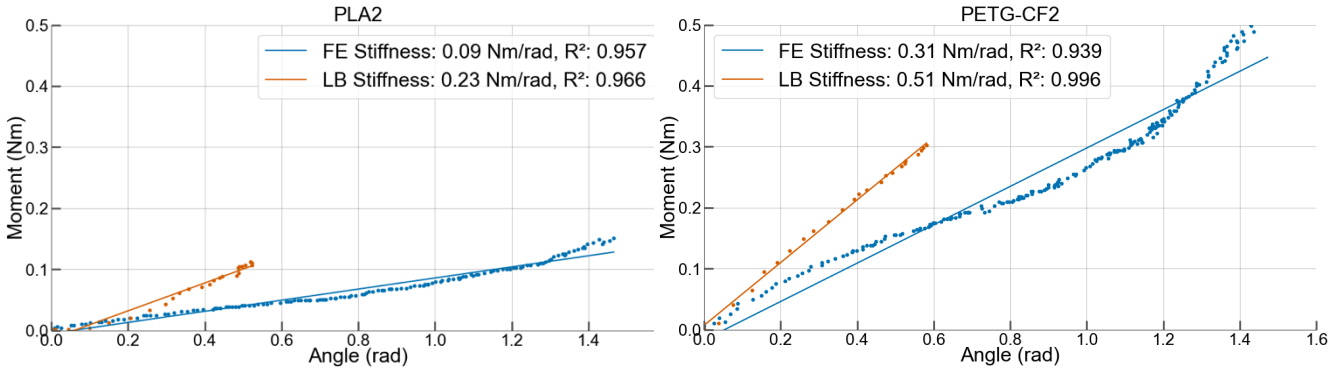

Figure 14: Experimental moment-angle curves for prototypes PLA2 and PETG-CF2 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is shown including R^2 value.

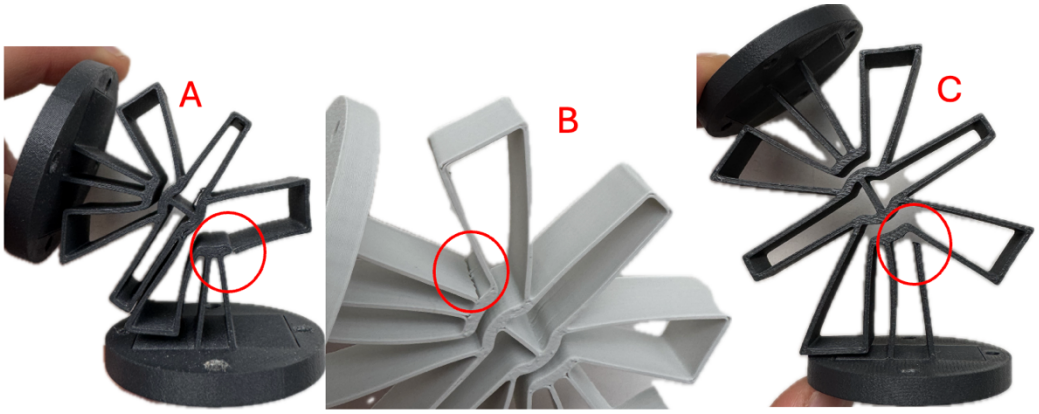

Figure 15: Images of the failed hinges from A: PETG-CF1, B: PLA1, C: PETG-CF3

Table 5: Geometric and material properties of final prototype after experiments

PETG-CF5	$R\ (mm)$	$t_1\ (mm)$	$w\ (mm)$	$d\ (mm)$	$E\ (GPa)$	$\sigma_y\ (MPa)$	$\sigma_{max}\ (MPa)$	$\frac{\sigma_y}{\sigma_{max}}(-)$	Expected Stiffness FE (Nm/rad)	Measured Stiffness FE (Nm/rad)
	20	0.6	2 x 10	-	2.89	83	61.6	1.35	0.11	-
Measured	20	0.8	2 x 10	56			65.0	1.28	0.26	0.22

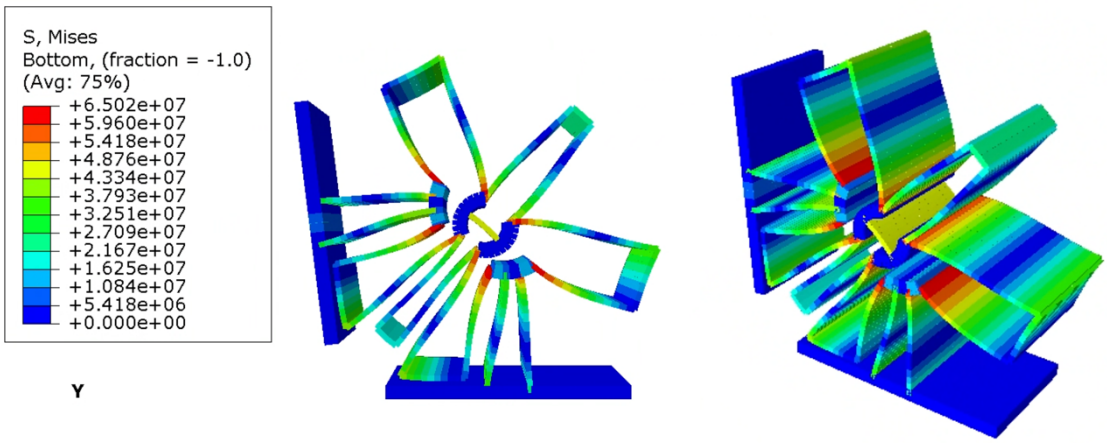

Figure 16: FEM simulation of prototype PETG-CF5, deflected till 1.7 radians. The maximum stress occurs at the connection between the flexures and the rigid bodies.

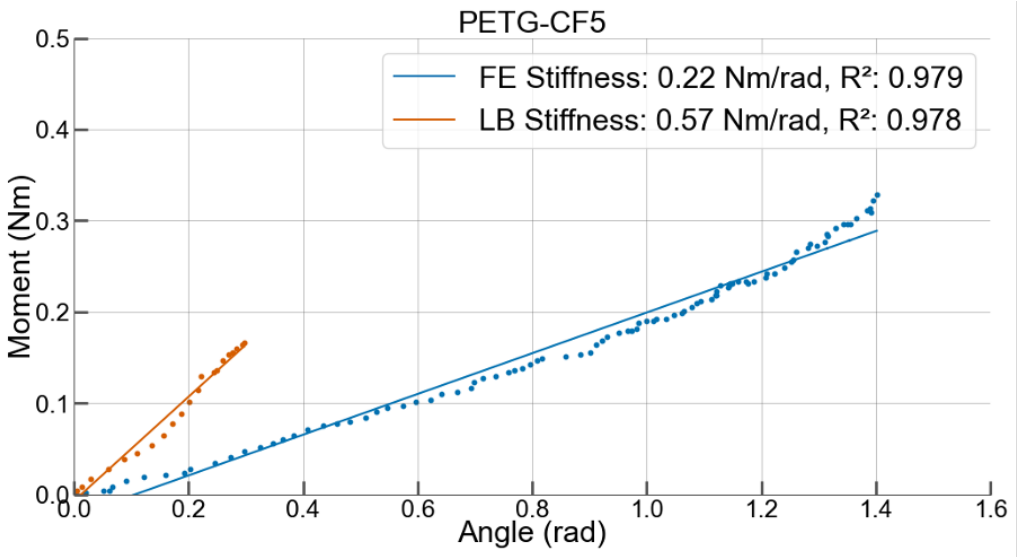

Figure 17: Experimental moment-angle curves for prototype PETG-CF5 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is shown including R^2 value.

4 Discussion

This section reflects on the biofidelity of the design, the validity of the model, and the outcomes of the computational and experimental results. Furthermore, recommendations for further research are stated to improve shaking simulations in infants.

4.1 Biofidelity

A key objective was to match the infant neck flexion-extension stiffness by Luck et al. [16] and Sullivan et al [17]., 0.2 Nm/rad in flexion and 0.4 Nm/rad in extension, while able to have a range of motion of 180 degrees. The prototypes built confirmed the concept of a monolithic hinge with a tunable stiffness, with a measured stiffness of 0.22 Nm/rad for a prototype with two hinges with geometric properties R = 20 mm, t1 = 0.8 mm and w = 10 mm. However, only the lower boundary of the target stiffness was reached when complying with the desired neck length in the final prototype, due to material and geometric limitations, as well as manufacturing limits. Furthermore, the flexion stiffness was equal to the extension stiffness as attempts to make the hinge asymmetric lead to contact between the flexures before the motion was completed. While this design can obtain the extrapolated structural bending stiffness by Sullivan et al. in FE and have a range of motion of 180 degrees, it does not consider the actual motion path of the head due to the seven vertebrae of the neck, as well as possible differences in material properties along the cervical spine.

This monolithic neck design simplifies the neck to a single bending axis, while, in contrast, real infant necks allow flexion, extension, lateral bending and axial rotation.
Research shows that the shaking movement occurs mostly in the sagittal plane, thus in flexion and extension [9, 11]. In addition to the lack of infant data, biofidelic flexion/extension was prioritized over lateral bending and axial rotation for this study. Preferably, the neck should allow for some motion in lateral bending and axial rotation, while not tunable and having a biofidelic range of motion as this would not be feasible within the design scope. Due to the flexures in the final design, the neck does not behave as a perfect hinge, allowing for motion in flexion/extension, lateral bending, and tension/compression, as seen in Figure 18.

Figure 18: From left to right; flexion/extension, lateral bending, compression of prototype PETG-CF5

A main challenge in improving surrogate infant necks for shaking experiments is replicating the dynamics of the head-neck complex. Real infant cervical spines consist of seven vertebrae, intervertebral discs, ligaments, and muscles, which will exhibit viscoelastic behavior, meaning the stiffness of the neck will be depending on the magnitude and rate of loading. The viscoelastic properties of actual infant necks will have a significant impact on head dynamics due to energy dissipation, which will be of particular interest in shaking situations due to their rapid and repetitive loading. Additionally, the work of Sullivan et al. remains based on extrapolation of functional spine unit experiments, of which it is unsure to what extent the extrapolation of separate motion segments approximates the structural mechanical properties of the whole infant cervical spine.

4.2 ABAQUS Model Validation

When comparing the stress and stiffness values calculated by the ABAQUS input script to the values provided by Fowler et al., some differences can be observed. For the maximum stress, the relative errors range from 0.37% to 3.38%. For the stiffness, when looking at the PP and Ti prototypes, the relative errors range from 0.32% to 4.44%. For the CNT prototypes, the predicted stiffness values were smaller by a factor 10^6 compared to the values from Fowler et al., while computing a corresponding maximum stress. Despite attempts to clarify this inconsistency, no definitive explanation was found. Taking the other prototypes in consideration, the errors were considered small enough. Due to the non-linearity and large deformations in the models, errors will occur as ANSYS (used by Fowler et al.) and ABAQUS use different solving methods which will introduce differences between these 2 models as input parameters were constant across both models. Since the PP and Ti prototypes matched and were closer to our geometric goal, the model was concluded sufficiently accurate for the design.

4.3 Material selection

The selection of a material proved to be critical to build a feasible hinge in terms of stiffness, durability, and manufacturability. Initial prototypes printed in PLA validated the feasibility of the design, while based on the simulations. their stiffness would not reach the target stiffness. Fiber reinforced polymers were more promising due to their low $\frac{E}{\sigma_y}$ ratio compared to other material classes. However, using these polymers manufactured by 3D printing could lead to problems such as bad layer adhesion and warping. A material with consistent prototype quality was found in PETG-CF, while only able to reach the target stiffness if the target length was increased. If a PETG-CF neck was used in a shaking simulation situation, this increase in length compared to the Q0 neck would influence the accelerations in the head.

The length used in the design requirements is based on the Q0 neck, which corresponds to a hinge radius of 20 mm. However, a study by Kasai et al. [29] state that the mean cervical spine length of a 1-year-old is 19.3 ± 1.5 mm measured from vertebrae C3-C7, meaning the Q0 neck is double the length of an actual infant neck. If parameters from Kasai et al. ($R \approx 10$ mm, t_1 = 0,6 mm) were put in Equation 3, the material ratio should be smaller than 24.5, which is not found in investigated materials seen in Appendix H & I, which include several polymers and metals, due to their ratio $\frac{E}{\sigma_y}$ being too large.

4.4 Comparison computational and experimental results

When comparing the ABAQUS results with the experimental results, the shape of the moment-angle curves for PLA prototypes mostly aligns, while for PETG-CF prototypes, several discrepancies can be found. The PLA prototypes can reach a rotation of 90 degrees, with a linear stiffness over the full range of motion. Looking to the PETG-CF prototypes, only PETG-CF4 was not able to reach a rotation of 90 degrees, with a maximum rotation of 76 degrees, due to the endplates making contact. However, PLA1 and PLA4, as well as PETG-CF1 and PETG-CF3 failed after several rotations before testing.
When comparing the calculated stiffness to the expected stiffness, the values are between 14.3% and 29.5% lower than expected. This may be due to 3D printing induced anisotropy of the flexures. As the hinge is printed on its side, the hinge is anisotropic in the lateral direction and thus weaker. Simulations show that the stiffness is directly proportional to the width of the hinge, however, the simulations are based on an isotropic model, which can explain the lower experimental stiffness compared to the computational results.

Furthermore, the wrap angle of the cable around the pulley is assumed constant in the moment calculations, as well as the assumption that the pulley is massless and frictionless. However, in the real-world experiment, the angle will not be perfectly constant due to the hand-applied tension, which, in combination with a non-ideal pulley, will alter the moments calculated about the center of the hinge. Firstly, the effect of the wrap angle on the calculated moment can be captured by the following equation: $\sum M_J = F_C\ d - \sqrt{2}F_C\,(d - 0.012)\cos(\theta) + \sqrt{2} * 0.06F_C \sin(\theta)$, with θ the wrap angle divided by 2, instead of the simplified $\sum M_J = 0.072F_C$. However, as the deviation of the wrap angle was not measured, this expression cannot be used. Furthermore, assuming the pulley is ideal will lead to an over-estimation of the moment around the hinge center.

Additionally, the moment-angle curve of PETG-CF2 showed parts of nonlinear behavior, which is not expected as a linearly elastic material model was assumed.
Research by Economides et al. [30] show that PETG-CF is highly elastic but can inhibit viscoelastic behavior depending on its 3D print parameters. While it may be an explanation for the deviation in the moment-angle curves, it cannot be concluded as mechanical properties between different 3D printing filaments can vary widely, and only elastic material properties are stated by the manufacturer.
Another possible explanation may be that the material locally deforms plastically due to fiber debonding, which would influence the stiffness.

4.5 Data Analysis

A power law regression model was chosen to fit the simulation data, with a corresponding R^2 of 0.999 for both regression models. This means the stiffness can be predicted very well using the parameters identified. However, as the model is fitted only using values of R between 20 and 70 and t_1 values between 0.4 and 1.2, it is unclear if there would be significant changes in the relations for extreme values. For calculating the stiffness of the different necks, a linear fit was used to compute the stiffness for every neck, with R^2 values ranging from 0.939 to 0.999. This means the linear model captures the behavior well enough, confirming the elastic behavior from the neck.

4.6 Future Research and Recommendations

To improve the biofidelity and the use of the surrogate in shaking simulations, several aspects require further improvement. Firstly, dynamic shaking experiments will be essential to validate the neck under conditions suited for shaking experiments, rather than relying on static testing. While being able to measure the structural bending stiffness, static tests do not consider the effects of rapid, repetitive movement seen in IHI-ST. Also, dynamic testing needs to be performed to ensure the neck will not undergo plastic deformation or fatigue failing during rapid, repetitive movement in shaking simulations. Viscoelastic behavior will be of particular interest during rapid, repetitive movement, explaining the need to implement viscoelastic segments in the neck. This could be achieved by choosing a viscoelastic material, or by incorporating viscoelastic parts in addition to the elastic hinges.

Second, the current design only considers the stiffness and range of motion in flexion and extension as objective, leaving lateral bending and axial rotation unaccounted for. Additionally, the flexion and extension stiffness in this neck design are equal, due to the symmetric shape of the hinge. Attempts to tune the stiffnesses separately by making the hinge width twice as large on one side instead of a constant width were unsuccessful, due to fracture and contact in the hinge before the full rotation, as seen in Figure 19.

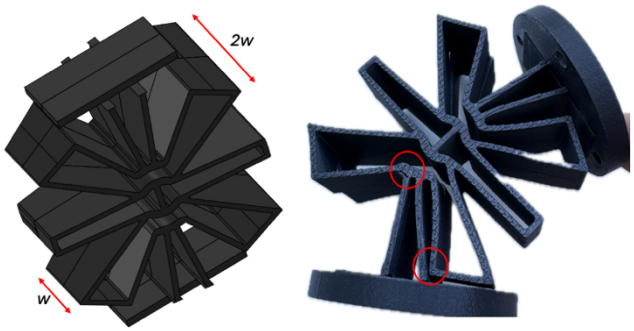

Figure 19: Failed prototype with varying hinge width w

Therefore, a multi-degrees of freedom or segmented concepts could improve the biofidelty of the neck, though no experimental stiffness was found in lateral bending or axial rotation. For instance, this could

be achieved by stacking multiple hinges with different orientations to allow a tunable stiffness in lateral bending and axial rotation. To approximate the structure of an infant neck, multiple hinges with a lower range of motion could be stacked to influence the way the neck deflects. Furthermore, the final prototype design length exceeds the neck length measured by Kasai et al. [29], due to material limitations. This may lead to altered head paths and accelerations during shaking experiments compared to necks which approximate the length of an infant's neck.
Third, material limitations currently make it this design not feasible to comply with design requirements R1 and R5 in terms of geometry and stiffness. Although fiber reinforced polymer composites showed promise, the experimental stiffness values were lower than expected, indicating print optimization is needed or a manufacturing method which does not introduce anisotropy has to be considered. Alternative composites with a low $\frac{E}{\sigma_y}$ while having a sufficient high Young's modulus to reach the target stiffness could lead to a feasible solution if manufactured properly.

Lastly, the starting point for this design were the experimental values by Luck and Sullivan, based on an extrapolation of experimental FSU stiffness data, making it uncertain to what extent these values are true to reality.
While the surrogate neck can be tuned to various stiffness values and thus quickly adapted when new knowledge emerges, it is uncertain if these properties will be ever experimented due to ethical considerations.

5 Conclusion

The objective of this thesis was to design a high range of motion, tunable surrogate neck for shaking simulations, based on experimental data. To accurately perform shaking simulations, infant dummies based on experimental infant data should be used, instead of scaled adult properties which provide no validated representation of the premature neck.

Experimental FSU stiffness values by Sullivan et al. and Luck et al. (0.2 Nm/rad in flexion and 0.4 Nm/rad in extension) were used due to the lack of other studies performed, and the desired range of motion was established at 90 degrees. A monolithic compliant hinge by Fowler et al. was used as working principle and using finite element simulations the geometric and material properties were optimized to meet the design requirements. Prototypes were built from PLA and PETG-CF and tested on their stiffness, reaching the lower limit of the design requirements, explainable by 3D printing induced anisotropy and material and manufacturing limitations. However, the desired stiffness was only attainable at a neck length longer than for actual infants.
Nevertheless, the desired range of motion was reached, allowing for chin-to-chest and occiput-to-back contact.
Future research should focus on dynamic testing of the neck design for durability and incorporating all motions of the infant's neck. Additionally, viscoelastic behavior of the infant neck should be incorporated, as well as the head trajectory due to the rotation of the neck. Lastly, lateral bending and axial rotation properties must be obtained to validate a surrogate neck. If experimental infant neck properties would be ever available, the methodology in this thesis can be used to design a surrogate neck able to rotate 90 degrees with a predetermined stiffness.
Inflicted head injury by shaking trauma is a complex phenomenon that current used dummies fail to fully capture, underscoring the need for improved biofidelic surrogates mirroring real-world infant head-neck dynamics.

References

[1] Talvik, I., Alexander, R. C., Janson, S., Palusci, V. J., & Victora, C. G. (2007). Inflicted traumatic brain injury (ITBI) or shaken baby syndrome (SBS) in Estonia. Acta Paediatrica, 95(7), 799-804. https://doi.org/10.1111/j.1651-2227.2006.tb02343.x

[2] Niederkrotenthaler, T., Xu, L., Parks, S. E., & Sugerman, D. E. (2013). Descriptive factors of abusive head trauma in young children—United States, 2000-2009. Child Abuse & Neglect, 37(7), 446-455. https://doi.org/10.1016/j.chiabu.2013.02.002

[3] Keenan, H. T., Runyan, D. K., Marshall, S. W., Nocera, M. A., Merten, D. F., & Sinal, S. H. (2003). A population-based study of inflicted traumatic brain injury in young children. JAMA, 290(5), 621-626. https://doi.org/10.1001/jama.290.5.621

[4] Fanconi, M., & Lips, U. (2010). Shaken baby syndrome in Switzerland: Results of a prospective follow-up study, 2002-2007. European Journal of Pediatrics, 169(8), 1023-1028. https://doi.org/10.1007/s00431-010-1175-x

[5] Ellingson, K. D., Leventhal, J. M., & Weiss, H. B. (2008). Using hospital discharge data to track inflicted traumatic brain injury. American Journal of Preventive Medicine, 34(4 Suppl), S157-S162. https://doi.org/10.1016/j.amepre.2007.12.021

[6] Christian, C. W., Block, R., & Committee on Child Abuse and Neglect, American Academy of Pediatrics. (2009). Abusive head trauma in infants and children. Pediatrics, 123(5), 1409-1411. https://doi.org/10.1542/peds.2009-0408

[7] Choudhary, A. K., Servaes, S., Slovis, T. L., Palusci, V. J., Hedlund, G. L., Narang, S. K., ... & Christian, C. W. (2018). Consensus statement on abusive head trauma in infants and young children. Pediatric Radiology, 48(8), 1048-1065. https://doi.org/10.1007/s00247-018-4149-1

[8] Narang, S. K., Estrada, C., Greenberg, S., & Lindberg, D. (2016). Acceptance of Shaken Baby Syndrome and Abusive Head Trauma as medical diagnoses. The Journal of Pediatrics, 177, 273-278. https://doi.org/10.1016/j.jpeds.2016.06.036

[9] van Zandwijk, J. P., Vester, M. E. M., Bilo, R. A., van Rijn, R. R., & Loeve, A. J. (2019). Modeling of inflicted head injury by shaking trauma in children: What can we learn?: Part II: A systematic review of mathematical and physical models. Forensic Science, Medicine, and Pathology, 15(3), 423-436. https://doi.org/10.1007/s12024-019-00093-7

[10] Huelke, D. F. (1998). An overview of anatomical considerations of infants and children in the adult world of automobile safety design. Annual Proceedings/Association for the Advancement of Automotive Medicine, 42, 93–113. http://europepmc.org/articles/PMC3400202/

[11] Vester, M. E. M., Bilo, R. A. C., Loeve, A. J., Van Rijn, R. R., & Van Zandwijk, J. P. (2019). Modeling of inflicted head injury by shaking trauma in children: what can we learn? Forensic Science, Medicine And Pathology, 15(3), 408–422. https://doi.org/10.1007/s12024-019-0082-3

[12] Jung, B., & Bhutta, B. S. (2020). Anatomy, Head and Neck, Neck Movements. StatPearls. https://pubmed.ncbi.nlm.nih.gov/32491487/

[13] McIntosh, L., McKenna, K., & Gustafsson, L. (2003). Active and Passive Shoulder Range of Motion in Healthy Older People. ˜The British Journal Of Occupational Therapy/British Journal Of Occupational Therapy, 66(7), 318–324. https://doi.org/10.1177/030802260306600706

[14] Schiks, L. A., Dankelman, J., & Loeve, A. J. (2020). Thresholds for the assessment of inflicted head injury by shaking trauma in infants: A systematic review. Forensic Science International, 306, 110060. https://doi.org/10.1016/j.forsciint.2019.110060

[15] Marsille, Y (2024). The structural mechanical properties of infant necks: Testing and surrogates :A systematic literature review

[16] Luck, J. F. (2012). The Biomechanics of the Perinatal, Neonatal and Pediatric Cervical Spine: Investigation of the Tensile, Bending and Viscoelastic Response. https://dukespace.lib.duke.edu/dspace/bitstream/10161/5850/1/Luck_duke_0066D_11591.pdf

[17] Sullivan, S., Coats, B., & Margulies, S. S. (2015). Biofidelic neck influences head kinematics of parietal and occipital impacts following short falls in infants. Accident Analysis and Prevention, 82, 143–153. https://doi.org/10.1016/j.aap.2015.05.020

[18] Öhman, A. M., & Beckung, E. R. (2008). Reference values for range of motion and muscle function of the neck in infants. Pediatric Physical Therapy, 20(1), 53–58. https://doi.org/10.1097/pep.0b013e31815ebb27

[19] Castle, K. B., Kernozek, T. W., & Warren, E. (2020). Two-dimensional versus three-dimensional measurement of infant cervical active motion. Physiotherapy Theory and Practice, 38(6), 805–817. https://doi.org/10.1080/09593985.2020.1790069[35] Yingling, V. R., Callaghan, J. P., & McGill, S. M. (1997). Dynamic loading affects the mechanical properties and failure site of porcine spines. Clinical Biomechanics, 12(5), 301–305. https://doi.org/10.1016/s0268-0033(97)00009-0

[20] Akpinar, S., Saboori, P., & Walker, G. (2015). Accelerations and Jerks Associated With Shaken Baby Syndrome. ASME. https://doi.org/10.1115/imece2015-52450

[21] Jenny, C. A., Bertocci, G., Fukuda, T., Rangarajan, N., & Shams, T. (2017). Biomechanical Response of the Infant Head to Shaking: An Experimental Investigation. Journal Of Neurotrauma, 34(8), 1579–1588. https://doi.org/10.1089/neu.2016.4687

[22] Q0 (6 week old, Newborn) User Manual. (n.d.). Humanetics.

[23] Fowler, R. M., Maselli, A., Pluimers, P., Magleby, S. P., & Howell, L. L. (2014). Flex-16: A large-displacement monolithic compliant rotational hinge. Mechanism And Machine Theory, 82, 203–217. https://doi.org/10.1016/j.mechmachtheory.2014.08.008

[24] Dearden, J., Grames, C., Orr, J., Jensen, B. D., Magleby, S. P., & Howell, L. L. (2017). Cylindrical cross-axis flexural pivots. *Precision Engineering*, *51*, 604–613. https://doi.org/10.1016/j.precisioneng.2017.11.001

[25] Van Boeijen, A., Daalhuizen, J., & Zijlstra, J. (2020). Delft Design Guide : Perspectives - Models - Approaches - Methods. https://research.tudelft.nl/en/publications/delft-design-guide-perspectives-models-approaches-methods

[26] ABAQUS, 2014. Abaqus Scripting Reference Guide, Dassault Systèmes Simulia Corp.; Providence, RI, USA

[27] Bambu Lab PETG-CF. (z.d.). https://eu.store.bambulab.com/nl-nl/collections/fiber-reinforced/products/petg-cf?variant=43982559412443

[28] Brown, D., Hanson, R., & Christian, W. (2025). Tracker Video Analysis and Modeling Tool (Version 6.3.0) [Computer software]. Retrieved April 9, 2025, from https://physlets.org/tracker/

[29] Economides, A. L., Islam, M. N., & Baxevanakis, K. P. (2024). Additively Manufactured Carbon Fibre PETG Composites: Effect of Print Parameters on Mechanical Properties. Polymers, 16(23), 3336. https://doi.org/10.3390/polym16233336

[30] Kasai, T., Ikata, T., Katoh, S., Miyake, R., & Tsubo, M. (1996). Growth of the Cervical Spine With Special Reference to Its Lordosis and Mobility. Spine, 21(18), 2067–2073. https://doi.org/10.1097/00007632-199609150-00003

[31] Howell, L. L., Magleby, S. P., & Olsen, B. M. (2013). Handbook of compliant mechanisms. In Wiley eBooks. http://cds.cern.ch/record/1616767

[32] Hopkins, J. B., & Culpepper, M. L. (2009). Synthesis of multi-degree of freedom, parallel flexure system concepts via Freedom and Constraint Topology (FACT) – Part I: Principles. Precision Engineering, 34(2), 259–270. https://doi.org/10.1016/j.precisioneng.2009.06.008

[33] ASTM A227 Class II Spring Steel :: MakeItFrom.com. (2020, 30 mei). https://www.makeitfrom.com/material-properties/ASTM-A227-Class-II-Spring-Steel

[34] Javid, F., Angeles, J., Pasini, D., & Cecere, R. (2012). Shape Optimization of a Self-deployable Anchor Designed for Percutaneous Mitral Valve Repair. Journal Of Medical Devices, 6(1). https://doi.org/10.1115/1.4005780

[35] Injection 1018I | UBE Corporation. (z.d.). UBE Corporation. https://www.ube.com/ube/en/contents/chemical/nylon/1018i.html

[36] Alhayek, A., Syamsir, B. A., Anggraini, V., Muda, Z. C., & Nor, N. M. (2020). Design enhancement of sustainable glass fiber reinforced polymer (GFRP) cross arm. Journal Of Green Engineering, 10(12), 13491–13507. https://research.monash.edu/en/publications/design-enhancement-of-sustainable-glass-fiber-reinforced-polymer-

[37] Callister, W. (2000). Fundamentals of Materials Science and Engineering. http://ci.nii.ac.jp/ncid/BB10536675

[38] ASM Material Data Sheet. (z.d.-b). https://asm.matweb.com/search/specificmaterial.asp?bassnum=mq304a

[39] Boyi Technology. (2025, 14 april). Acrylic (PMMA) CNC Machining Services | Milling & Cutting Parts - BOYI. BOYI. https://www.boyiprototyping.com/cnc-machining/plastics/acrylic/

Appendix

A. Concept Selection

The first concept idea exists of neck endplates with one or multiple joints in between them for the ROM, and springs to provide stiffness. To achieve the ROM, the following joints were examined:

- *Ball socket joint (3DOF)*
- *Cardan joint (2DOF)*
- *Hinge (1DOF)*

To achieve the stiffness, the following springs were examined:

- *Compression/extension springs*
- *Torsion springs*
- *Leaf springs*

Concept prototypes were modeled in SolidWorks, after which it turns out multiple stages of spring joint combinations must be used to achieve the desired ROM, which can be seen in Figure 1. Furthermore, by using springs, every stage would only allow for 1 tunable DOF due to the preferred direction of deflection of the springs, which means multiple stages must be used to obtain more than 1 DOF. In short, if a design with 3DOFs and the desired ROM needs to be realized, multiple stages need to be used. This would be undesirable as space is limited due to length constraints. If the surrogate neck would be longer than the original Q0 dummy neck and actual infant neck, the head kinematics will not be mimic the actual kinematics, leading to inaccurate shaking simulation results.

Figure 1: SolidWorks model of a concept surrogate neck design using a ball and socket joint in combination with torsion springs, including Q0 mounts

Compliant mechanism

Another possible solution set might arise from existing compliant mechanism designs, beside the compliant mechanism concept with joints and springs. Compliant mechanisms are mechanisms which transfer force and motion through elastic deformation of materials used in the mechanism [31]. The Q0 dummy neck can be seen as a compliant mechanism itself, due to the rubber elastically deforming. However, when used in shaking simulations, the rubber may deform plastically or break. Furthermore, this material will age and lose its initial mechanical properties. This explains the wish to use a neck mechanism built from non-degrading materials for their structural integrity.

Several existing compliant mechanism designs were examined on their usefulness in this context, as well as existing design frameworks for compliant mechanisms using the Handbook of Compliant mechanisms. However, using the designs described in the Handbook of Compliant mechanisms, only small deformations can be obtained. Furthermore, the FACT design methodology was investigated on how to design a 3DOF compliant mechanism. [32] Designs with 3 DOFs were found, while the designs are unable to reach the desired deformation of 90 degrees.

Therefore, existing high rotation compliant mechanisms were researched and resulted in two possible compliant mechanisms able to make a rotation of 90 degrees.

Firstly, a possible working mechanism was found in research by Fowler et al. [23]. They designed a compliant monolithic hinge for deploying solar panels in space, depicted in Figure 2 and 3. Key advantages stated in this research were:

- Stiffness of the hinge can be influenced by different geometric and material parameters using a computational model
- The hinge can achieve a rotation of 90 degrees
- The hinge can be manufactured using planar manufacturing methods

The main disadvantage stated is the low off-axis stiffness. While this is a problem for their application, it is desirable to have a low off-axis stiffness and therefore the possibility to have motion outside the bending plane.

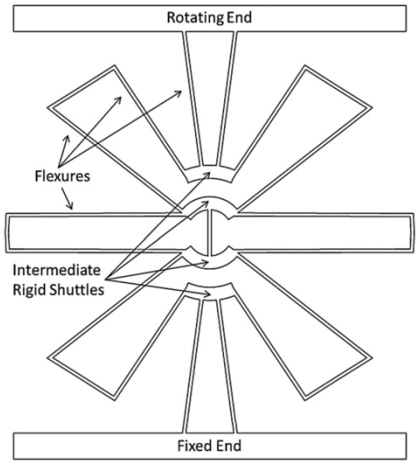

Figure 2: Shape of the monolithic compliant hinge by Fowler et al. [23]

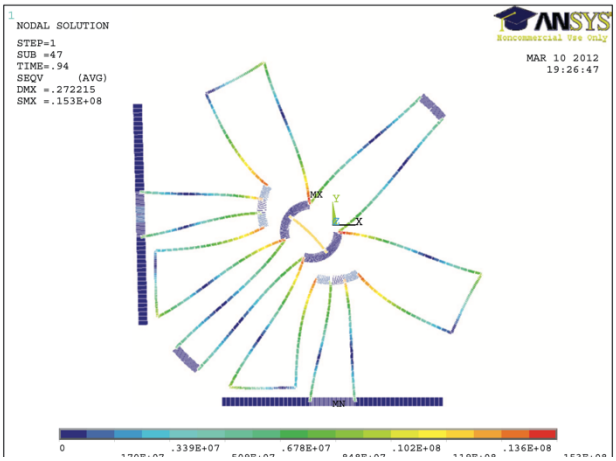

Figure 3: Deflected shape of the monolithic hinge by Fowler et al. [23]

A second possible working mechanism was found in research of Dearden et al., describing the design of a cylindrical cross-axis flexural pivot [29]. This pivot was able to reach a deflection angle of 9 to 85 degrees when manufactured from steel and nitinol, respectively. A picture of this pivot can be found in Figure 5. This means that an elastic material will not provide the desired range of motion, meaning the pivots must be stacked. Furthermore, this mechanism was more complex to manufacture compared to the design of Fowler et al., as it is no planar part.

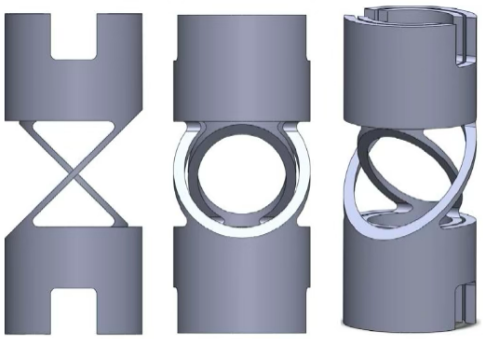

Figure 4: The cylindrical cross-axis flexural pivot by Dearden et al. [24]

Harris profiles and the selection of the final concept can be found in Section 2.1.

B. Abaqus Code

```
from abaqus import *
from abaqusConstants import *
from visualization import *
import math
import part
import step
import load
import mesh
import interaction
import section
import odbAccess

#Geometric parametres PETGCF-4
R = 35E-3 #[m]
t1 = 0.8E-3 #[m]
t2 = 2E-3 #[m]
w = 10E-3 #[m]
Midden = True

materiaal = 'PLA'
Emod = 2.75E9 #[N/m^2]
Poisson = 0.33

#Mesh Parametres
MeshSize = 0.00063

#Simulation parametres
RotationValue = 1.7
TimeValue = 0.2

#Angles and length factors
if Midden is True:
   angles = {'a1': 7, 'a2': 22, 'a3': 34, 'a4': 45, 'a5': 56, 'a6': 57, 'a7': 85, 'a8': 90}
else:
   angles = {'a1': 10, 'a2': 25, 'a3': 39, 'a4': 46, 'a5': 49, 'a6': 55, 'a7': 84, 'a8': 90}

if Midden is True:
   factors = {'f1': 1.05, 'f2': 0.31, 'f3': 0.94, 'f4': 1.25, 'f5': 0.95, 'f6': 1.0, 'f7': 0.15, 'f8': 0.85}
else:
   factors = {'f1': 1.05, 'f2': 0.35, 'f3': 0.88, 'f4': 1.25, 'f5': 0.94, 'f6': 0.86, 'f7': 0.2, 'f8': 0.9}

#Degrees to rad
angles_rad = {key: math.radians(value) for key, value in angles.items()}
#Keypoints cartesian coordinates
keypoints = {
    'k1': {'X': 0, 'Y': 0},
   'k2': {'X': 0, 'Y': R},
   'k3': {'X': -factors['f1'] * R * math.sin(angles_rad['a1']), 'Y': R},
   'k4': {'X': -factors['f2'] * R * math.sin(angles_rad['a1']), 'Y': factors['f2'] * R * math.cos(angles_rad['a1'])},
   'k5': {'X': -factors['f2'] * R * math.sin(angles_rad['a2']), 'Y': factors['f2'] * R * math.cos(angles_rad['a2'])},
   'k6': {'X': -factors['f3'] * R * math.sin(angles_rad['a3']), 'Y': factors['f3'] * R * math.cos(angles_rad['a3'])},
   'k7': {'X': -factors['f4'] * R * math.sin(angles_rad['a3']), 'Y': R},
   'k8': {'X': -factors['f5'] * R * math.sin(angles_rad['a4']), 'Y': factors['f5'] * R * math.cos(angles_rad['a4'])},
   'k9': {'X': -factors['f6'] * R * math.sin(angles_rad['a5']), 'Y': factors['f6'] * R * math.cos(angles_rad['a5'])},
   'k10': {'X': -factors['f7'] * R * math.sin(angles_rad['a6']), 'Y': factors['f7'] * R * math.cos(angles_rad['a6'])},
   'k11': {'X': -factors['f8'] * R * math.sin(angles_rad['a7']), 'Y': factors['f8'] * R * math.cos(angles_rad['a7'])},
   'k12': {'X': -factors['f8'] * R, 'Y': 0},
   'k13': {'X': 0, 'Y': -R},
   'k14': {'X': -factors['f1'] * R * math.sin(angles_rad['a1']), 'Y': -R},
   'k15': {'X': -factors['f2'] * R * math.sin(angles_rad['a1']), 'Y': -factors['f2'] * R * math.cos(angles_rad['a1'])},
   'k16': {'X': -factors['f2'] * R * math.sin(angles_rad['a2']), 'Y': -factors['f2'] * R * math.cos(angles_rad['a2'])},
   'k17': {'X': -factors['f3'] * R * math.sin(angles_rad['a3']), 'Y': -factors['f3'] * R * math.cos(angles_rad['a3'])},
   'k18': {'X': -factors['f4'] * R * math.sin(angles_rad['a3']), 'Y': -R},
   'k19': {'X': -factors['f5'] * R * math.sin(angles_rad['a4']), 'Y': -factors['f5'] * R * math.cos(angles_rad['a4'])},
   'k20': {'X': -factors['f6'] * R * math.sin(angles_rad['a5']), 'Y': -factors['f6'] * R * math.cos(angles_rad['a5'])},
   'k21': {'X': -factors['f7'] * R * math.sin(angles_rad['a6']), 'Y': -factors['f7'] * R * math.cos(angles_rad['a6'])},
   'k22': {'X': -factors['f8'] * R * math.sin(angles_rad['a7']), 'Y': -factors['f8'] * R * math.cos(angles_rad['a7'])},
   'k23': {'X': factors['f1'] * R * math.sin(angles_rad['a1']), 'Y': R},
   'k24': {'X': factors['f2'] * R * math.sin(angles_rad['a1']), 'Y': factors['f2'] * R * math.cos(angles_rad['a1'])},
   'k25': {'X': factors['f2'] * R * math.sin(angles_rad['a2']), 'Y': factors['f2'] * R * math.cos(angles_rad['a2'])},
   'k26': {'X': factors['f3'] * R * math.sin(angles_rad['a3']), 'Y': factors['f3'] * R * math.cos(angles_rad['a3'])},
   'k27': {'X': factors['f4'] * R * math.sin(angles_rad['a3']), 'Y': R},
   'k28': {'X': factors['f5'] * R * math.sin(angles_rad['a4']), 'Y': factors['f5'] * R * math.cos(angles_rad['a4'])},
   'k29': {'X': factors['f6'] * R * math.sin(angles_rad['a5']), 'Y': factors['f6'] * R * math.cos(angles_rad['a5'])},
   'k30': {'X': factors['f7'] * R * math.sin(angles_rad['a6']), 'Y': factors['f7'] * R * math.cos(angles_rad['a6'])},
   'k31': {'X': factors['f8'] * R * math.sin(angles_rad['a7']), 'Y': factors['f8'] * R * math.cos(angles_rad['a7'])},
   'k32': {'X': factors['f8'] * R, 'Y': 0},
   'k33': {'X': factors['f1'] * R * math.sin(angles_rad['a1']), 'Y': -R},
```

```
    'k34': {'X': factors['f2'] * R * math.sin(angles_rad['a1']), 'Y': -factors['f2'] * R * math.cos(angles_rad['a1'])},
    'k35': {'X': factors['f2'] * R * math.sin(angles_rad['a2']), 'Y': -factors['f2'] * R * math.cos(angles_rad['a2'])},
    'k36': {'X': factors['f3'] * R * math.sin(angles_rad['a3']), 'Y': -factors['f3'] * R * math.cos(angles_rad['a3'])},
    'k37': {'X': factors['f4'] * R * math.sin(angles_rad['a3']), 'Y': -R},
    'k38': {'X': factors['f5'] * R * math.sin(angles_rad['a4']), 'Y': -factors['f5'] * R * math.cos(angles_rad['a4'])},
    'k39': {'X': factors['f6'] * R * math.sin(angles_rad['a5']), 'Y': -factors['f6'] * R * math.cos(angles_rad['a5'])},
    'k40': {'X': factors['f7'] * R * math.sin(angles_rad['a6']), 'Y': -factors['f7'] * R * math.cos(angles_rad['a6'])},
    'k41': {'X': factors['f8'] * R * math.sin(angles_rad['a7']), 'Y': -factors['f8'] * R * math.cos(angles_rad['a7'])},
    'k42': {'X': -factors['f7'] * R * math.sin(angles_rad['a7']), 'Y': factors['f7'] * R * math.cos(angles_rad['a7'])},
    'k43': {'X': factors['f7'] * R * math.sin(angles_rad['a7']), 'Y': factors['f7'] * R * math.cos(angles_rad['a7'])},
    'k44': {'X': -factors['f7'] * R * math.sin(angles_rad['a7']), 'Y': -factors['f7'] * R * math.cos(angles_rad['a7'])},
    'k45': {'X': factors['f7'] * R * math.sin(angles_rad['a7']), 'Y': -factors['f7'] * R * math.cos(angles_rad['a7'])},
    'k46': {'X': 0, 'Y': factors['f7'] * R},
    'k47': {'X': 0, 'Y': -factors['f7'] * R}
}


#Start Model
model_name = 'Flex'
mdb.Model(name=model_name)
model = mdb.models[model_name]
part_name = 'Flex'
part = model.Part(name=part_name, dimensionality=TWO_D_PLANAR, type=DEFORMABLE_BODY)

#New sketch
sketch = model.ConstrainedSketch(name='KeypointSketch', sheetSize=200.0)

#Add keypoints
keypoint_coords = {key: (coords['X'], coords['Y']) for key, coords in keypoints.items()}

#Add lines
if Midden is True:
    connection_pairs = [
        ('k3', 'k4'), ('k23', 'k24'),  #Top
        ('k8', 'k6'), ('k8','k9'), ('k6', 'k5'), ('k9', 'k10'),    #Left
        ('k26', 'k28'), ('k28', 'k29'), ('k29', 'k30'), ('k25', 'k26'),   #Right
        ('k11', 'k12'), ('k12', 'k22'), #Left Vertical
        ('k31', 'k32'), ('k32', 'k41'), #Right Vertical
        ('k14', 'k15'), ('k33', 'k34'),   #Bottom Vertical
        ('k19', 'k17'), ('k17', 'k16'), ('k19', 'k20'), ('k21', 'k20'),  #Bottom Left
        ('k36', 'k38'), ('k38', 'k39'), ('k39', 'k40'), ('k36', 'k35'),  #Bottom Right
        ('k40', 'k41'), ('k7', 'k3'), ('k3', 'k2'), ('k2', 'k23'), ('k23', 'k27'), ('k10', 'k11'), ('k30', 'k31'), ('k21', 'k22'), ('k18', 'k14'), ('k14', 'k13'), ('k13', 'k33'), ('k33', 'k37'),
# Horizontalen  # Bovenste balk
        ('k46', 'k47') #Mid Flexure
    ]
else:
     connection_pairs = [
        ('k3', 'k4'), ('k23', 'k24'),  #Top
        ('k8', 'k6'), ('k8','k9'), ('k6', 'k5'), ('k9', 'k10'),    #Left
        ('k26', 'k28'), ('k28', 'k29'), ('k29', 'k30'), ('k25', 'k26'),   #Right
        ('k11', 'k12'), ('k12', 'k22'), #Left vertical
        ('k31', 'k32'), ('k32', 'k41'), #Right Vertical
        ('k14', 'k15'), ('k33', 'k34'),   #Bottom Vertical
        ('k19', 'k17'), ('k17', 'k16'), ('k19', 'k20'), ('k21', 'k20'),  #Bottom Left
        ('k36', 'k38'), ('k38', 'k39'), ('k39', 'k40'), ('k36', 'k35'),  #Bottom Right
        ('k40', 'k41'), ('k7', 'k3'), ('k3', 'k2'), ('k2', 'k23'), ('k23', 'k27'), ('k10', 'k11'), ('k30', 'k31'), ('k21', 'k22'), ('k18', 'k14'), ('k14', 'k13'), ('k13', 'k33'), ('k33', 'k37')
#Horizontals
    ]

for start, end in connection_pairs:
    sketch.Line(point1=keypoint_coords[start], point2=keypoint_coords[end])

#Add arcs
arc_connections = [
    ('k10', 'k30'),   # Arc connecting k10 -> k46 -> k30 with center at k1
    ('k5', 'k25'),#, 'k24') #, 'k25'),    # Arc connecting k5 -> k4 -> k24 with center at k1
    ('k40', 'k21'),#, 'k47'),   # Arc connecting k25 -> k16 -> k15 with center at k1
    ('k35', 'k16'),#, 'k15')#, 'k16'),   # Arc connecting k34 -> k35 -> k21 with center at k1
]
 for start, end in arc_connections:
    sketch.ArcByCenterEnds(center=keypoint_coords['k1'],
                           point1=keypoint_coords[start],
                           point2=keypoint_coords[end],
                           direction=CLOCKWISE)

#Create part
part.BaseWire(sketch=sketch)

#Create Sets
if Midden:
```

```
    t1_edge_indices = [0,2,3,4,5,6,7,8,9,11,14,15,16,17,21,26,27,28,29,30,31,32,33,34,35,37,40,42,43,47,48]
    t2_edge_indices = [1,10,13,18,19,20,22,36,41,44]
    top_edge_indices = [12,23,24,25]
    bottom_edge_indices = [38,39,45,46]
else:
    t1_edge_indices = [0,2,3,4,5,7,8,9,10,12,13,14,15,21,22,23,24,26,27,28,29,30,34,35,36,37,38,41,44,45,46,47,48]
    t2_edge_indices = [1,6,11,20,25,31,32,33]
    top_edge_indices = [39,40,42,43]
    bottom_edge_indices = [16,17,18,19]

edges_t1 = [i for i in t1_edge_indices]
part.Set(name='t1', edges=[part.edges[i:i+1] for i in edges_t1])
regiont1 = [part.edges[i:i+1] for i in edges_t1]

edges_t2 = [i for i in t2_edge_indices]
part.Set(name='t2', edges=[part.edges[i:i+1] for i in edges_t2])
regiont2 = [part.edges[i:i+1] for i in edges_t2]

edges_top = [i for i in top_edge_indices]
part.Set(name='Top', edges=[part.edges[i:i+1] for i in edges_top])
regiontop = [part.edges[i:i+1] for i in edges_top]

edges_bottom = [i for i in bottom_edge_indices]
part.Set(name='Bottom', edges=[part.edges[i:i+1] for i in edges_bottom])
regionbottom = [part.edges[i:i+1] for i in edges_bottom]

#Define Output Requests
if Midden is True:
    sensor = [22]
else:
    sensor = [36]

#Make assembly
a = model.rootAssembly

#Material defining
model.Material(name=materiaal)
model.materials[materiaal].Elastic(table=((Emod, Poisson),))

#Beam sections
section_t1 = 't1'
section_t2 = 't2'
section_Bottom = 'Bottom'
section_Top = 'Top'
model.RectangularProfile(name='t1', a=w, b=t1)
model.RectangularProfile(name='t2', a=w, b=t2)
model.BeamSection(name=section_t1, profile='t1', material=materiaal, integration=DURING_ANALYSIS, poissonRatio = Poisson)
model.BeamSection(name=section_t2, profile='t2', material=materiaal, integration=DURING_ANALYSIS, poissonRatio = Poisson)
model.BeamSection(name=section_Bottom, profile='t2', material=materiaal, integration=DURING_ANALYSIS, poissonRatio = Poisson)
model.BeamSection(name=section_Top, profile='t2', material=materiaal, integration=DURING_ANALYSIS, poissonRatio = Poisson)

#Section Assignments
part.SectionAssignment(region=regiont1, sectionName='t1')
part.SectionAssignment(region=regiont2, sectionName='t2')
part.SectionAssignment(region=regiontop, sectionName='Bottom')
part.SectionAssignment(region=regionbottom, sectionName='Top')

part.seedPart(size=MeshSize, deviationFactor = 0.1,minSizeFactor = 0.1)

part.generateMesh()

#Beam orientation
orientation_vector = (0.0, 0.0, -1.0)
part.assignBeamSectionOrientation(region=regiont1, method=N1_COSINES, n1=orientation_vector)
part.assignBeamSectionOrientation(region=regiont2, method=N1_COSINES, n1=orientation_vector)
part.assignBeamSectionOrientation(region=regiontop, method=N1_COSINES, n1=orientation_vector)
part.assignBeamSectionOrientation(region=regionbottom, method=N1_COSINES, n1=orientation_vector)

#Generate instance
inst = a.Instance(dependent=ON, name = 'I', part = part)

node = [i for i in sensor]
part.Set(name='Sensor', nodes=[part.nodes[i:i+1] for i in sensor])

if Midden is True:
    BottomNode = [34,35,36,40,41]
    TopNode = [22,24,13]
else:
    BottomNode = [0,16,17,18,19]
    TopNode = [22,24,13]
```

```
node2 = [i for i in BottomNode]
part.Set(name='BottomNode', nodes=[part.nodes[i:i+1] for i in BottomNode])

node3 = [i for i in TopNode]
part.Set(name='TopNode', nodes=[part.nodes[i:i+1] for i in TopNode])

#Add step
step_name = 'Static'
model.StaticStep(name=step_name, previous='Initial', description='Static general step for loading.', nlgeom = ON)
model.steps[step_name].setValues(timePeriod=1, maxNumInc=50, initialInc=0.02, timeIncrementationMethod = FIXED)

#Add boundary conditions
model.EncastreBC(name='VastOnder', createStepName=step_name,
        region=a.instances['I'].sets['BottomNode'])

model.DisplacementBC(name='RotatieBoven', createStepName=step_name, region=a.instances['I'].sets['Sensor'], u1=UNSET, u2=UNSET, u3=UNSET,
ur1=UNSET, ur2=UNSET, ur3=RotationValue)

#Delete earlier data
for xy_data_name in list(session.xyDataObjects.keys()):
  del session.xyDataObjects[xy_data_name]

for plot_name in list(session.xyPlots.keys()):
  del session.xyPlots[plot_name]

#Delete earlier job
job_name = 'Job-1'
if job_name in mdb.jobs.keys():
  del mdb.jobs[job_name]

#Run simulation
mdb.Job(name=job_name, model='Flex', type=ANALYSIS)
mdb.jobs[job_name].submit()
mdb.jobs[job_name].waitForCompletion()

odb_path = f"{job_name}.odb"
if odb_path in session.odbs:
  session.odbs[odb_path].close()
odb = session.openOdb(name=odb_path)

# Access the instance and node set for data extraction
instance = odb.rootAssembly.instances['I']
sensor_set_name = 'SENSOR'

if sensor_set_name not in instance.nodeSets.keys():
  raise ValueError(f"The set '{sensor_set_name}' is not found in instance 'I'.")

sensor_node_set = instance.nodeSets[sensor_set_name]

#Data extraction
step_name = 'Static'
step = odb.steps[step_name]
ur3_values = []
rm3_values = []
mises_values = []
time_values = []

for frame in step.frames:
  time = frame.frameValue
  time_values.append(time)

  #Rotation data
  ur3_data = frame.fieldOutputs['UR3']
  ur3_subset = ur3_data.getSubset(region=sensor_node_set)
  ur3_avg_value = sum([value.data for value in ur3_subset.values]) / len(ur3_subset.values)
  ur3_values.append(ur3_avg_value)

  #Moment data
  rm3_data = frame.fieldOutputs['RM3']
  rm3_subset = rm3_data.getSubset(region=sensor_node_set)
  rm3_avg_value = sum([value.data for value in rm3_subset.values]) / (len(rm3_subset.values))
  rm3_values.append(rm3_avg_value)

#Stress data
  mises_data = frame.fieldOutputs['S']  # Field output includes stress components
  mises_max_value = max(value.mises for value in mises_data.values if hasattr(value, 'mises'))
  mises_values.append(mises_max_value)

odb.close()
```

```
#Moment vs rotation
combined_xy_data = session.XYData(data=list(zip(ur3_values, rm3_values)), name='Moment vs Rotation', xValuesLabel ='Rotation (rad)', yValuesLabel =
'Moment (Nm)')
#Mises stress vs Time
combined_xy_data_mises_time = session.XYData(data=list(zip(time_values, mises_values)),name= 'Max Mises Stress vs Time', xValuesLabel = 'Time (s)'
, yValuesLabel = 'Stress (MPa)')
#Mises stress vs rotation
combined_xy_data_mises_rotation = session.XYData(data=list(zip(ur3_values, mises_values)),name='Max Mises Stress vs Rotation', xValuesLabel =
'Rotation (rad)', yValuesLabel = 'Stress (MPa)')

#Calculate the slope of the combined XYData (Moment vs Rotation) and print
#Check if there is more than one data point to calculate the slope
if len(ur3_values) > 1:
  delta_rotation = ur3_values[-1] - ur3_values[0]
  delta_moment = rm3_values[-1] - rm3_values[0]

  if delta_rotation != 0:
    slope = delta_moment / (delta_rotation)
    print(f"The stiffness of the part is {slope} N*m/rad for t1 = {t1*1000} mm")
  else:
    print("Error: Division by zero encountered in slope calculation (no change in rotation).")
else:
  print("Not enough data points to calculate slope.")

#Print Max Stress
maxstress = mises_max_value
print(f"The max stress is {maxstress} MPa")
```

C. Mesh validation

Table 1: Geometric and material properties used by Fowler er al. [23] The ABAQUS model is validated using these input parameters for different prototypes PP-1, PP-2, Ti-1, CNT-1 and CNT-2 by comparing stiffness and stress results from both models. R, t1, t2, and w represent the geometric parameters. a1-a8 and f1-f8 decide the angles and lengths of the flexures. Material properties by Fowler et al are adopted.

Parameter	PP-1	PP-2	Ti-1	CNT-1	CNT-2
R [mm]	152.4	152.4	66.6	4	2
t1 [mm]	2.5	2.5	0.7	0.030	0.020
t2 [mm]	8.0	8.0	3.0	0.100	0.080
w [mm]	12.7	12.7	12.7	0.250	0.250
a1	10°	7°	7°	10°	10°
a2	25°	22°	22°	25°	25°
a3	39°	30°	34°	39°	39°
a4	46°	41°	45°	46°	46°
a5	49°	52°	56°	49°	49°
a6	55°	52°	57°	55°	55°
a7	84°	85°	85°	84°	84°
a8	90°	90°	90°	90°	90°
f1	1.05	1.05	1.05	1.05	1.05
f2	0.35	0.31	0.29	0.35	0.35
f3	0.88	0.94	0.94	0.88	0.88
f4	1.25	1.25	1.25	1.25	1.25
f5	0.94	0.95	0.95	0.94	0.94
f6	0.86	1.0	1.0	0.86	0.86
f7	0.20	0.15	0.15	0.20	0.20
f8	0.90	1.0	1.0	1.0	1.0
Yield Strength [MPa]	31	31	827	105	105
Young's Modulus [GPa]	1.379	1.379	113.8	6	6
Poisson's ratio [-]	0.3	0.3	0.34	0.28	0.28

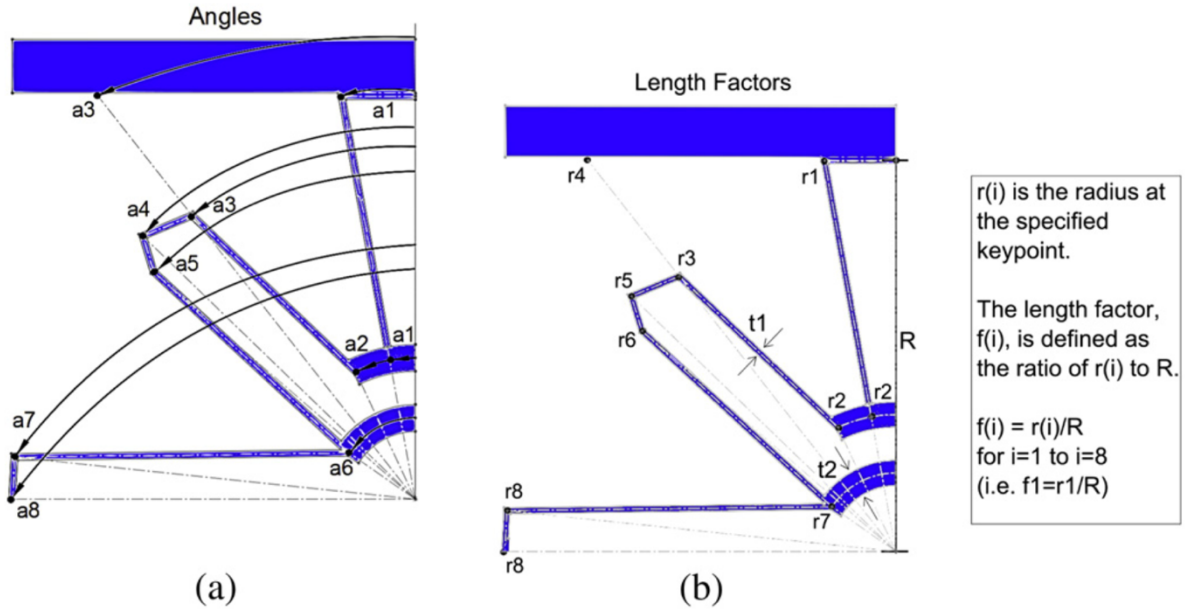

Figure 5: Geometry determining angles (a) and length factors (b), adopted from Fowler et al. [23]

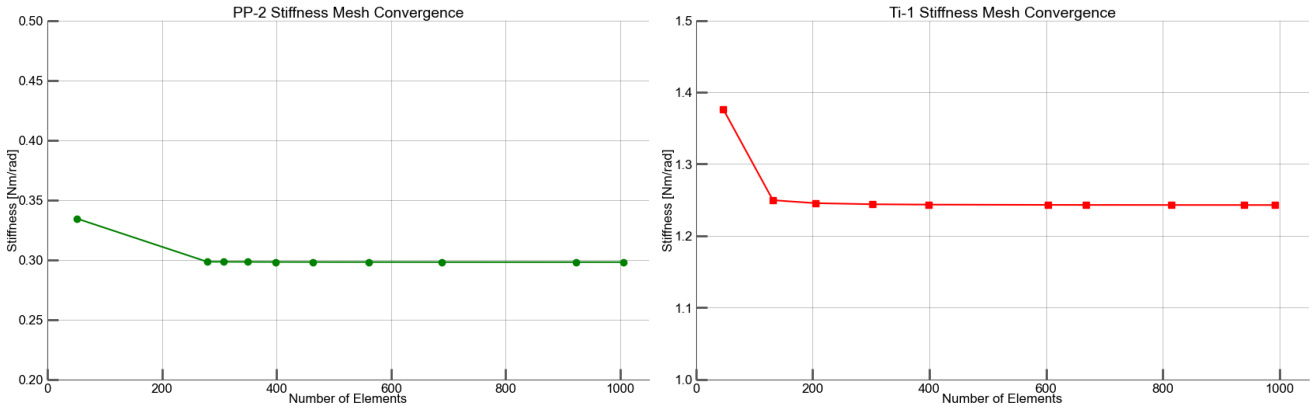

Figure 6: Mesh convergence test for stiffness using prototypes PP-2 and Ti-1 by Fowler et al. [23]

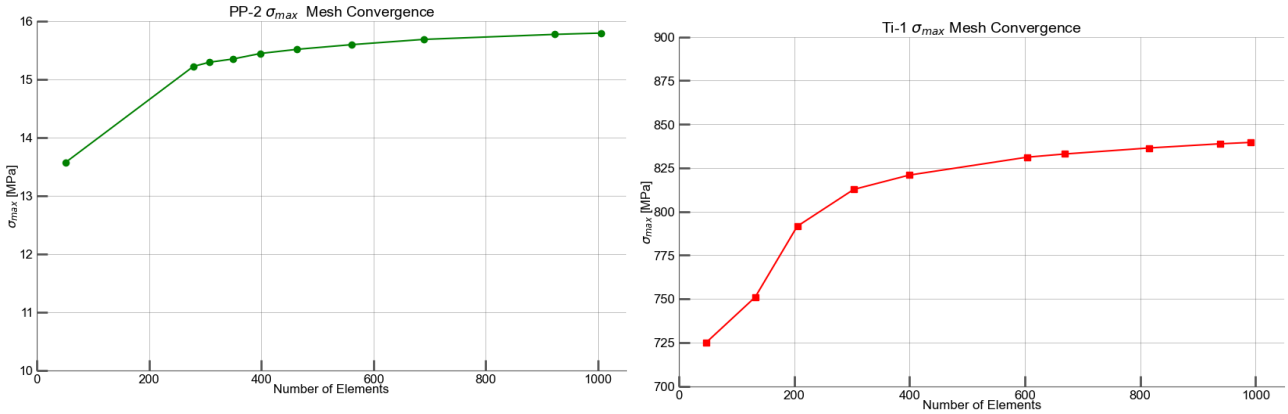

Figure 7: Mesh convergence test for maximum stress using prototypes PP-2 and Ti-1 by Fowler et al. [23]

D. Simulation results

Table 2: Stiffness and maximum stress simulation results using spring steel, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
Spring Steel	190	1850	0.39

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
30	1.2	1.711	4996.49	0.37	38.02
30	1.1	1.336	4646.65	0.4	40.88
30	1	1.016	4277.27	0.43	44.41
30	0.9	0.748	3890.57	0.48	48.83
30	0.8	0.53	3488.87	0.53	54.45
30	0.7	0.358	3074.55	0.6	61.79
30	0.6	0.227	2649.97	0.7	71.7
30	0.5	0.132	2217.41	0.83	85.68
35	1.2	1.478	4312.29	0.43	44.05
35	1.1	1.153	4005.7	0.46	47.43
35	1	0.876	3683.3	0.5	51.58
35	0.9	0.644	3347.01	0.55	56.76
35	0.8	0.456	2998.8	0.62	63.35
35	0.7	0.307	2640.66	0.7	71.95
35	0.6	0.195	2274.53	0.81	83.53
35	0.5	0.113	1902.24	0.97	99.88

Table 3: Stiffness and maximum stress simulation results using austenitic nitinol, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
Nitinol (Austenite)	70	600	0.33

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
20	0.8	0.288	1900.71	0.32	36.83
21	0.8	0.275	1813.61	0.33	38.6
22	0.8	0.263	1734.69	0.35	40.35
23	0.8	0.252	1662.99	0.36	42.09
24	0.8	0.242	1596.86	0.38	43.84
25	0.8	0.233	1534.39	0.39	45.62
30	0.8	0.195	1285.37	0.47	54.46
40	0.8	0.147	968.39	0.62	72.28
50	0.8	0.118	775.47	0.77	90.27
60	0.8	0.099	646.5	0.93	108.28
75	0.8	0.079	517.53	1.16	135.26

Table 4: Stiffness and maximum stress simulation results using martensitic nitinol, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
Nitinol (Martensite)	40	120	0.33

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
20	0.8	0.165	1086.12	0.11	36.83
21	0.8	0.157	1036.35	0.116	38.6
22	0.8	0.15	992.15	0.121	40.32
23	0.8	0.144	950.7	0.126	42.07
24	0.8	0.138	912.49	0.132	43.84
25	0.8	0.133	876.79	0.137	45.62
30	0.8	0.112	734.5	0.163	54.46
40	0.8	0.084	552.59	0.217	72.39

Table 5: Stiffness and maximum stress simulation results using PLA, including failure ratio and shape factor

Material	E **(GPa)**	σ_y **(MPa)**	**Poisson's ratio (-)**
PLA (Bambu)	3	50	0,33

R (mm)	**t1 (mm)**	**Stiffness (Nm/rad)**	**Max Stress (MPa)**	$\frac{\sigma_y}{\sigma_{max}}$ **(-)**	$\frac{E}{\sigma_{max}}$ **(-)**
40	1.2	0.021	59.76	0.84	50.20
50	1.2	0.017	48.05	1.04	62.44
60	1.2	0.014	40.14	1.25	74.73
70	1.2	0.012	34.46	1.45	87.06
25	1.2	0.032	93.55	0.53	32.07
25	1.1	0.025	87.16	0.57	34.42
25	1	0.019	80.38	0.62	37.32
25	0.9	0.014	73.23	0.68	40.97
25	0.8	0.010	65.76	0.76	45.62
25	0.7	0.007	58.02	0.86	51.70
25	0.6	0.004	50.07	1.00	59.92
25	0.5	0.002	41.93	1.19	71.55
30	1.2	0.027	78.89	0.63	38.03
30	1.1	0.021	73.37	0.68	40.89
30	1	0.016	67.54	0.74	44.42
30	0.9	0.012	61.43	0.81	48.84
30	0.8	0.008	55.09	0.91	54.46
30	0.7	0.006	48.55	1.03	61.80
30	0.6	0.004	41.84	1.19	71.70
30	0.5	0.002	35.01	1.43	85.69

Table 6: Stiffness and maximum stress simulation results using carbon-fiber reinforced PLA, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
PLA-CF (Bambu)	3.7	114	0.4

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
20	0.8	0.015	100.42	1.14	36.84
21	0.8	0.015	95.87	1.19	38.59
22	0.8	0.014	91.7	1.24	40.35
23	0.8	0.013	87.87	1.3	42.11
24	0.8	0.013	84.34	1.35	43.87
25	0.8	0.012	81.08	1.41	45.63
30	0.8	0.01	67.93	1.68	54.47
40	0.8	0.008	51.11	2.23	72.4
50	0.8	0.006	40.97	2.78	90.32
60	0.8	0.005	34.17	3.34	108.28
75	0.8	0.004	27.35	4.17	135.27
20	1.2	0.048	141.14	0.81	26.22
20	1.1	0.038	131.94	0.86	28.04
20	1	0.029	122.05	0.93	30.32
20	0.9	0.021	111.52	1.02	33.18
20	0.8	0.015	100.42	1.14	36.84
20	0.7	0.01	88.82	1.28	41.65
20	0.6	0.007	76.8	1.48	48.18
20	0.5	0.004	64.43	1.77	57.42
25	1.2	0.039	115.3	0.99	32.09
25	1.1	0.031	107.44	1.06	34.44
25	1	0.024	99.08	1.15	37.34
25	0.9	0.017	90.28	1.26	40.98
25	0.8	0.012	81.08	1.41	45.63
25	0.7	0.008	71.55	1.59	51.71
25	0.6	0.005	61.74	1.85	59.93
25	0.5	0.003	51.71	2.2	71.55
30	1.2	0.033	97.25	1.17	38.05
30	1.1	0.026	90.45	1.26	40.91
30	1	0.02	83.27	1.37	44.44
30	0.9	0.015	75.74	1.51	48.85
30	0.8	0.01	67.93	1.68	54.47
30	0.8	0.015	100.42	1.14	36.84
30	0.8	0.015	95.87	1.19	38.59
30	0.8	0.014	91.7	1.24	40.35

Table 7: Stiffness and maximum stress simulation results using Nylon, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
Nylon	1.4	50	0.39

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
20	1.2	0.018	53.41	0.94	26.21
21	1.2	0.017	51.14	0.98	27.38
22	1.2	0.017	49.04	1.02	28.55
23	1.2	0.016	47.1	1.06	29.72
24	1.2	0.015	45.3	1.1	30.9
30	1.2	0.013	36.8	1.36	38.02
30	1.1	0.01	34.23	1.46	40.88
30	1	0.007	31.51	1.59	44.41
30	0.9	0.006	28.66	1.74	48.83
30	0.8	0.004	25.7	1.94	54.45
30	0.7	0.003	22.65	2.21	61.79
30	0.6	0.002	19.52	2.56	71.7
30	0.5	0. 001	16.34	3.06	85.68
35	1.2	0.011	31.76	1.57	44.05
35	1.1	0.008	29.51	1.69	47.43
35	1	0.006	27.13	1.84	51.58
35	0.9	0.005	24.66	2.03	56.76
35	0.8	0.003	22.09	2.26	63.35
35	0.7	0.002	19.46	2.57	71.95
35	0.6	0.001	16.76	2.98	83.53
35	0.5	0.001	14.02	3.57	99.88

Table 8: Stiffness and maximum stress simulation results using ABS, including failure ratio and shape factor

Material	E (GPa)	σ_y (MPa)	Poisson's ratio (-)
ABS (Bambu)	3.7	114	0.4

R (mm)	t1 (mm)	Stiffness (Nm/rad)	Max Stress (MPa)	$\frac{\sigma_y}{\sigma_{max}}$ (-)	$\frac{E}{\sigma_{max}}$ (-)
20	0.8	0.006	40.74	0.95	36.82
21	0.8	0.006	38.89	0.99	38.57
22	0.8	0.006	37.21	1.03	40.31
23	0.8	0.005	35.66	1.08	42.07
24	0.8	0.005	34.22	1.12	43.83
25	0.8	0.005	32.88	1.17	45.62
25	1.2	0.016	46.79	0.82	32.06
25	1.1	0.013	43.59	0.88	34.41
25	1	0.01	40.2	0.96	37.32
25	0.9	0.007	36.62	1.05	40.96
25	0.8	0.005	32.88	1.17	45.62
25	0.7	0.003	29.01	1.33	51.7
25	0.6	0.002	25.03	1.54	59.92
25	0.5	0.001	20.97	1.84	71.54
30	1.2	0.014	39.45	0.98	38.02
30	1.1	0.011	36.69	1.05	40.88
30	1	0.008	33.77	1.14	44.41
30	0.9	0.006	30.72	1.25	48.83
30	0.8	0.004	27.55	1.4	54.45
30	0.7	0.003	24.27	1.59	61.79
30	0.6	0.002	20.92	1.84	71.7
30	0.5	0.001	17.51	2.2	85.68
35	1.2	0.012	34.05	1.13	44.05
35	1.1	0.009	31.63	1.22	47.43
35	1	0.007	29.08	1.32	51.58
35	0.9	0.005	26.43	1.46	56.76
35	0.8	0.004	23.68	1.63	63.35
35	0.7	0.002	20.85	1.85	71.95

E. Python code Maximum stress regression

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
from sklearn.metrics import r2_score
from aquarel import load_theme
import matplotlib.lines as mlines

#Materials E/sigmay
nylon    = [28]
pla      = [60]
placf    = [32.5]
nitinola  = [116.7]
Abs      = [39.0]
titanium  = [137.6]
PMMA     = [55]
nitinolm  = [333.3]
spring   = [102.7]

levels = [nylon, pla, placf, nitinola, Abs, titanium, PMMA, nitinolm, spring]
colors = ['green','red','aquamarine','orange','violet','olive','pink','blue','chocolate']
labels = ['Nylon','PLA','PLA-CF','Nitinol (austenite)','ABS','Titanium','PMMA','Nitinol (martensite)','Spring Steel']

#Sort
combined = list(zip(levels, colors, labels))
combined_sorted = sorted(combined, key=lambda x: x[0])  # sort by x[0] -> numeric level
levels_sorted, colors_sorted, labels_sorted = zip(*combined_sorted)

levels = list(levels_sorted)
colors = list(colors_sorted)
labels = list(labels_sorted)

#Load dataset
df_full = pd.read_csv("/Users/yaromarsille/Documents/Vormfactoren.csv")
df_full["R"]        = df_full["R"].astype(float)
df_full["t1"]       = df_full["t1"].astype(float)
df_full["Youngs/Max"] = df_full["Youngs/Max"].astype(float)

#Check for zero/negative
df_log = df_full[
   (df_full["R"]       > 0) &
   (df_full["t1"]      > 0) &
   (df_full["Youngs/Max"] > 0)
].copy()

#Power law fit
df_log["ln_R"]  = np.log(df_log["R"])
df_log["ln_t1"] = np.log(df_log["t1"])
df_log["ln_Y"]  = np.log(df_log["Youngs/Max"])

X_data = sm.add_constant(df_log[["ln_R","ln_t1"]])  # for intercept
y_data = df_log["ln_Y"]
model_log = sm.OLS(y_data, X_data).fit()

c0, c1, c2 = model_log.params
A    = np.exp(c0)  # e^(c0) => A
alpha = c1
beta  = c2

print("\nPower Law Fit")
print(model_log.summary())
print(f"ln(A)={c0:.4f}, alpha={alpha:.4f}, beta={beta:.4f}")
print(f"=> (E/σ_max) = {A:.4f} * R^{alpha:.4f} * t1^{beta:.4f}")

#Evaluate R^2 in linear space
ln_yhat = model_log.predict(X_data)
yhat    = np.exp(ln_yhat)
r2_power = r2_score(df_log["Youngs/Max"], yhat)
print(f"Power-law R^2 in linear space: {r2_power:.4f}")

#Function
def power_law(R, t1, A, alpha, beta):
   """
   Return E/σ_max = A * R^alpha * t1^beta
   """
   return A * (R**alpha) * (t1**beta)
#3D plotting grids
```

```
R_range_full  = np.linspace(df_full["R"].min(),  df_full["R"].max(), 50)
t1_range_full = np.linspace(df_full["t1"].min(), df_full["t1"].max(), 50)
R_mesh_full, t1_mesh_full = np.meshgrid(R_range_full, t1_range_full)
Y_surf_full = power_law(R_mesh_full, t1_mesh_full, A, alpha, beta)

#R between 20 and 30
df_filtered = df_full[(df_full["R"]>=20) & (df_full["R"]<=30)]
R_range_filt  = np.linspace(20,30,50)
t1_range_filt = np.linspace(df_filtered["t1"].min(), df_filtered["t1"].max(), 50)
R_mesh_filt, t1_mesh_filt = np.meshgrid(R_range_filt, t1_range_filt)
Y_surf_filt = power_law(R_mesh_filt, t1_mesh_filt, A, alpha, beta)

#Appearance
theme = (load_theme("scientific")
      .set_grid(draw=False, width=0.5)
      .set_font(family="medium")
      .set_overrides({"ytick.minor.visible": False,
                  "xtick.minor.visible": False}))
theme.apply()

#Fig1: Full dataset
fig1 = plt.figure(figsize=(10,10))
ax1  = fig1.add_subplot(111, projection='3d')
ax1.set_title("E/$σ_{max}$, including Power Law fit",fontsize = 14 , fontweight='medium', x = 0.6, y=0.96)

for mat in df_full["Ref."].unique():
   sub = df_full[df_full["Ref."] == mat]
   ax1.scatter(sub["R"], sub["t1"], sub["Youngs/Max"],
           label=mat, s=80, edgecolors='black', alpha=0.8, depthshade=True)

ax1.plot_surface(R_mesh_full, t1_mesh_full, Y_surf_full,
            color='orange', alpha=0.4, edgecolor='none')
ax1.set_xlabel("R [mm]", fontsize=14, labelpad=5)
ax1.set_ylabel("t1 [mm]", fontsize=14, labelpad=5)
ax1.set_zlabel("E/$σ_{max}$", fontsize=14, labelpad=5)
ax1.set_xlim(0,70)
ax1.set_ylim(0,1.4)
ax1.set_zlim(0,250)
ax1.view_init(elev=20, azim=60)
ax1.legend(title="Material", loc="upper left", bbox_to_anchor=(1,0.7), fontsize=12)

#Fig2: Filtered data set
fig2 = plt.figure(figsize=(12,6))
ax2  = fig2.add_subplot(111, projection='3d')
ax2.set_title("E/$σ_{max}$ for 20 < R < 30", fontsize=16)

for mat in df_filtered["Ref."].unique():
   sub = df_filtered[df_filtered["Ref."] == mat]
   ax2.scatter(sub["R"], sub["t1"], sub["Youngs/Max"],
           label=mat, s=80, edgecolors='black', alpha=0.8, depthshade=False)

ax2.plot_surface(R_mesh_filt, t1_mesh_filt, Y_surf_filt,
            color='orange', alpha=0.4, edgecolor='none')
ax2.set_xlabel("R [mm]", fontsize=14, labelpad=10)
ax2.set_ylabel("t1 [mm]", fontsize=14)
ax2.set_zlabel("E/$σ_{max}$", fontsize=14)
ax2.set_xlim(20,30)
ax2.set_ylim(0,1.2)
ax2.set_zlim(0,100)
ax2.view_init(elev=30, azim=140)
ax2.legend(title="Material", loc="upper left", bbox_to_anchor=(1,1), fontsize=12)

#Fig3: projection at t1 = 1.2 mm
fig3, ax3 = plt.subplots(figsize=(10,6))
df_proj_12 = df_full[df_full["t1"]==1.2]
for mat in df_proj_12["Ref."].unique():
   sub = df_proj_12[df_proj_12["Ref."] == mat]
   ax3.scatter(sub["R"], sub["Youngs/Max"], label=mat, s=80,
           edgecolors='black', alpha=0.8)

R_line = np.linspace(1,70,200)
Y_line = power_law(R_line, np.full_like(R_line,1.2), A, alpha, beta)
ax3.plot(R_line, Y_line, color='orange', alpha=0.7, label="$E/$σ_{max}$")
ax3.set_xlim(0,70)
ax3.set_ylim(0,200)
ax3.set_xlabel("R [mm]", fontsize=14)
ax3.set_ylabel("E/$σ_{max}$", fontsize=14)
ax3.set_title("E/$σ_{max}$ at t1 = 1.2 mm", fontsize=16)
ax3.legend(loc="upper left", bbox_to_anchor=(1,1), fontsize=12)
ax3.grid(True, linestyle='--', linewidth=0.5)
```

```
#Fig4: projection at t1 = 0.6 mm
fig4, ax4 = plt.subplots(figsize=(10,6))

df_proj_08 = df_full[df_full["t1"]==0.6]
for mat in df_proj_08["Ref."].unique():
   sub = df_proj_08[df_proj_08["Ref."]==mat]
   ax4.scatter(sub["R"], sub["Youngs/Max"], label=mat, s=80,
           edgecolors='black', alpha=0.8)

Y_line1 = power_law(R_line, np.full_like(R_line,0.6), A, alpha, beta)
ax4.plot(R_line, Y_line1, color='orange', alpha=0.7, label="$E/σ_{max}$")

ax4.set_xlim(0,70)
ax4.set_ylim(0,200)
ax4.set_xlabel("R [mm]", fontsize=12)
ax4.set_ylabel("$E/σ_{max}$", fontsize=12)
ax4.set_title("$E/σ_{max}$ at t1 = 0.6 mm", fontsize=14)
ax4.legend(loc="upper left", bbox_to_anchor=(1,1), fontsize=12)
ax4.grid(True, linestyle='--', linewidth=0.5)

#Fig5: projection at t1 = 1.2 mm
fig5, ax5 = plt.subplots(figsize=(10,6))
df_proj_R20 = df_full[df_full["R"]==20]
for mat in df_proj_R20["Ref."].unique():
   sub = df_proj_R20[df_proj_R20["Ref."]==mat]
   ax5.scatter(sub["t1"], sub["Youngs/Max"], label=mat, s=80,
           edgecolors='black', alpha=0.8)

t_line = np.linspace(0.1,1.4,200)
Y_line = power_law(np.full_like(t_line,20), t_line, A, alpha, beta)
ax5.plot(t_line, Y_line, color='red', alpha=0.7, label="$E/σ_{max}$")

ax5.set_xlim(0,1.4)
ax5.set_ylim(0,200)
ax5.set_xlabel("t1 [mm]", fontsize=12)
ax5.set_ylabel("$E/σ_{max}$", fontsize=12)
ax5.set_title("$E/σ_{max}$ at R = 20 mm", fontsize=14)
ax5.legend(loc="upper left", bbox_to_anchor=(1,1), fontsize=12)
ax5.grid(True, linestyle='--', linewidth=0.5)

#Fig6: projection at R = 25 mm
fig6, ax6 = plt.subplots(figsize=(10,6))

df_proj_R25 = df_full[df_full["R"]==25]
for mat in df_proj_R25["Ref."].unique():
   sub = df_proj_R25[df_proj_R25["Ref."]==mat]
   ax6.scatter(sub["t1"], sub["Youngs/Max"], label=mat, s=80,
           edgecolors='black', alpha=0.8)

Y_line = power_law(np.full_like(t_line,25), t_line, A, alpha, beta)
ax6.plot(t_line, Y_line, color='red', alpha=0.7, label="$E/σ_{max}$")

ax6.set_xlim(0,1.4)
ax6.set_ylim(0,200)
ax6.set_xlabel("t1 [mm]", fontsize=12)
ax6.set_ylabel("$E/σ_{max}$", fontsize=12)
ax6.set_title("$E/σ_{max}$ at R = 25 mm", fontsize=14)
ax6.legend(loc="upper left", bbox_to_anchor=(1,1), fontsize=12)
ax6.grid(True, linestyle='--', linewidth=0.5)

#Fig7: contour plot full
R_range_projection  = np.linspace(0.1,70,50)
t1_range_projection = np.linspace(0.1,1.4,50)
R_mesh_proj, t1_mesh_proj = np.meshgrid(R_range_projection, t1_range_projection)
Z_mesh_proj = power_law(R_mesh_proj, t1_mesh_proj, A, alpha, beta)

fig7, ax7 = plt.subplots(figsize=(8,6))
for lev, color in zip(levels, colors):
   CS = ax7.contour(R_mesh_proj, t1_mesh_proj, Z_mesh_proj,
             levels=[lev], colors=color, linewidths=2)
   ax7.clabel(CS, inline=True, fmt='%.1f')
ax7.plot(R_range_projection, np.full_like(R_range_projection, 0.6),
      color='black', linestyle='--', alpha=0.7, label='Minimum 3D printing wall thickness')

handles = []
for color, label in zip(colors, labels):
   line = mlines.Line2D([], [], color=color, label=label, linewidth=2)
   handles.append(line)
```

```
min_line = mlines.Line2D([], [], color='black', linestyle='--',
                  label='Minimum 3D printing wall thickness')
handles.append(min_line)

ax7.legend(handles=handles, title='$E/σ_{max}$  [-]', loc="upper left",
        bbox_to_anchor=(1,1), fontsize=12)
ax7.set_xlabel("R [mm]", fontsize=12)
ax7.set_ylabel("t1 [mm]", fontsize=12)
ax7.set_ylim(0,1.4)
ax7.set_xlim(0,70)
ax7.set_title("Minimum Geometries for 90 degrees rotation", fontsize=14)
ax7.grid(True, linestyle='--', linewidth=0.5)

#Fig8: contour plot filtered
t1_range_projection2 = np.linspace(1,1.3,50)
R_range_projection2  = np.linspace(20,50,50)
R_mesh2, t1_mesh2 = np.meshgrid(R_range_projection2, t1_range_projection2)
Z_mesh2 = power_law(R_mesh2, t1_mesh2, A, alpha, beta)

fig8, ax8 = plt.subplots(figsize=(8,6))
handles = []
for lev, color, label in zip(levels, colors, labels):
   CS = ax8.contour(R_mesh2, t1_mesh2, Z_mesh2,
              levels=[lev], colors=color, linewidths=2)
   ax8.clabel(CS, inline=True, fmt='%.1f')
   # Add legend handle only if contour is drawn
   if CS.allsegs and len(CS.allsegs[0])>0:
      line = mlines.Line2D([], [], color=color, label=label, linewidth=2)
      handles.append(line)

ax8.plot(R_range_projection, np.full_like(R_range_projection, 0.6),
       color='black', linestyle='--', alpha=0.7, label='Minimum 3D printing wall thickness')

min_line = mlines.Line2D([], [], color='black', linestyle='--',
                  label='Minimum 3D printing wall thickness')
handles.append(min_line)

ax8.legend(handles=handles, title='E/$σ_{max}$  [-]', loc="upper left",
        bbox_to_anchor=(1,1), fontsize=12)
ax8.set_xlabel("R [mm]", fontsize=12)
ax8.set_ylabel("t1 [mm]", fontsize=12)
ax8.set_ylim(1.1,1.3)
ax8.set_xlim(20,50)
ax8.set_title("Eligible Geometries", fontsize=14)
ax8.grid(True, linestyle='--', linewidth=0.5)

#Fig9: contour plot filtered
t1_range_projection1 = np.linspace(0.5,0.7,50)
R_range_projection1  = np.linspace(0.1,30,50)
R_mesh1, t1_mesh1 = np.meshgrid(R_range_projection1, t1_range_projection1)
Z_mesh1 = power_law(R_mesh1, t1_mesh1, A, alpha, beta)

fig9, ax9 = plt.subplots(figsize=(8,6))
handles = []
for lev, color, label in zip(levels, colors, labels):
   CS = ax9.contour(R_mesh1, t1_mesh1, Z_mesh1,
              levels=[lev], colors=color, linewidths=2)
   ax9.clabel(CS, inline=True, fmt='%.1f')
   if CS.allsegs and len(CS.allsegs[0])>0:
      line = mlines.Line2D([], [], color=color, label=label, linewidth=2)
      handles.append(line)
ax9.plot(R_range_projection, np.full_like(R_range_projection, 0.6),
       color='black', linestyle='--', alpha=0.7, label='Minimum 3D printing wall thickness')

min_line2 = mlines.Line2D([], [], color='black', linestyle='--',
                   label='Minimum 3D printing wall thickness')
handles.append(min_line2)

ax9.legend(handles=handles, title='E/$σ_{max}$  [-]', loc="upper left",
        bbox_to_anchor=(1,1), fontsize=12)
ax9.set_xlabel("R [mm]", fontsize=12)
ax9.set_ylabel("t1 [mm]", fontsize=12)
ax9.set_ylim(0.5,0.7)
ax9.set_xlim(0,30)
ax9.set_title("Minimum Geometries for R < 30", fontsize=14)
ax9.grid(True, linestyle='--', linewidth=0.5)

plt.show()
theme.apply_transforms()
```

F. Python code Stiffness regression

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score

#Data
file_path = "/Users/yaromarsille/Documents/Stijfheid Regressie.csv"
df = pd.read_csv(file_path)

#Define different w values
w_values = [1,10,20]
regression_results = {}

for w_val in w_values:
  #Scale stiffness
  df[f"Stiffness_w{w_val}"] = df["Stiffness"] * w_val

  #Log-transform variables for power-law regression
  df_log = np.log(df[["R [mm]", "t1 [mm]", "E [MPa]", f"Stiffness_w{w_val}"]])

  #Independent and dependent variables
  X_log = df_log[["R [mm]", "t1 [mm]", "E [MPa]"]].values
  y_log = df_log[f"Stiffness_w{w_val}"].values

  #Fit linear regression in log-space (equivalent to power-law fitting)
  model = LinearRegression()
  model.fit(X_log, y_log)

  #Compute predictions and R^2 score
  y_log_pred = model.predict(X_log)
  r2 = r2_score(y_log, y_log_pred)

  #Extract regression parameters
  params = np.append(np.exp(model.intercept_), model.coef_)

  #Store results
  regression_results[w_val] = {
    "Equation": f"Stiffness = {params[0]:.6f} * R^{params[1]:.3f} * t1^{params[2]:.3f} * E^{params[3]:.3f}",
    "R^2": r2}

print(regression_results[1])

#Material Youngs modulus
levels = {"Nylon": [1.4], "PLA": [3], "PLA-CF": [3.7], "Nitinol (austenite)": [70], "ABS": [1.5], "Titanium": [114], "PMMA": [3.3], "Nitinol (martensite)": [40],
"Spring Steel": [190]}
colors = ['green', 'red', 'aquamarine', 'orange', 'violet', 'olive', 'pink', 'khaki', 'chocolate']

target = 0.5
Cons = target/(params[0]/w_values[-1])

#Generate 3D surface plots and 2D contour plots for each w value
for w_val in w_values:
  fig1 = plt.figure(figsize=(10, 10))
  ax1 = fig1.add_subplot(111, projection='3d')
  R_vals = np.linspace(df["R [mm]"].min(), df["R [mm]"].max(), 30)
  R_vals1 = np.linspace(1e-6,70, 50)
  t1_vals = np.linspace(df["t1 [mm]"].min(), df["t1 [mm]"].max(), 30)
  t1_vals1 = np.linspace(0.4,1.4, 50)
  R_mesh, t1_mesh = np.meshgrid(R_vals1, t1_vals1)
  equation = (Cons/1000)*((R_mesh**-params[1])/(w_val*t1_mesh**params[2])) #to GPa
  surf = ax1.plot_surface(R_mesh, t1_mesh, equation, alpha=0.7, linewidth=0.5)
  ax1.set_xlabel("R [mm]", fontsize=10, labelpad=5)
  ax1.set_ylabel("t1 [mm]", fontsize=10, labelpad=5)
  ax1.set_zlabel("E [GPa]", fontsize=10, labelpad=0)
  ax1.zaxis.set_tick_params(labelsize=9)
  ax1.ticklabel_format(style='plain')
  ax1.set_title(f"Stiffness fit for 1 Nm/rad and $w$ = {w_val} mm", fontsize=14, y = 0.96)
  ax1.view_init(elev=20, azim=60)

  #Generate 2D contour plot
  fig, ax2 = plt.subplots(figsize=(10, 6))

  for label, level in levels.items():
    material_contours = ax2.contour(R_mesh, t1_mesh, equation, levels=level, colors=[colors[list(levels.keys()).index(label)]], linewidths=2.0)
    ax2.clabel(material_contours, fmt={level[0]: label}, inline=True, fontsize=10)
```

```
    ax2.set_xlabel("R [mm]", fontsize=12)
    ax2.set_ylabel("t1 [mm]", fontsize=12)
    ax2.set_xlim(0,70)
    ax2.set_ylim(0,1.4)
    ax2.set_title(f"Stiffness = 0.5 Nm/rad, w = {w_val}", fontsize=14)
    ax2.grid(True, linestyle='--', linewidth=0.5)

    #Generate 2D contour plot
    fig, ax3 = plt.subplots(figsize=(10, 6))

    for label, level in levels.items():
        material_contours = ax3.contour(R_mesh, t1_mesh, equation, levels=level, colors=[colors[list(levels.keys()).index(label)]], linewidths=2.0)
        ax3.clabel(material_contours, fmt={level[0]: label}, inline=True, fontsize=10)

    ax3.set_xlabel("R [mm]", fontsize=12)
    ax3.set_ylabel("t1 [mm]", fontsize=12)
    ax3.set_xlim(20,30)
    ax3.set_ylim(1,1.4)
    ax3.set_title(f"Stiffness = 0.5 Nm/rad, w = {w_val}", fontsize=14)
    ax3.grid(True, linestyle='--', linewidth=0.5)

plt.show()
```

G. Data analysis

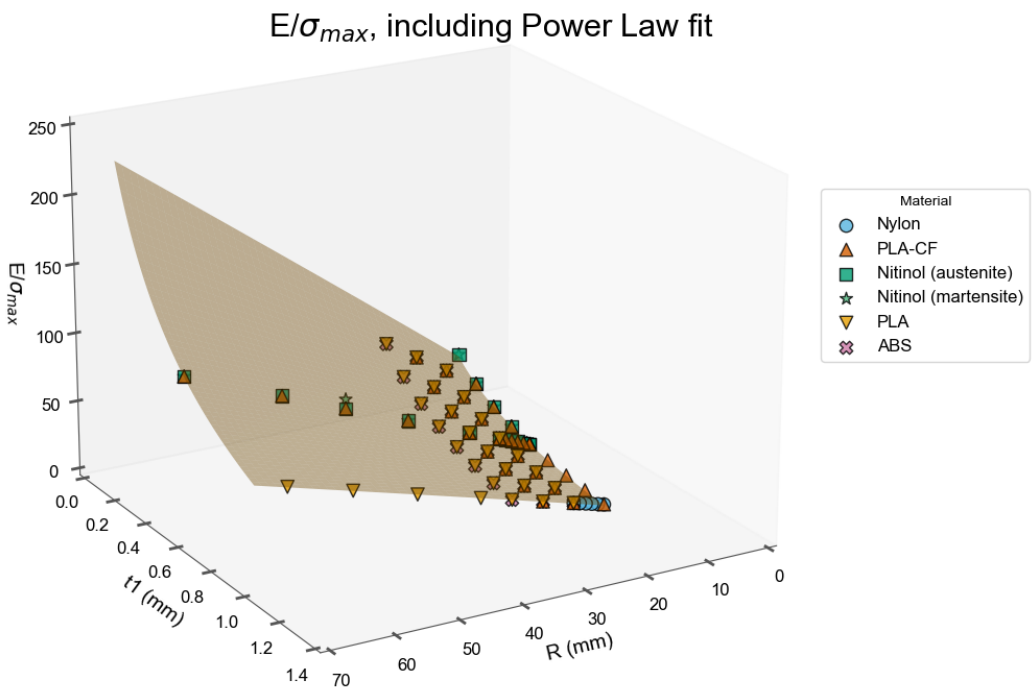

Figure 8: 3D plot of the Young's modulus/maximum stress ratio vs. t1 and R for various materials, including a power law fit plane

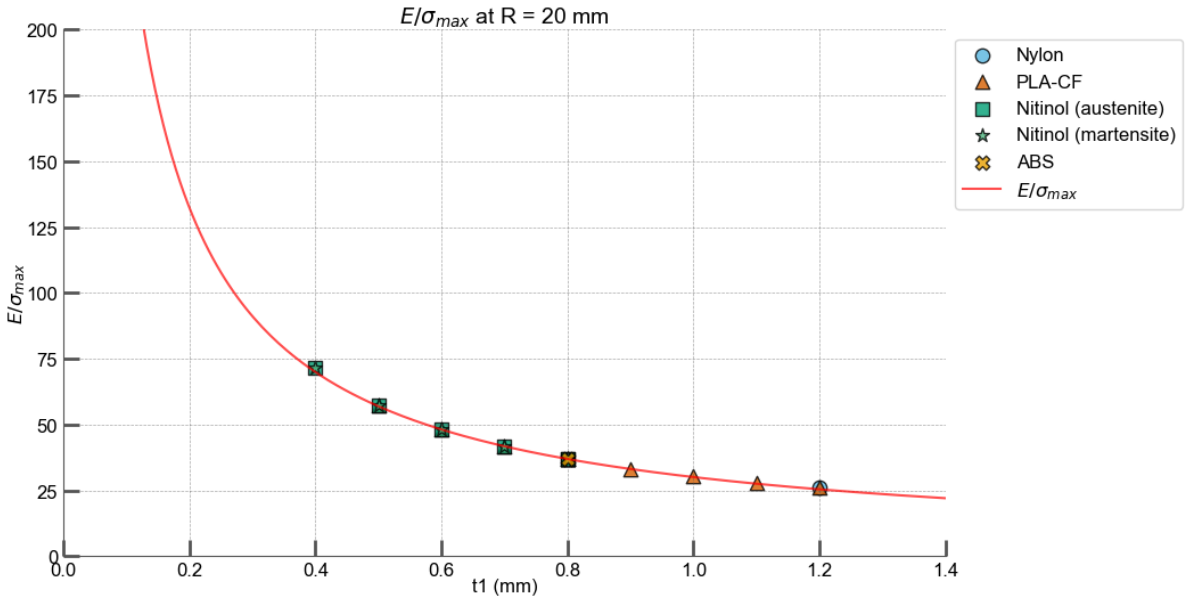

Figure 9: Graph of the Young's modulus/maximum stress ratio for t1 at R = 20, including the regressing line

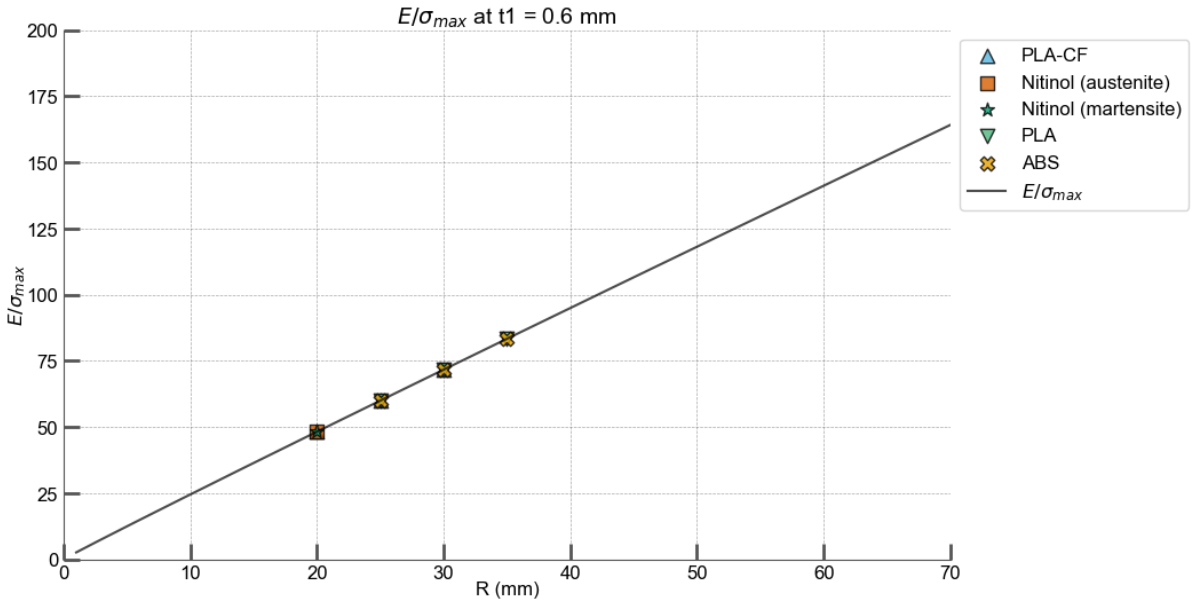

Figure 10: Graph of the Young's modulus/maximum stress ratio for R at t1 = 0.6 mm, including the regression lines.

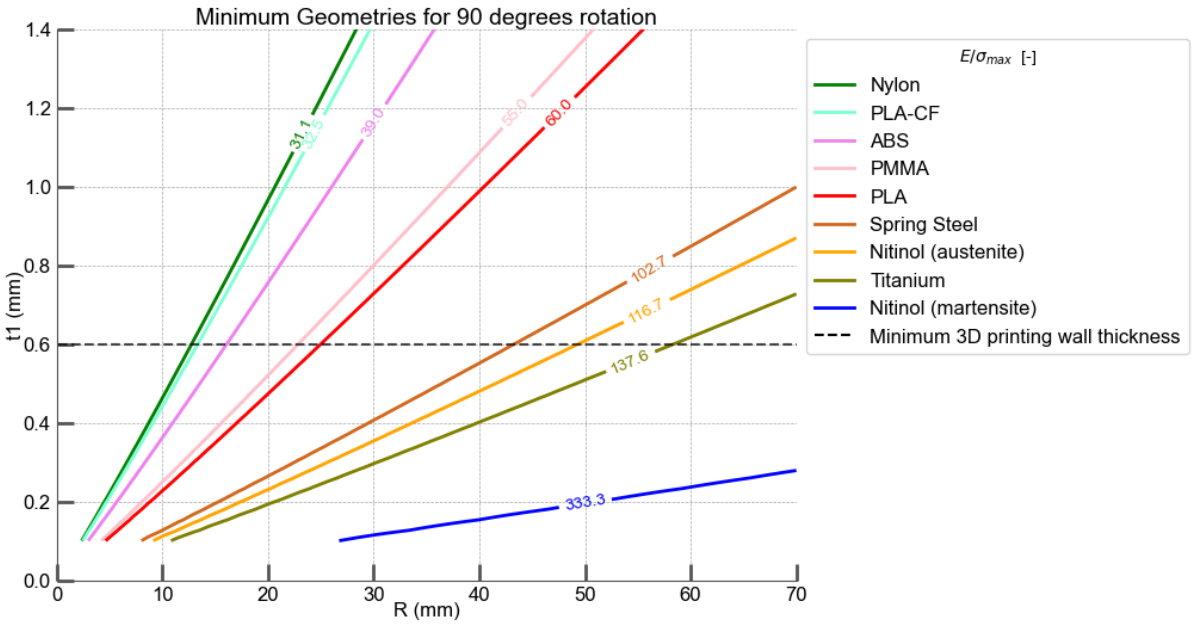

Figure 11: Graph of the Young's modulus/Yield stress ratio for R and t1 for various materials. The geometry should be chosen on the right side of the line per material to prevent plastic deformation. The number depicted in each line is the ratio E/max for every material.

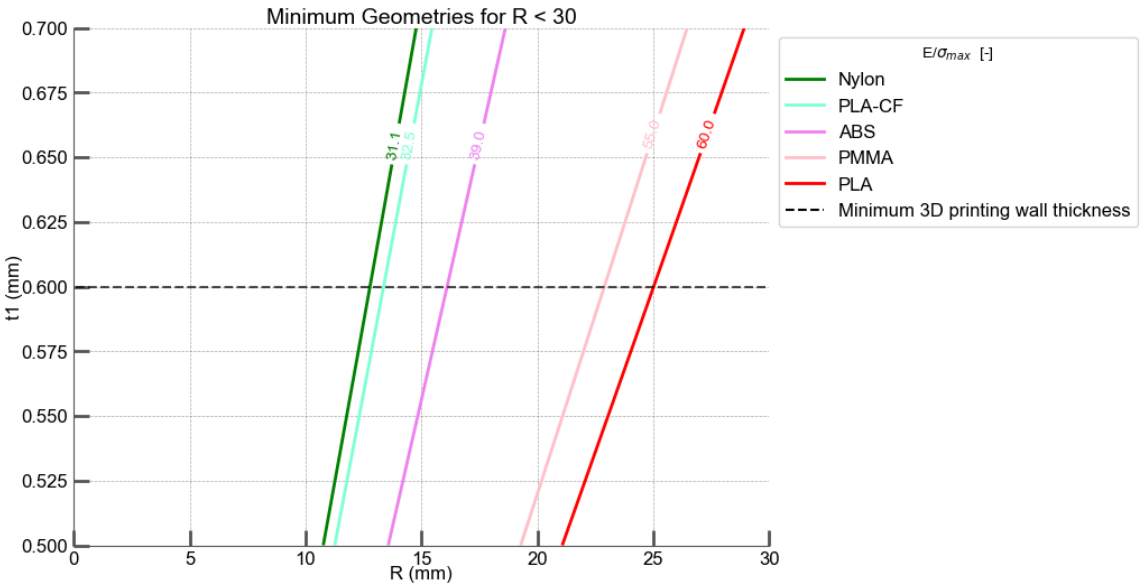

Figure 12: Graph of the Young's modulus/Yield stress ratio for R < 30 mm, at t1 at the minimum 3D printing size for various materials. The geometry should be chosen right of the line per material to prevent failure. The number depicted in each line is the ratio E/max for every material.

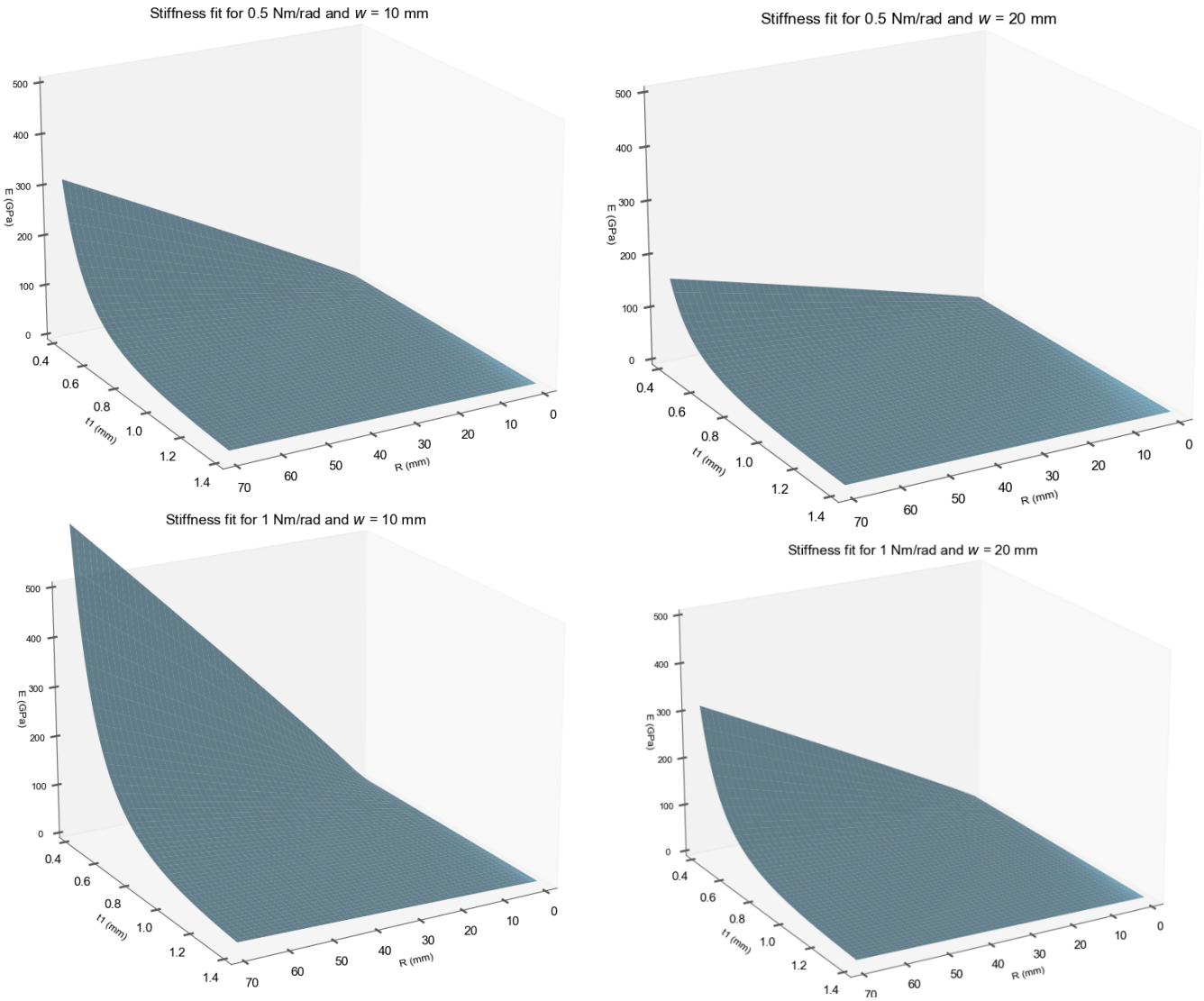

Figure 13: Stiffness regression plane for t1, R and E for target stiffness 0.5 and 1 Nm/rad, for flexure widths w = 10 and 20 mm.

H. Investigated materials before manufacturing method

Table 9: Different materials examined on their Young's modulus to Yield stress ratio

Material	Young's Modulus (GPa)	Yield Stress (MPa)	Poisson's ratio (-)	$\frac{E}{\sigma_y}$ (-)
GRFP [36]	36	550	0.28	65.5
Titanium [23]	113.8	827	0.34	137.6
Aluminium [37]	69	200	0.3	345
Stainless Steel 304 [38]	200	215	0.29	930.2
Nitinol (austenite) [34]	70	600	0.33	116.7
Nitinol (martensite) [34]	40	120	0.33	333.3
Spring Steel [33]	190	1850	0.29	102.7
Nylon [35]	1.4	45	0.39	31.1
PLA [27]	3	50	0.33	60.0
ABS [27]	1.5	38,5	0.3	39.0
PLA-CF [27]	3.7	96	0.4	38.5
PP [23]	1.38	31	0.3	44.5
PMMA [39]	3.3	60	0.35	55.0
PETG [27]	2.05	50	0.35	41.0
TPU [27]	2.58	64	0.45	40.3

I. Investigated materials after manufacturing method

Table 10: Different fiber reinforced polymers manufactured by Bambu Labs [27] examined on their Young's modulus to Yield stress ratio

Material	Young's Modulus (GPa)	Yield Stress (MPa)	Poisson's ratio (-)	$\frac{E}{\sigma_y}$ (-)
PLA-CF	3.7	96	Not provided by manufacturer	38.5
PETG-CF	2.89	83		34.8
PET-CF	5.32	131		40.6
PAHT-CF	4.12	140		29.4
PA6-CF	5.46	151		36.2
ASA-CF	3.74	72		51.9
PPS-CF	7.16	142		50.4
PA6-GF	3.67	120		30.6

J. Material and geometry selection charts

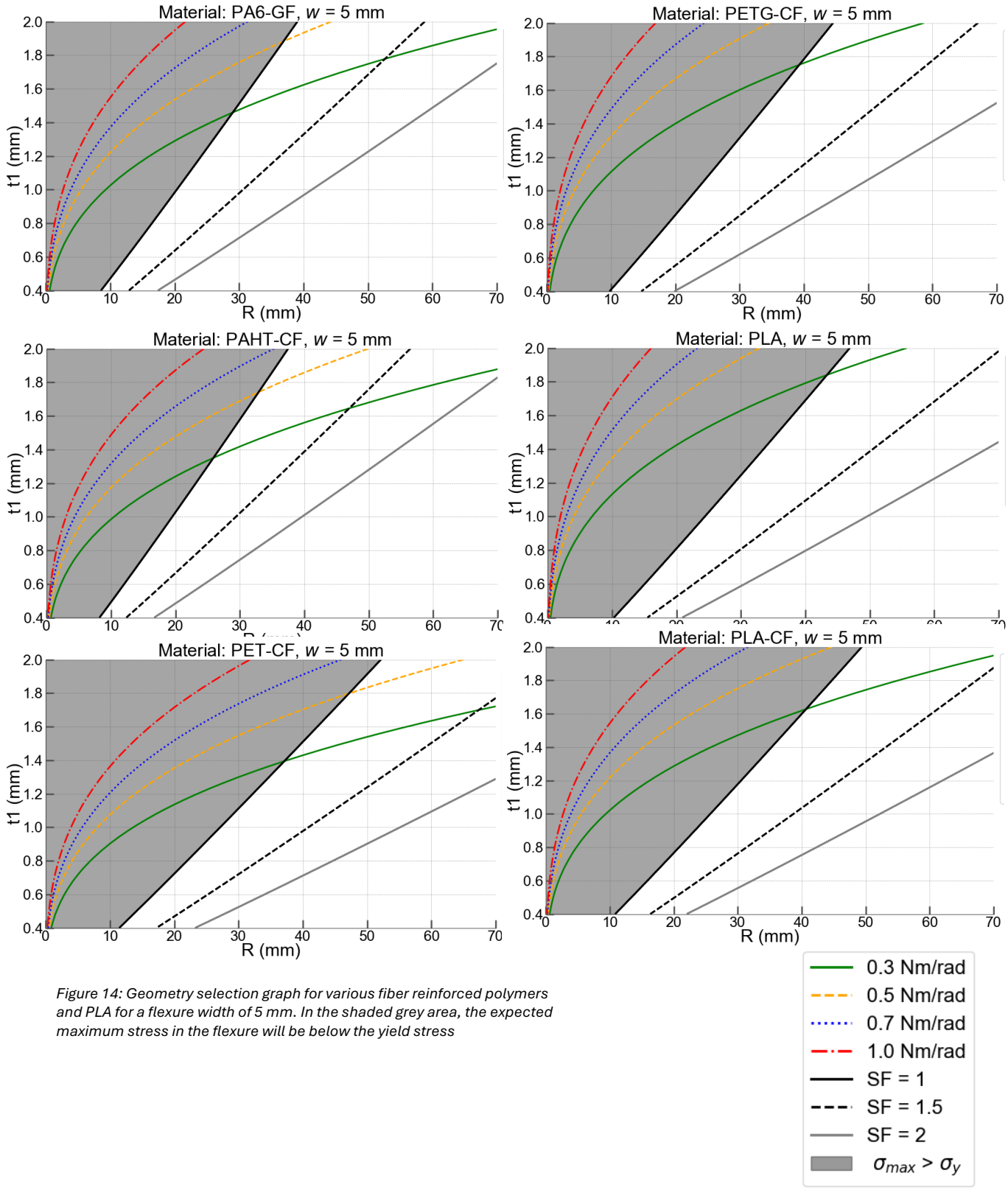

Figure 14: Geometry selection graph for various fiber reinforced polymers and PLA for a flexure width of 5 mm. In the shaded grey area, the expected maximum stress in the flexure will be below the yield stress

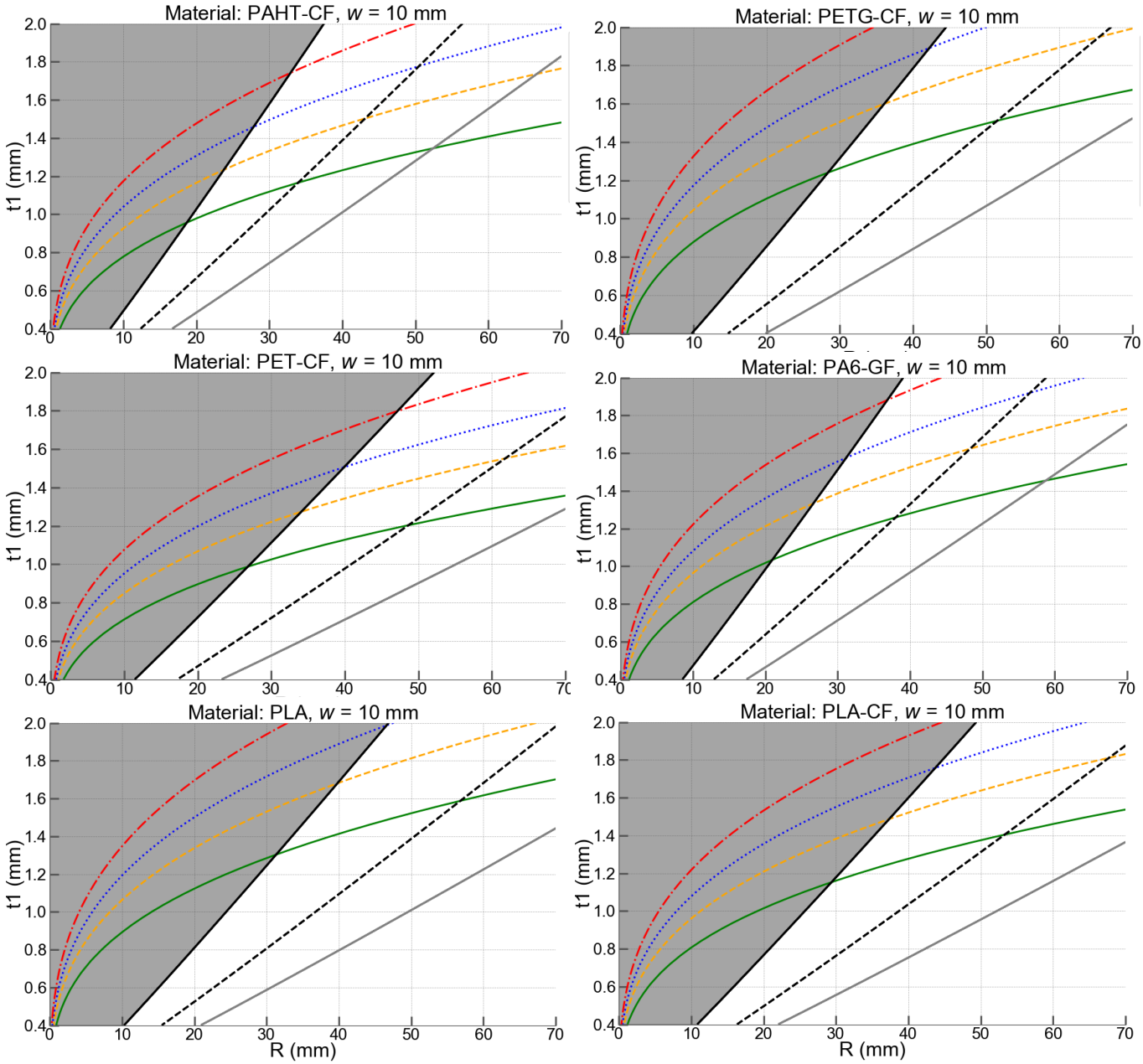

Figure 15: Geometry selection graph for various fiber reinforced polymers and PLA for a flexure width of 10 mm. In the shaded grey area, the expected maximum stress in the flexure will be above the yield stress

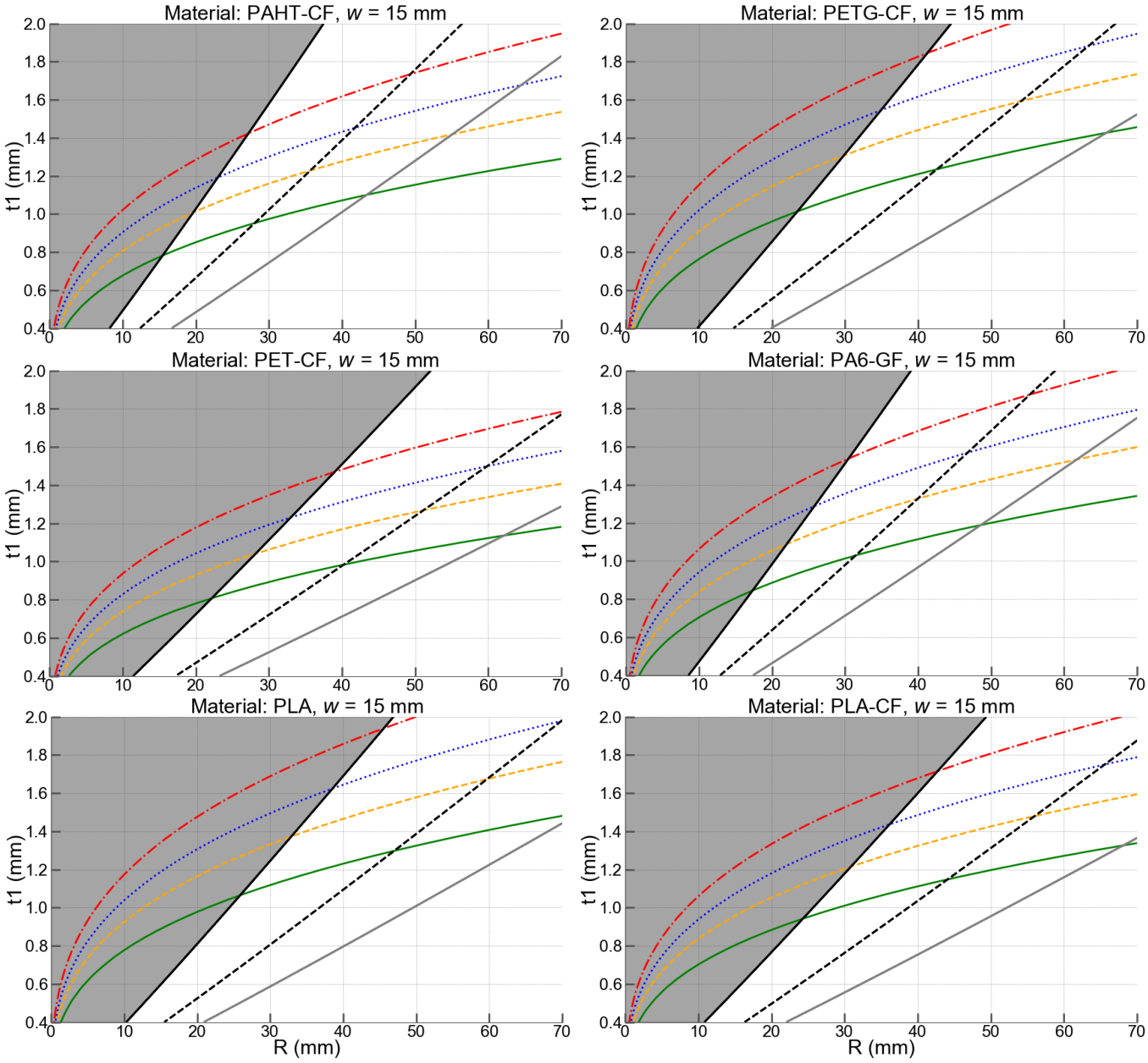

Figure 16: Geometry selection graph for various fiber reinforced polymers and PLA for a flexure width of 15 mm. In the shaded grey area, the expected maximum stress in the flexure will be above the yield stress

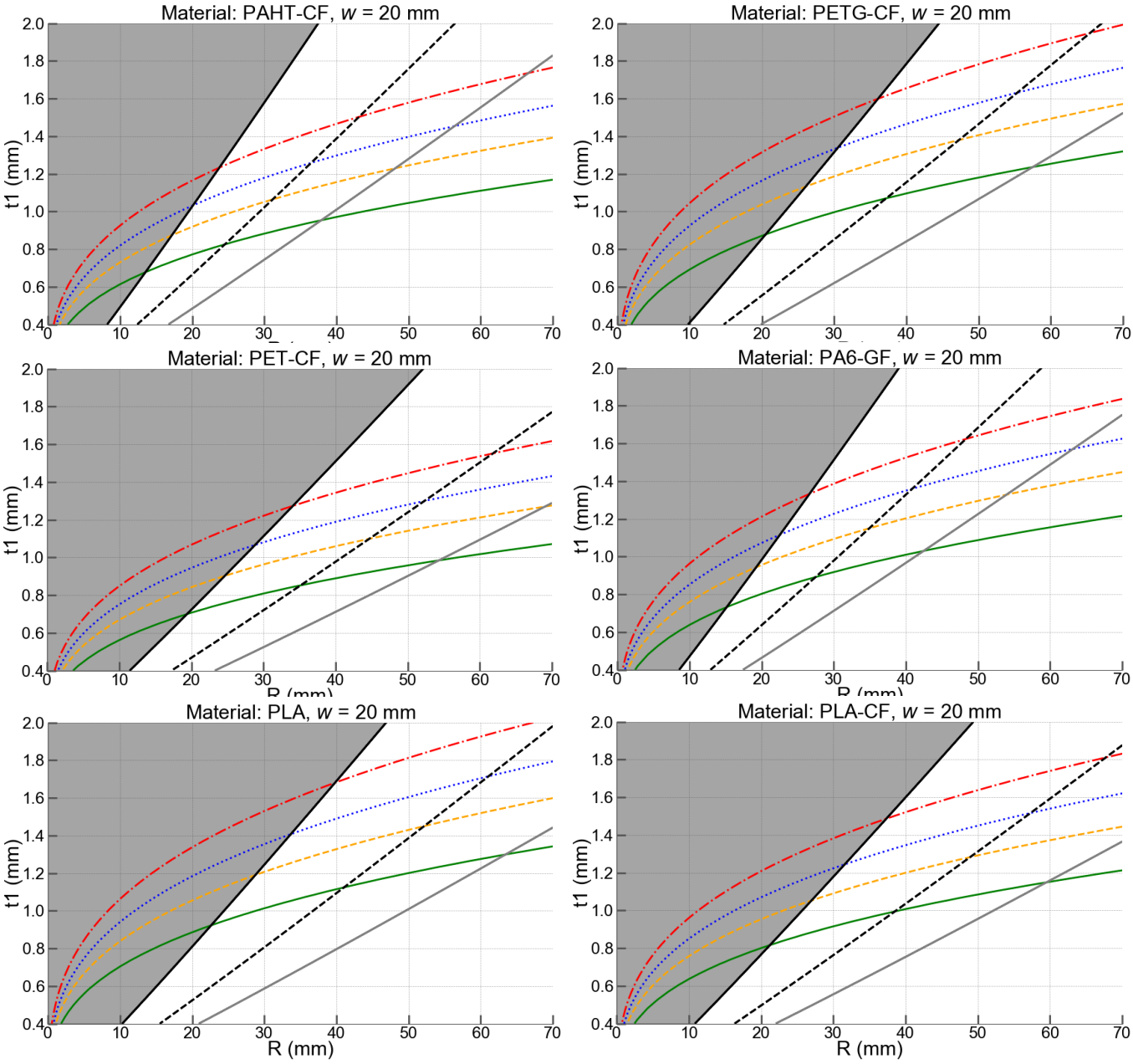

Figure 17: Geometry selection graph for various fiber reinforced polymers and PLA for a flexure width of 20 mm. In the shaded grey area, the expected maximum stress in the flexure will be above the yield stress

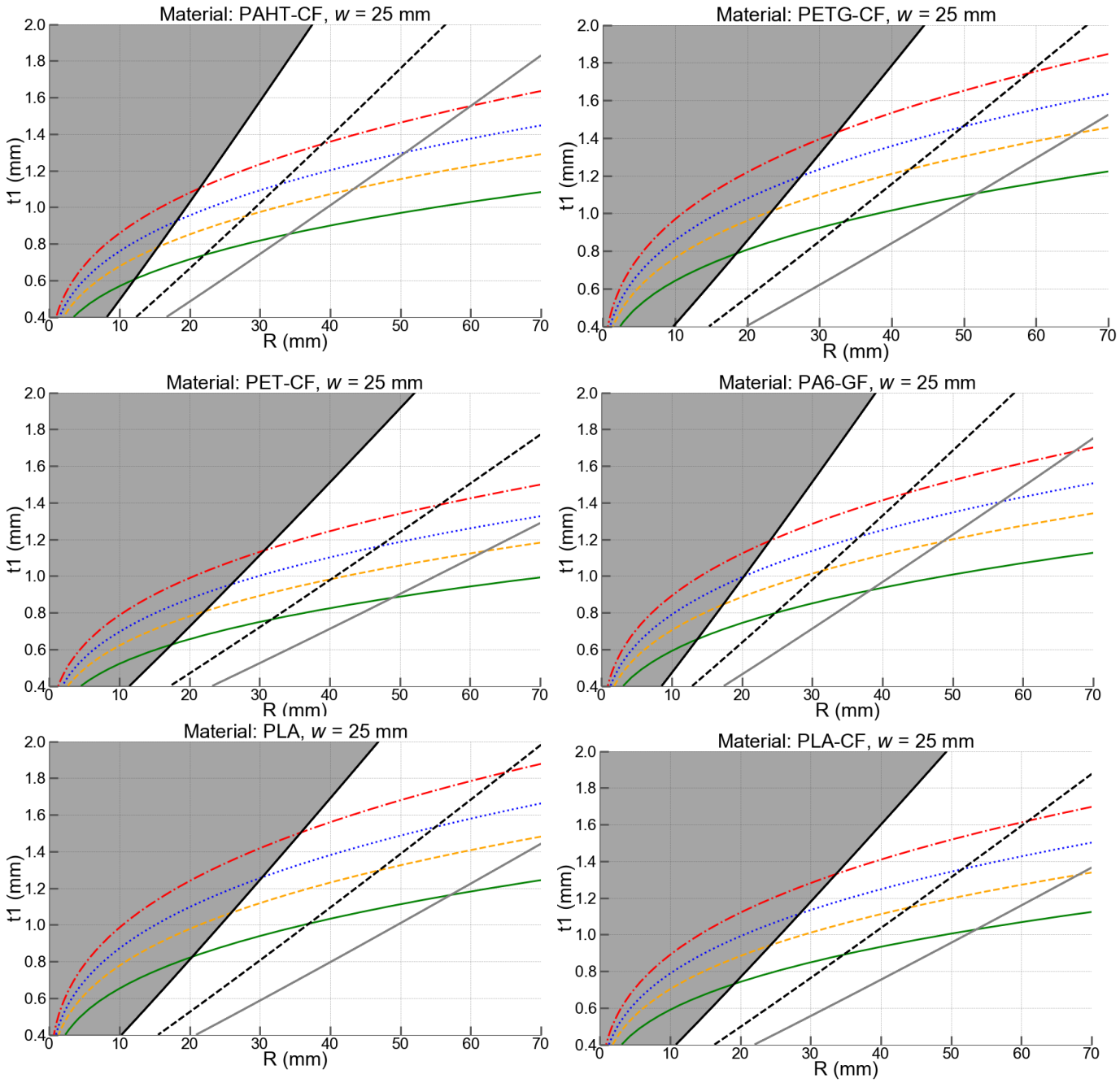

Figure 18: Geometry selection graph for various fiber reinforced polymers and PLA for a flexure width of 25 mm. In the shaded grey area, the expected maximum stress in the flexure will be above the yield stress

K. ABAQUS results prototypes

PLA Prototypes; all prototype dimensions and extracted stiffnesses can be found in Table 2 in the main report

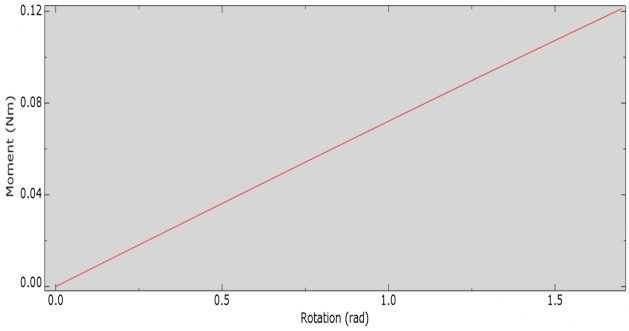

Figure 19: Expected moment rotation curve for prototype PLA1 in FE. The stiffness can be calculated from the slope of the line

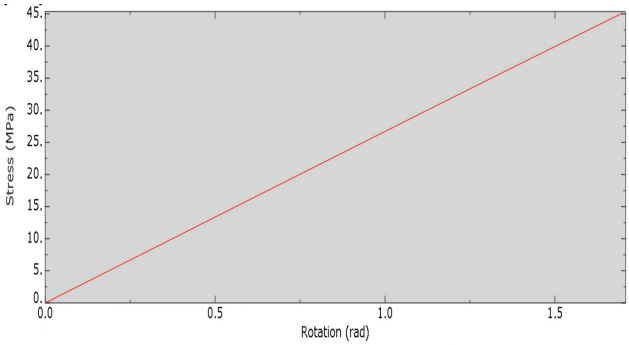

Figure 20: Expected stress-rotation curve for prototype PLA1 in FE.

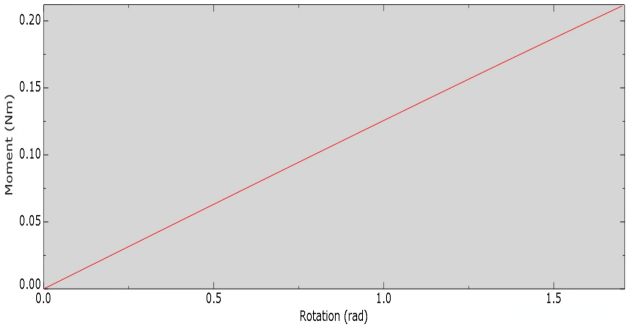

Figure 21: Expected moment rotation curve for prototype PLA2 in FE. The stiffness can be calculated from the slope of the line

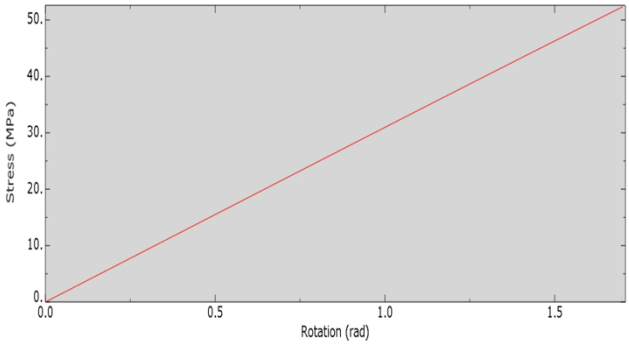

Figure 22: Expected stress-rotation curve for prototype PLA2 in FE.

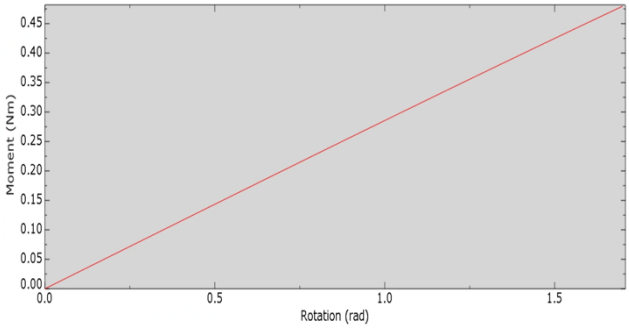

Figure 23: Expected moment rotation curve for prototype PLA3 in FE. The stiffness can be calculated from the slope of the line

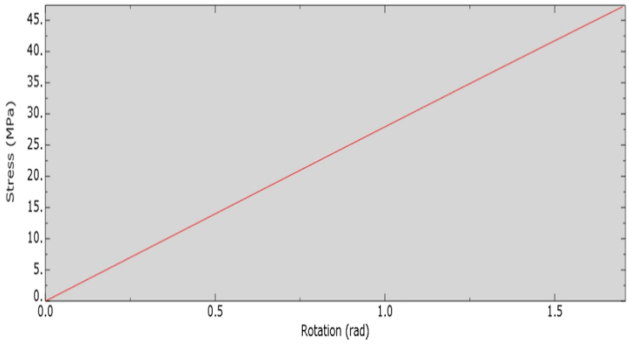

Figure 24: Expected stress-rotation curve for prototype PLA3 in FE.

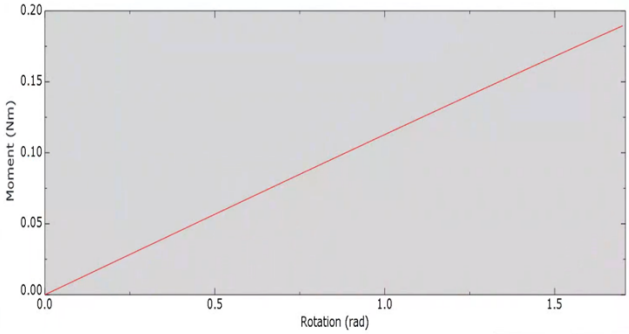

Figure 25: Expected moment-rotation curve for prototype PLA4 in FE. The stiffness can be calculated from the slope of the line

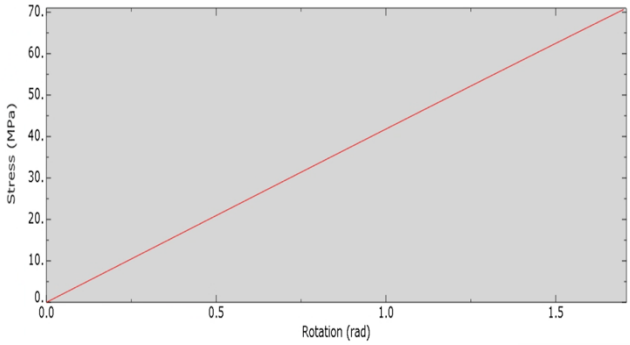

Figure 26: Expected stress-rotation curve for prototype PLA4 in FE.

PETG-CF Prototypes; all prototype dimensions and extracted stiffnesses can be found in Table 2 in the main report

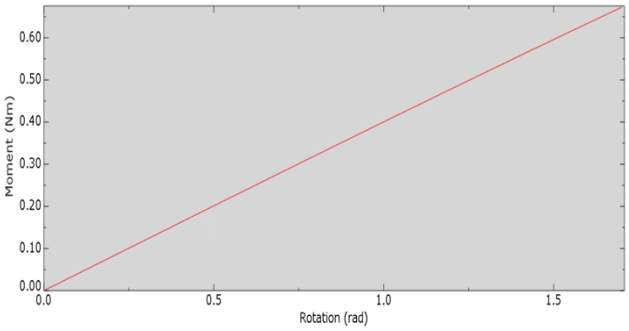

Figure 27: Expected moment-rotation curve for prototype PETG-CF1 in FE. The stiffness can be calculated from the slope of the line

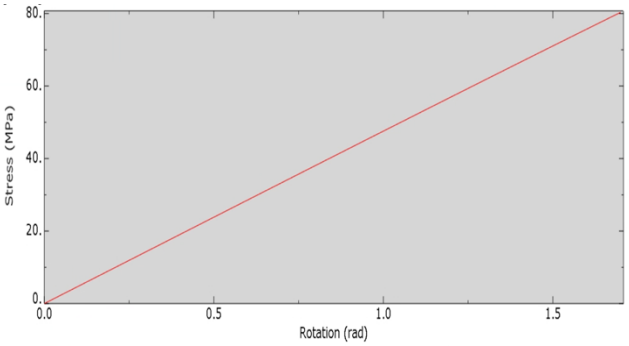

Figure 28: Expected stress-rotation curve for prototype PETG-CF1 in FE.

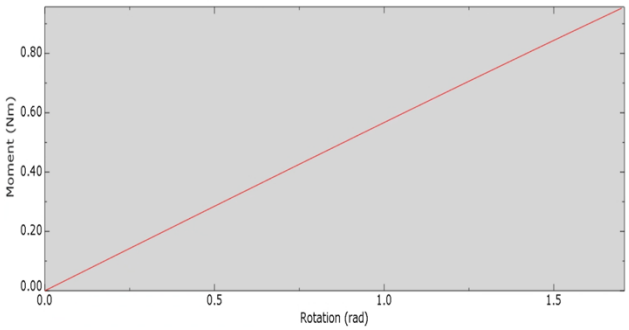

Figure 29: Expected moment-rotation curve for prototype PETG-CF2 in FE. The stiffness can be calculated from the slope of the line

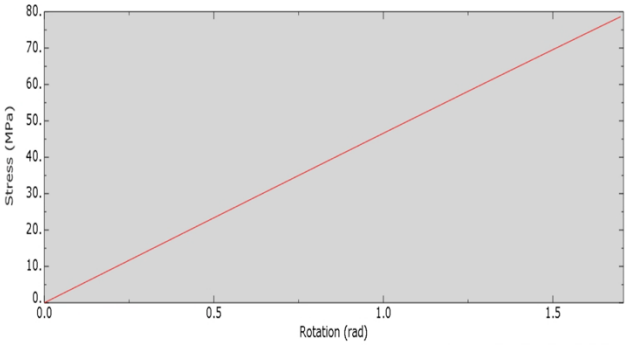

Figure 30: Expected stress-rotation curve for prototype PETG-CF2 in FE.

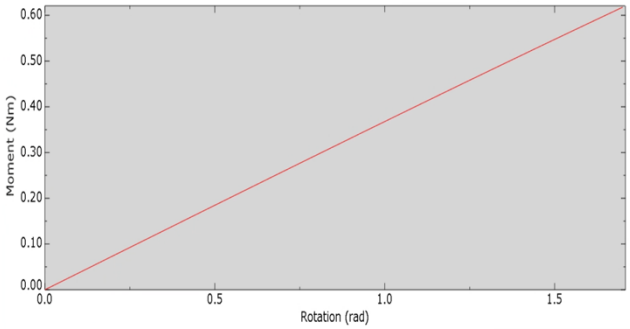

Figure 31: Expected moment-rotation curve for prototype PETG-CF3 in FE. The stiffness can be calculated from the slope of the line

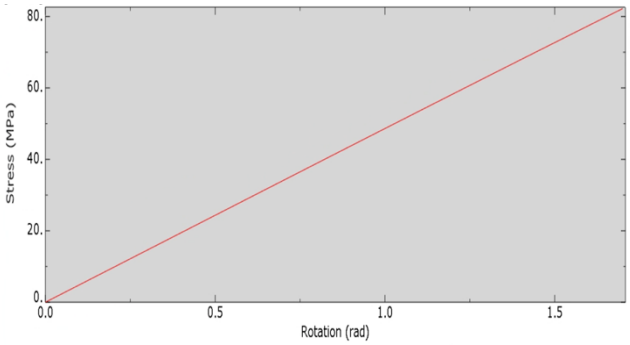

Figure 32: Expected stress-rotation curve for prototype PETG-CF3 in FE.

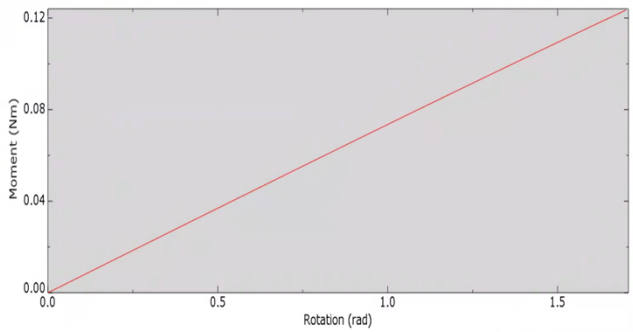

Figure 33: Expected moment-rotation curve for prototype PETG-CF4 in FE. The stiffness can be calculated from the slope of the line

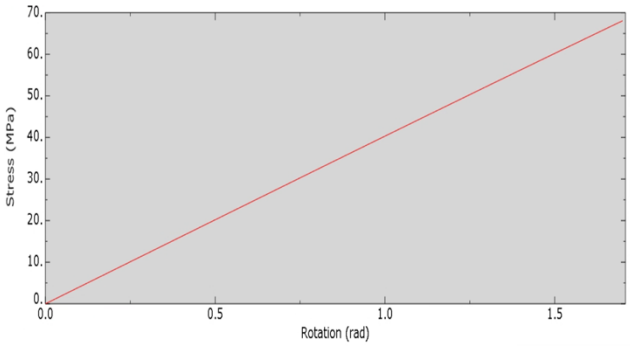

Figure 34: Expected stress-rotation curve for prototype PETG-CF4 in FE.

L. Moment calculations

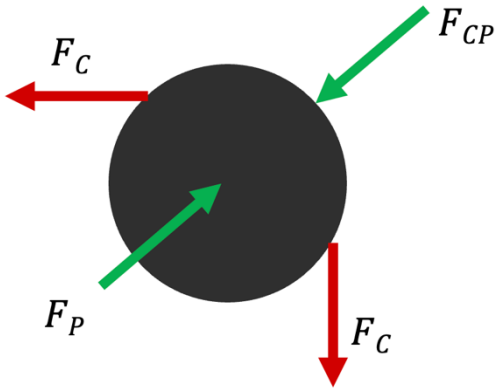

Figure 35: Free body diagram of the pulley. F_C is the tension force in the cable, F_{CP} the force exerted by the cable on the pulley and F_P the reaction force of the pulley

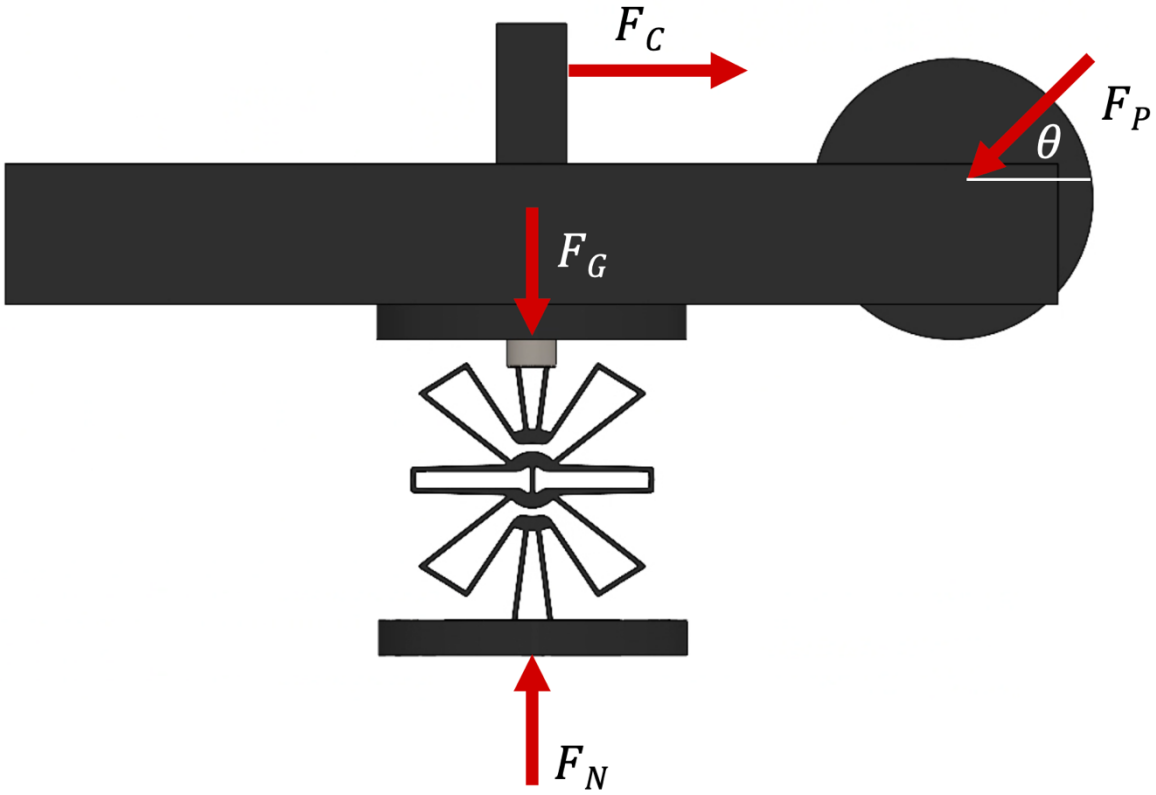

Figure 36: Free body diagram of the hinge in experimental setup. F_C is the tension force in the cable, F_P the reaction force of the pulley, F_G the weight of the system and F_N the normal force, and θ the angle between the pulley reaction force and

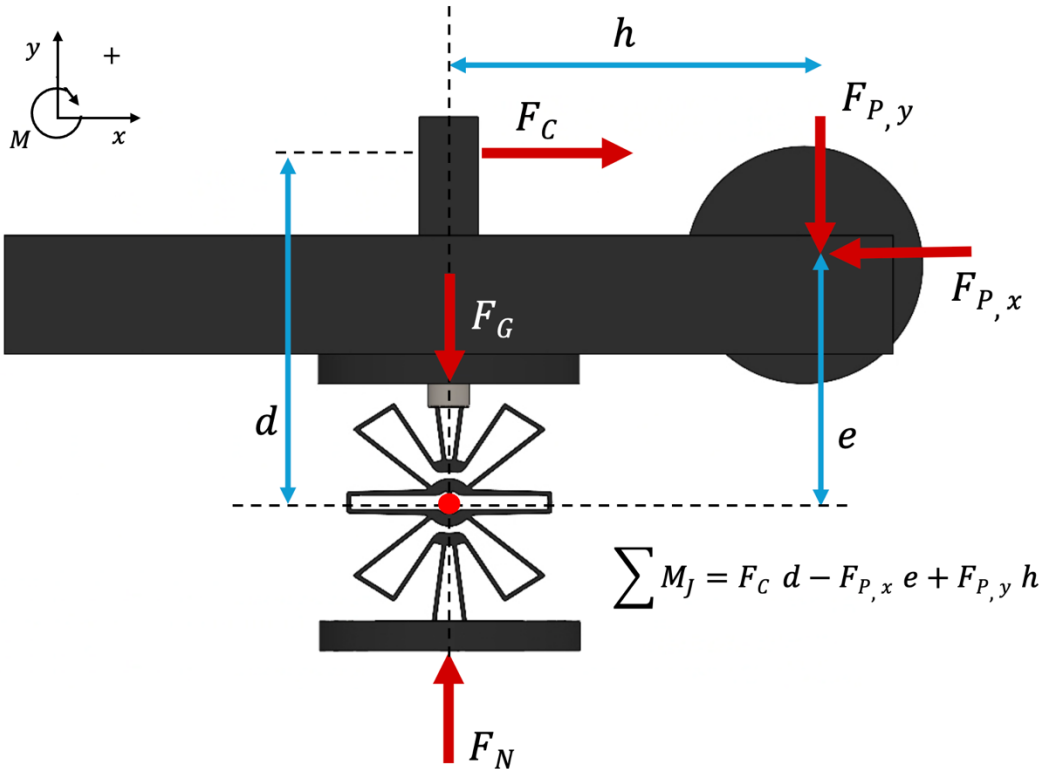

Figure 37: Free body diagram of the hinge in experimental setup. F_C is the tension force in the cable, $F_{P,x}$ and $F_{P,y}$ the horizontal and vertical projections of the reaction force of the pulley, F_G the weight of the system and F_N the normal force. d, h and e are the perpendicular distances to the center of the hinge.

$$\theta = 45° \rightarrow \quad F_{P\ x} = F_{P\ y} = F_C$$

$$\sum M_J = F_C\ d - F_C\ e + F_C\ h$$

$$e = d - 12\ mm$$

$$h = 60\ mm$$

$$\sum M_J = F_C\ d - F_C\ (d - 0.012) + 0.06F_C$$

$$\sum M_J = F_C\ d - F_C\ d + 0.012F_C + 0.06F_C$$

$$\boxed{\sum M_J = 0.072F_C}$$

$$\theta \neq 45° \rightarrow$$

$$F_{P,x} = \sqrt{2}F_C\ \cos(\theta)$$

$$F_{P,y} = \sqrt{2}F_C\ \sin(\theta)$$

$$\sum M_J = F_C\ d - \sqrt{2}F_C\ e\cos(\theta) + \sqrt{2}F_C\ h\sin(\theta)$$

$$\boxed{\sum M_J = F_C\ d - \sqrt{2}F_C\ (d - 0.012)\cos(\theta) + \sqrt{2} * 0.06F_C \sin(\theta)}$$

M. Experimental results for flexion/extension (FE) and lateral bending (LB) tests

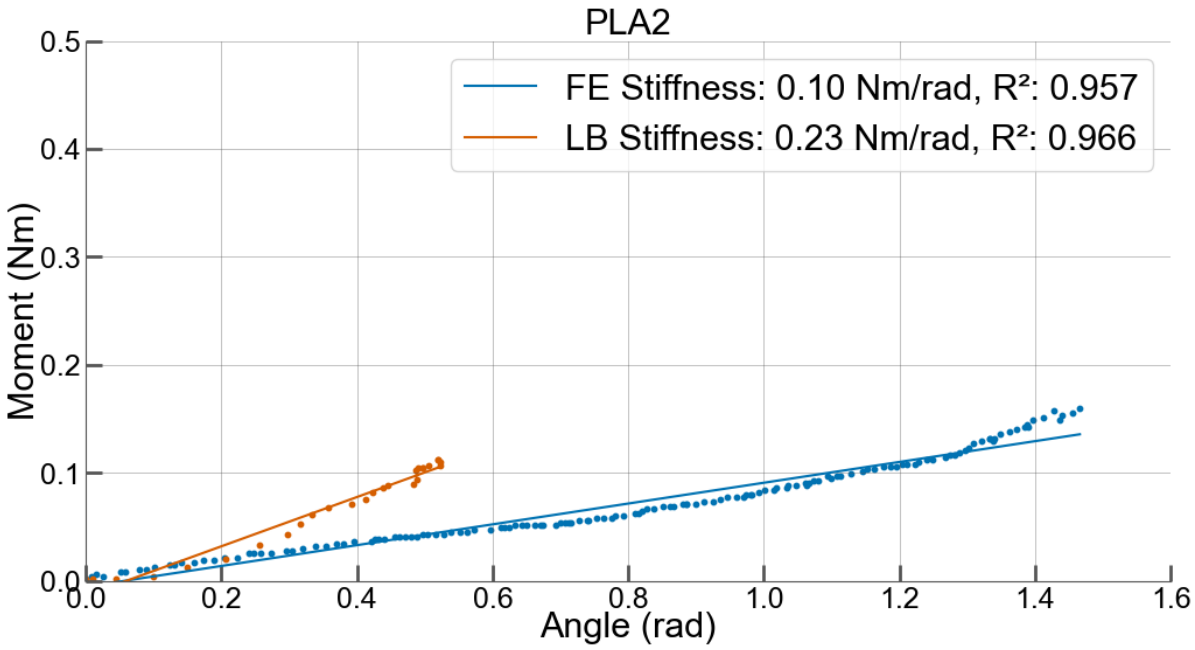

Figure 38: Experimental moment-angle curves for prototype PLA2 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is also shown including R^2 value.

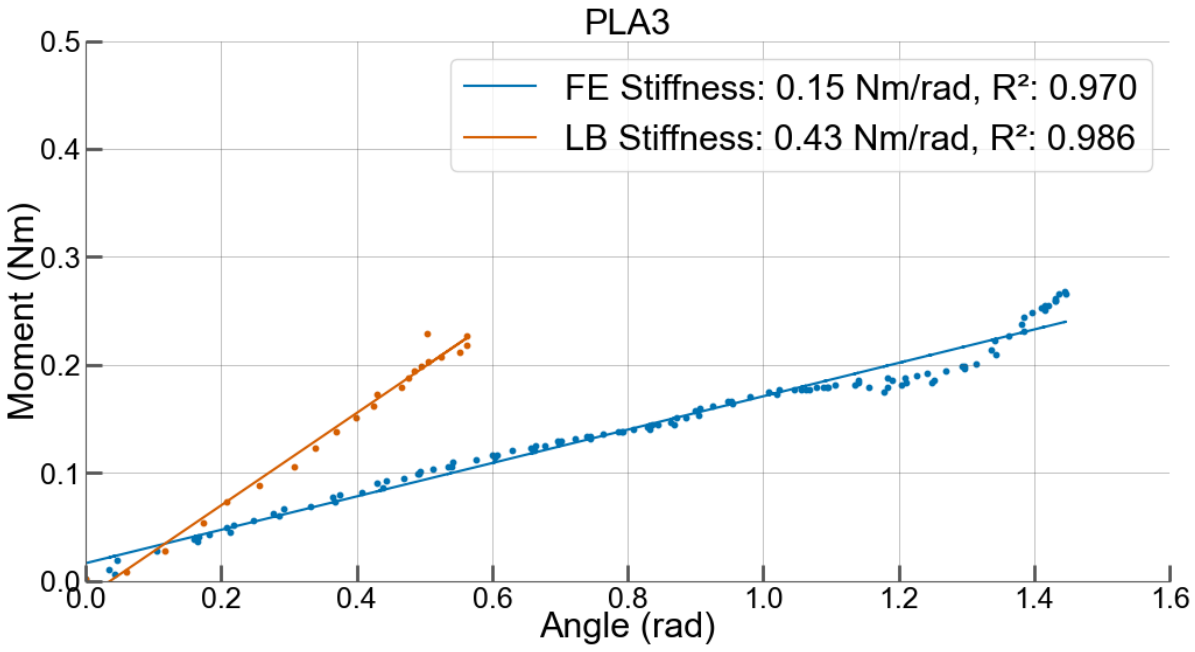

Figure 39: Experimental moment-angle curves for prototype PLA3 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is also shown including R^2 value.

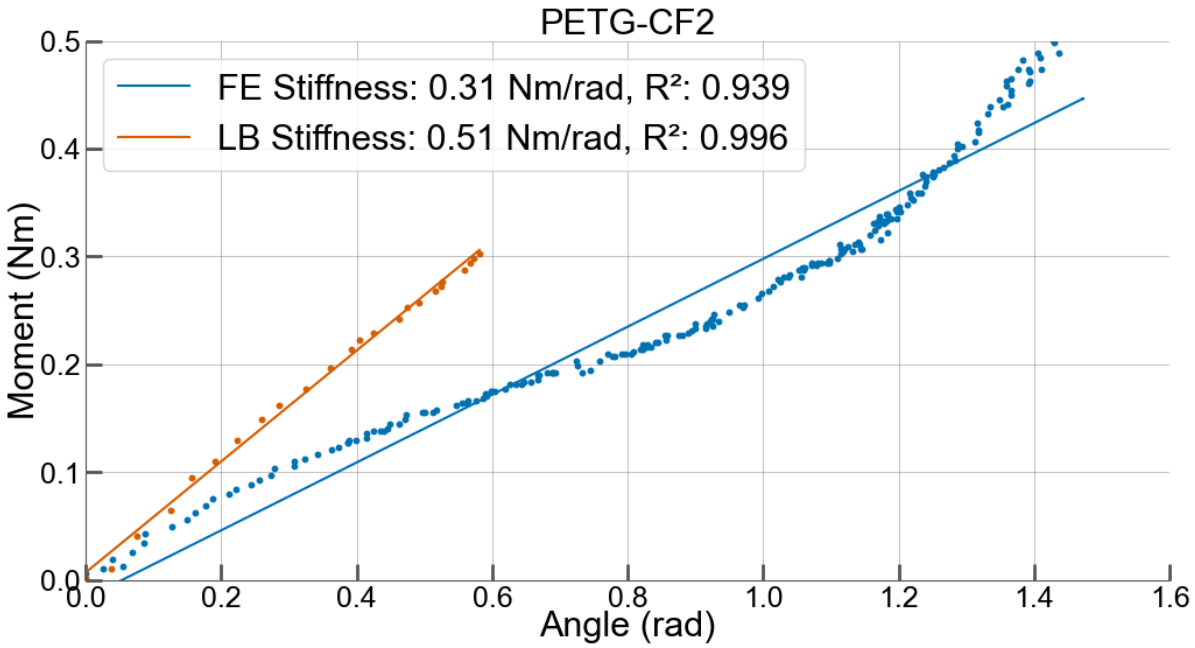

Figure 40: Experimental moment-angle curves for prototypes PETG-CF2 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is also shown including R^2 value.

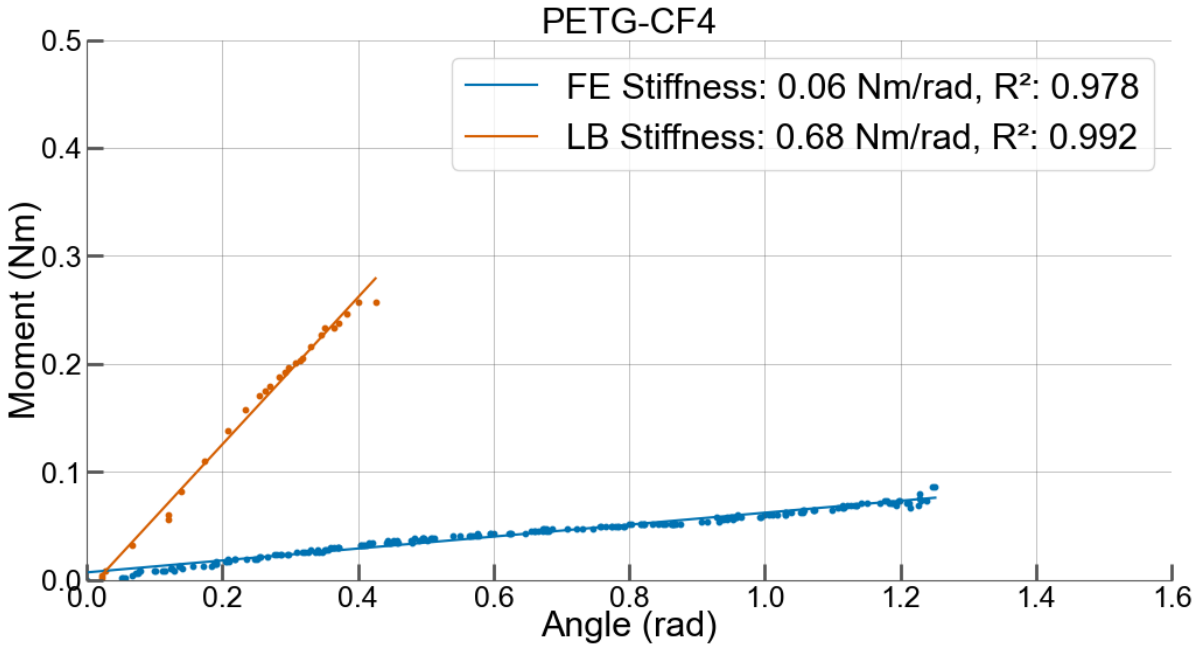

Figure 41: Experimental moment-angle curves for prototype PETG-CF4 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is also shown including R^2 value.

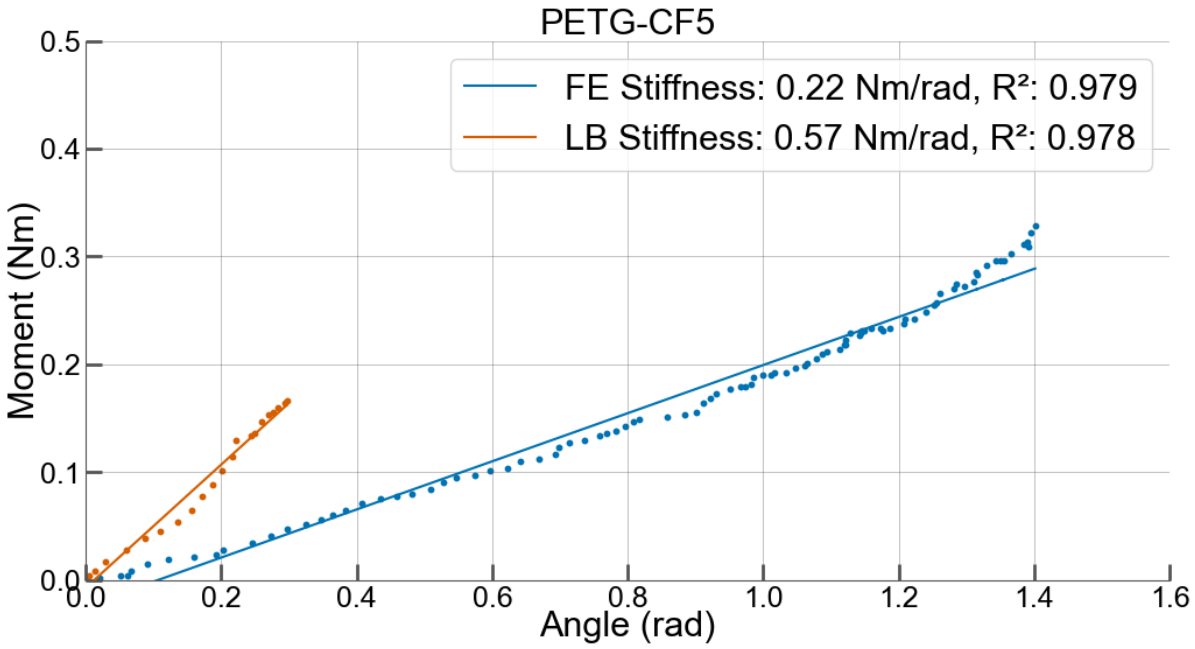

Figure 42: Experimental moment-angle curves for prototype PETG-CF5 in FE and LB. Blue data points represent FE, orange represent LB. The linear fit used to compute the stiffness is also shown including R^2 value.


TUDelft