2010-05-13

Design and Analysis of a Compliant Mechanism Spinal Implant

Eric M. Stratton
Brigham Young University - Provo

Design and Analysis of a Compliant Mechanism Spinal Implant

Eric M. Stratton

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Larry L. Howell, Chair
Anton E. Bowden
Spencer P. Magleby

Department of Mechanical Engineering

Brigham Young University

August 2010

ABSTRACT

Design and Analysis of a Compliant Mechanism Spinal Implant

Eric M. Stratton

Department of Mechanical Engineering

Master of Science

This thesis introduces and presents the modeling of a novel compliant spinal implant designed to reduce back pain and restore function to degenerate spinal disc tissues as well as provide a mechanical environment conducive to healing the tissues. The initial objectives for this device development and the focus of this work are modeling and validation of the force-deflection relationships and stress analysis. Modeling was done using the pseudo-rigid-body model to create a 3 degree of freedom mechanism for flexion-extension (forward-backward bending) and a 5 degree of freedom mechanism for lateral bending (side-to-side). These models were analyzed using the principle of virtual work to obtain the force-deflection response of the device. The model showed good correlation to finite element analysis and experimental results. Also, described in this thesis is a model that incorporates an estimate of the combined stiffness of the biologic structures. This combined model is confirmed by cadaveric testing. A stress analysis of the implant for combined loading conditions is also presented. This work introduces and provides a foundation for the FlexSuRe™ spinal implant.

Keywords: FlexSuRe™, spine, implant, virtual work, compliant

ACKNOWLEDGMENTS

I appreciate Dr. Howell, Dr. Bowden and Dr. Magleby for being those to whom I look to for guidance, examples of excellent researchers and upstanding men. A special thanks to Dr. Howell for helping me work through the rigors and challenges of novel design and valuable research.

My family, my motivation in all I do, has supported me and provided a balanced, enjoyable life for which I am grateful.

I gratefully acknowledge Crocker Spinal Technologies for recognizing the value of this research and licensing the technology from Brigham Young University. I wish them success in further development and commercialization of this device to ultimately help people be free from debilitating back pain.

The financial support of the U.S National Science Foundation, through Grant No. CMMI-0800606, is gratefully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION AND BACKGROUND

1.1 Introduction

The motivation of this work is development of a device that returns the degenerate motion of the functional spinal unit (FSU) nearer to normal, with minimal disruption to existing tissues. The objective of this work is to introduce a compliant mehcanism spinal implant that shows provmis for such results and to develop the mathematical models for the force-deflection relationship of the device and the force-deflection relationship of the combined device and biologic stiffness.

Significant problems with fixation, such as pain recurrence and adjacent segment disease [2–5], has led researchers to consider alternate approaches. Next generation devices are minimally invasive, minimally traumatic, and provide pain relief through restoration of normal motion [3, 6]. Spinal implants have moved toward posterior dynamic stabilization as one method of pain reduction with a less invasive procedure. Most of these devices focus on removing most of the motion of the FSU, approximating fusion [6]. Other devices allow significantly more motion but still lack the ability to reduce pain through restoration of normal pain-free motion. This work presents a posterior dynamic stabilization implant that works in concert with the existing tissues of the FSU to return the degenerate FSU to a normal motion pattern. The work here introduces and provides a foundation for development of the FlexSuRe™ spinal implant. The name FlexSuRe™ comes from its ability to maintain *Flex*ibility while *Su*pporting a portion of the load in an attempt to *Re*store normal motion of the FSU.

1.2 Outline of Thesis

The remainder of this chapter presents the background to support later discussion. Chapter 2 presents the compliant mechanism spinal implant and its features, and derives a force deflection

model for flexion-extension, lateral bending, and axial rotation. Also included in this chapter is physical testing and finite element analysis to confirm the derivations. Chapter 3 presents additional analysis that incorporates the kinematic signature of the biologic structure as well as initial cadaveric testing that confirms the proposed analysis method. Chapter 4 presents the stress analysis of the device. Chapter 5 reviews the work presented as well as future areas of research to culminate in a successful next generation spinal implant.

1.3 Background

Back pain due to degraded spinal discs is a major problem in the world. In the United States alone, 349,000 spinal fusion procedures are performed annually at a cost of $20.2 billion per year. Projections for the year 2030 indicate 5 million lumbar fusions, resulting in a 2,200% growth [7]. Although fusion of the vertebral bodies is the current surgical standard for some causes of back pain, it does not reduce the patient's problematic pain in approximately half the treatments. Further, some contend that fusion increases the rate of degradation of the adjacent spinal discs due to the greater loads and deflections adjacent discs must carry [5].

Other treatment options for back pain include spinal disc arthroplasty and dynamic stabilization. Disc arthroplasty consists of removing the damaged or diseased biological spinal disc and replacing it by an orthopedic device. Current total disc replacement (TDR) technologies rely on mechanisms which have inherent disadvantages, including wear particles introduced into the body [8, 9], non-natural kinematics [10] and non-natural kinetics. Dynamic stabilization consists of devices attached to the posterior spine and reduce the abnormal motion but do not allow normal motion. The next generation of spinal devices will provide for functional restoration of the functional spinal unit (FSU) while allowing for possible regeneration of the biologic tissues and structures of the spine.

1.3.1 Description of Anatomical Structure

The intervertebral disc (IVD) is structurally composed of two regions: the nucleus pulposus and the annulus fibrosis. The nucleus pulposus (the center) is a gelatinous substance. It constitutes the main compressive-load-bearing component. The annulus fibrosis (the outer ring) is a fibrous

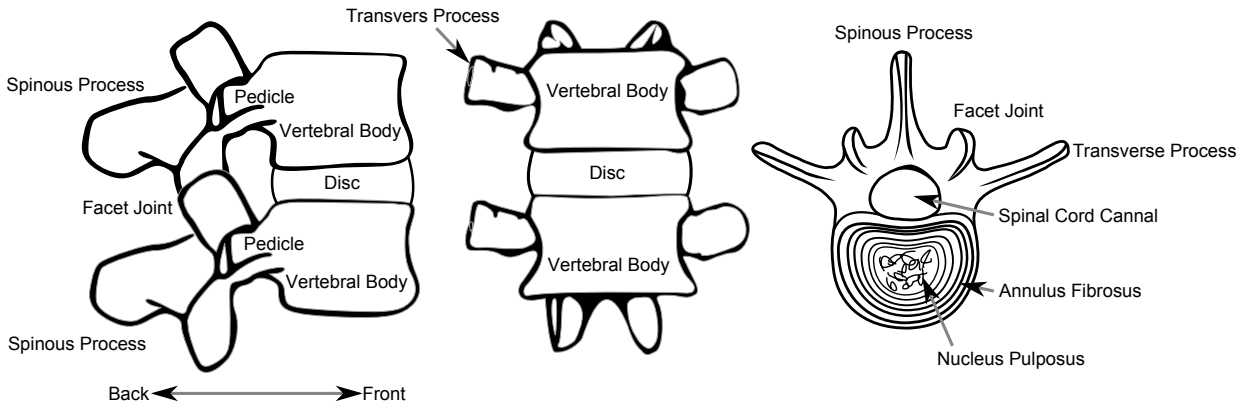

Figure 1.1: Spine anatomy.

material that alternates directions from layer to layer. The purpose of the annulus is to contain the nucleus pulposus and to contribute to the disc's force bearing capabilities.

The IVD is the largest avascular structure in the body and it relies on diffusion of nutrients. Passive diffusion is aided by bulk fluid flow during compression-relaxation cycles that occur during daily activities. The loading of the disc during motion compresses the disc, which drives fluid and waste products out. Resting reduces the load on the disc, allowing nutrients to diffuse into the disc aided by fluid transfer. These nutrients are essential for the good health and biological repair of the disc.

Other biologic structures of concern are the numerous ligaments and tendons attached to the vertebral body, the facet joints, as well as the spinal cord and the nerve roots. The ligaments are essential for proper motion and stability. The facets are capsule-enclosed surfaces that come in contact to prevent excessive movement of the vertebral bodies. The two facet joints and the intervertebral body represent a three-joint complex that work in unison for spinal motion. The nerve roots exit the spinal cord between the pedicles. Some of these biologic structures are shown in Figure 1.1.

1.3.2 Degenerate Intervertebral Disc

The structural lumbar spine elements produce a specific, normal motion pattern. As the disc deteriorates this motion pattern changes causing pain through excessive stretching or impacting of structures discussed previously. The body attempts to compensate for the altered kinematics

by increasing muscle activation to attempt to regain stability [6]. As the biologic tissues of the intervertebral disc degrade, the motion pattern becomes degenerate and disc height decreases. To alleviate pain and prevent further degeneration of adjacent levels of the spine, the motion pattern must be restored. Disc height must also be restored to lift the vertebral bodies off of the nerve roots [2, 6] and to relieve a portion of the load on the disc and facets.

1.3.3 Posterior Dynamic Stabilization

Posterior dynamic stabilization can be divided into two categories: essentially fusion devices and motion preservation devices. Essentially fusion devices are traditionally used as adjunct to fusion. The goal of these devices is to stabilize the spine until the bone graft can solidify or fuse the FSU while allowing small amplitude motions for bone growth stimulus. Motion preservation devices attempt to add stiffness to the FSU, allowing significantly more motion than essentially fusion devices. Within the motion preservation category are two subcategories: the single-function and multiple-function subcategories. These are summarized in Table 1.1.

Single-function devices focus on strategies to provide line-of-action forces between the pedicle screw attachments. Multiple-function devices provide line of action forces and applied torques. Single-function devices sometimes use ball joints, which ideally transmit no torque, limit the device to simple line-of-action force due to displacement of the pedicles. An advantage of a single-function device is that there is limited torque on the bone-screw interface, reducing occurrence of screw loosening [11]. A limitation is that the only adjustable kinetic parameter is the axial stiffness of the device.

In addition to line-of-action force, multiple-function devices provide torque stiffness due to relative rotation of the pedicles. A flexible rod connected via pedicle screws would fall into this category if it allows significant motion relative to motion limiting devices. However, flexible rods are designed to provide axial stiffness while their torque characteristics are considered a consequence of their axial stiffness, not as a tailorable parameter for spinal stability. A concern is the torque applied to the pedicle screw. Techniques of preventing screw loosening could be incorporated to negate this concern.

The implant incorporates a desired stiffness through a geometry with many parameters to adjust the stiffness in various loading modes. The device geometry can be dimensionally designed

Table 1.1: Posterior Dynamic Stabilization Classification

Posterior Dynamic Stabilization	
Category	Description
Motion Limiting	*Approximate Fusion*
Stiff Constructs	*Relatively Rigid Rods*
Pin Joint Constructs	*Abnormal Rotation Point*
Motion Preservation	*Stabilize Motion*
Single-Function	*Line of Action Force*
Multiple-Function	*Force and Torque*

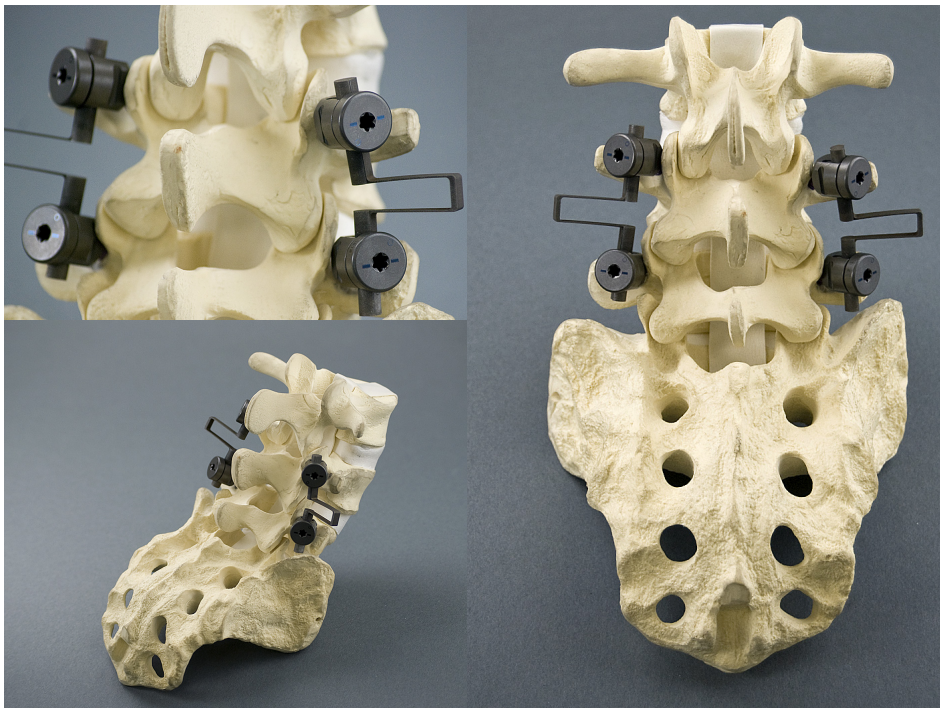

Figure 1.2: Illustrative FlexSuRe™ prototype.

to have a balanced effect on multiple motion patterns. Resultant forces and torques assist the biologic structures in flexion-extension, lateral bending, axial compression, and axial rotation. A illustrative prototype of one version of the implant, the FlexSuRe™, is shown in Figure 1.2.

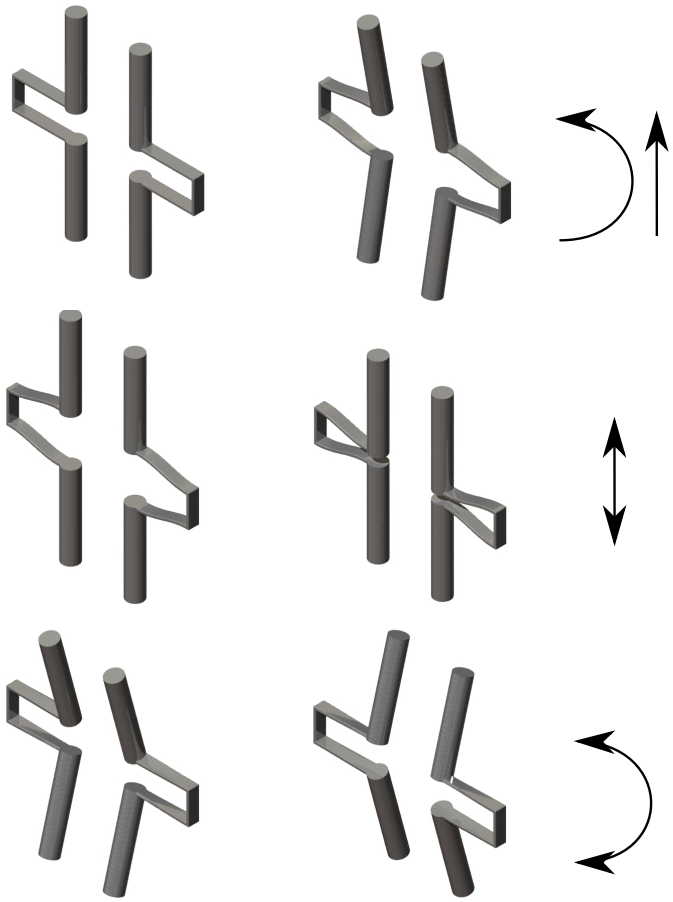

Figure 1.3: Deflected postions.

1.4 Compliant Mechanisms

Compliant mechanisms use deflection of their parts to provide a force-deflection response [12]. One advantage of compliant mechanisms is their ability to eliminate pin joints and other articulating contact surfaces that produce wear particles. Wear particles are highly detrimental to the function of an implant over time. Wear particles are detrimental to the health of the surrounding tissues and have caused significant problems with implant fixation [8, 13]. Other advantages include reduction in parts, limited or no assembly, single-piece design, simplified manufacturing and increased performance. Compliant mechanism provide complex motions with fewer components. Figure 1.3 shows the device deflected in various positions. The spine is a natural compliant mechanism, utilizing deflection of tissues for complex motion.

1.5 Functional Specifications

The implant is designed to reduce pain caused by abnormal spinal motion or position. The kinematics, such as the instantaneous axis of rotation, should be positioned in the lower portion of the back third of the disc space [14]. The kinetics of the FSU, such as the torque-rotation curves for the component motion patterns and the axial force-deflection response, should be adjusted to a more normal state. Addition of the stiffness that is lost due to early-stage degeneration [6] or altered stiffness due to disc height increase should return the kinetics to normal. The adjustable positioning of the vertebral bodies to distract the posterior elements reduces nerve root impingement. The anterior disc space should be distracted to obtain a more normal attitude of the vertebral body relative to the adjacent vertebral bodies. Further, this anterior distraction can reduce the load on the disc to provide a mechanical environment in which healing could possibly occur [3, 6]. It is desirable to have a single-piece implant that produces no wear due to articulating surfaces [9]. Additionally, the implant is designed to be minimally invasive and cause minimal trauma.

1.6 Possible Pre-Loaded Configuration

The implant's functional geometry exhibits a force-deflection response that aids the biologic structure motion pattern through forces and torques in all directions of motion. The force-deflection relationship of many loading conditions is well defined for this device geometry. Although not demonstrated in this thesis the implant shows promise that it can be pre-compressed for integrated distraction of the vertebral bodies. Further, it could be possible to pre-torque the implant to provide a lifting and rotating force on the anterior vertebral body. A combination of pre-compression and pre-torque could distract the posterior elements and the anterior elements. The same pre-stresses could be applied asymmetrically to provide unbalanced forces for treatment of a scoliotic spine.

CHAPTER 2. FORCE-DEFLECTION MODEL OF THE SPINAL IMPLANT

2.1 Introduction

The spinal implant is a compliant mechanism device being developed to restore normal motion to the degenerate spine. The intent is that the implant will provide a force-displacement response that, when combined with a degenerated intervertebral disc, will result in normal, pain free motion.

The ability to quickly simulate the force-deflection response of the implant during the design phase of the device is especially advantageous in evaluating device dimensions and tolerances. An analytical model of the force-displacement of the implant would help facilitate this goal and is the subject of this chapter.

First, some background on the spine is provided, followed by a description of the implant. The principle of virtual work combined with the pseudo-rigid-body model is used to develop the force-displacement model. The model is then compared to physical measurements and finite element model results. The model described here is valid for a wide range of geometries, including the FlexSuRe™ spinal implant, which was developed as part of this work.

2.2 Spinal Implant

The objective of the spinal implant is to assist in the restoration of normal motion to the degenerate spine and relieve a portion of the load on the disc and facet joints. The concept of restoring function is described by the torque-rotation curves shown in Fig. 2.1. The normal and degenerate curves are typical flexion-extension curves common in spinal biomechanics [15–17]. Other curves in this figure are derived from these normal and degenerate curves to show the methodology. The normal motion pattern degrades to the degenerate torque-rotation curve (top). The difference of these curves is the torsional stiffness that is lost due to collagen fiber disorganization and loss of

hydration (middle). The linear approximation of the difference is also shown (middle). If a device is integrated into the functional spinal unit with a torsional stiffness equal to the linear approximation of the difference curve, then the device and the degenerate tissues together return the motion pattern to the normal case (bottom). Thus, the change due to degeneration becomes the target performance of the proposed implant. The focus of this paper is the derivation of a model for the analysis of the force-displacement behavior of the implant.

2.2.1 Implant Requirements

The foremost requirement of a spinal implant is to eliminate chronic back pain for the patient. The spinal implant is proposed to do this with the following design requirements:

1. The implant must protect the spinal cord and the nerve roots from damage.
2. The implant must provide range of motion of a normal functional spinal unit to reduce the risk of overloading the adjacent motion segments. When one segment is limited in the amount of motion it can achieve then other motion segments must accommodate an increased portion of the load or deflection.
3. The implant should avoid wear. Problems have occurred because of the wear particles produced in the rubbing surfaces in traditional implants [9]. The wear particles can cause implant loosening because of bone degradation, or tissue degradation because of contamination.
4. The kinetics of the spine must be close to normal to reduce muscle fatigue.
5. The kinematics, such as the center of rotation, must be maintained to reduce the amount of strain of the muscles and tendons [14].
6. This implant should relieve a portion of the compressive load. This can increase disc height, eliminate tissue impingements, and facilitate possible disc healing.
7. The implant should restore normal torque-rotation signature of the FSU.

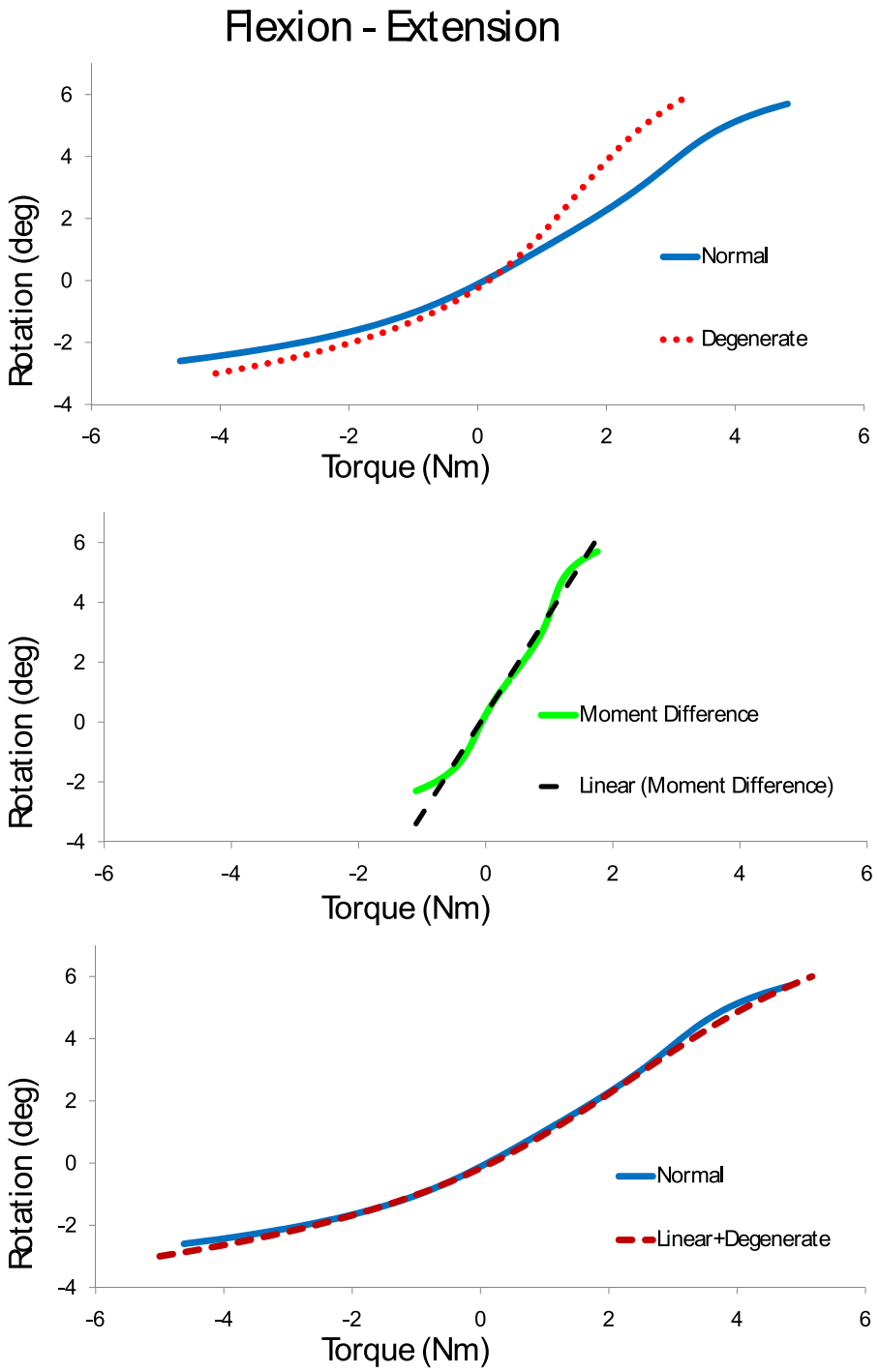

Figure 2.1: Torque rotation curves. Top: normal and degenerate. Middle: difference and linear approximation. Bottom: normal and combination.

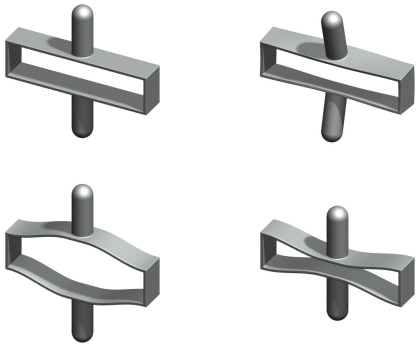

Figure 2.2: Top-Left: LET joint. Top-Right: torque. Bottom-Left: tensile. Bottom-Left: compression.

2.2.2 Device Description

The implant described here is a compliant mechanism based on the Lamina Emergent Torsional (LET) joint [18]. The LET joint is designed to be made from a sheet or lamina and emerge out of the plane of fabrication using mainly torsion of beams. This geometry allows flexibility in many directions, much like the intervertebral disc. Figure 2.2 shows the manufactured shape and deflected configurations of the LET joint. This functional geometry allows significant design flexibility.

The device consists a LET joint that has been split into two parts, that are affixed to the vertebral bodies, as shown in the FlexSuRe™ configuration of Figure 2.3. These two components assist the degenerated biologic tissues to restore normal function of the spinal unit. The implant has no joints or rubbing surfaces, thus increasing reliability and reducing the production of wear particles.

The device can be implanted with a positioning tool that can be used to place it in a configuration that is easily attached to the spine using pedicle screws (pedicle screws are commonly used to attach spinal devices to the posterior spine). The positioning tool places the device in a pre-strained configuration so that when the device is attached to the pedicle screws and released from the tool, it provides a force and a torque to lift and rotate the vertebral body. This pre-strained torque lifts the anterior and posterior portions of the vertebral body, relieving a portion of the load on the intervertebral disc. The pre-strained compression lifts the posterior portion of the disc and also relieves a portion of load on the facets by increasing disc height. In addition, it may be possi-

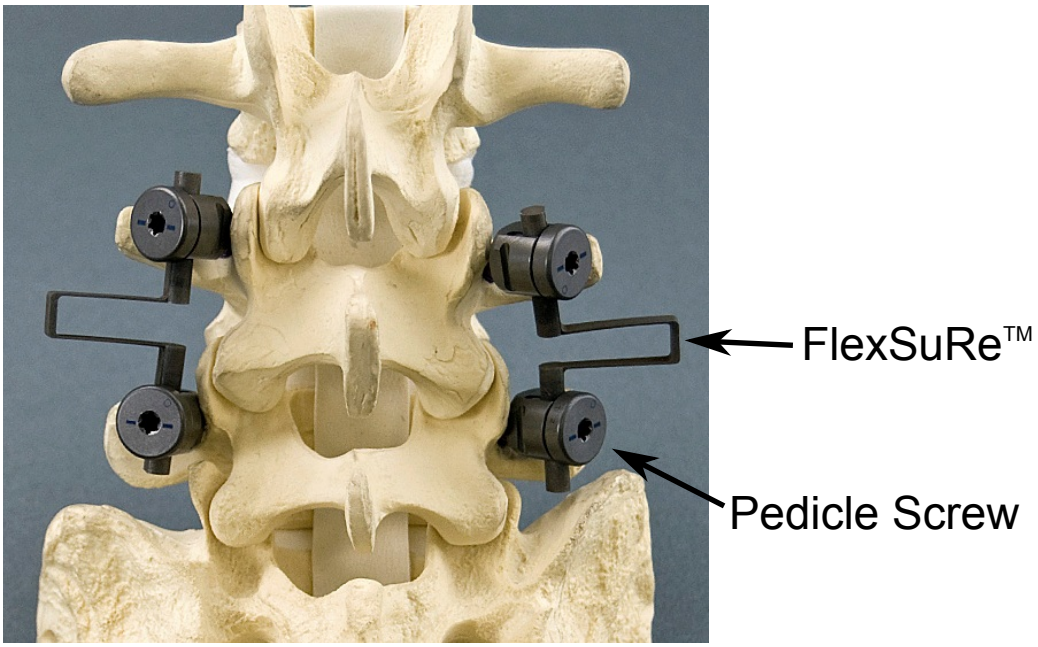

Figure 2.3: FlexSuRe™ implant (proof of concept).

ble to treat scoliosis by inducing unbalanced forces on the vertebral bodies to gradually force the spine into a healthy curvature by this same pre-strained procedure.

2.3 Mechanism Analysis

2.3.1 Pseudo-Rigid-Body Model

Compliant mechanisms traditionally are difficult to design and analyze because of their nonlinear motion and the combined motion and stress relationships. The pseudo-rigid-body model equates a compliant beam with a rigid-link, characteristic pivot, and corresponding torsional spring to simulate the energy storage in the deflected compliant beam. Through the pseudo-rigid-body model we obtain the advantages of rigid-link mechanism analysis with the characteristic advantages of compliant mechanisms, while taking into account large nonlinear motions of compliant elements [12].

For flexion-extension, the pseudo-rigid-body model was used to model the system with two torsional joints at the location of the horizontal beam and a linear spring representing the compression stiffness of the joint. This kinematic unit is depicted in Figure 2.4. Extending this concept to

Figure 2.4: LET pseudo-rigid-body model for flexion-extension.

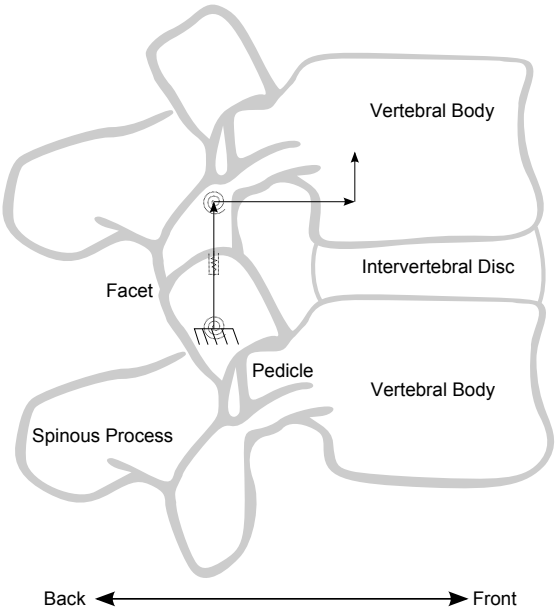

Figure 2.5: Pseudo-rigid-body model for flexion-extension with force application.

include the location of force application results in Figure 2.5, which represents a pseudo-rigid-body model of the mechanism. On the left side of the mechanism is a kinematically equivalent combination of torsional springs, K_1 and K_2, plus linear spring, K_{let} as shown in Figure 2.6. Similarly, the pseudo-rigid-body model for lateral bending and for axial compression of the spinal column is shown in Figure 2.7. This model consists of four fixed-guided beams with torsional springs at each characteristic pivot.

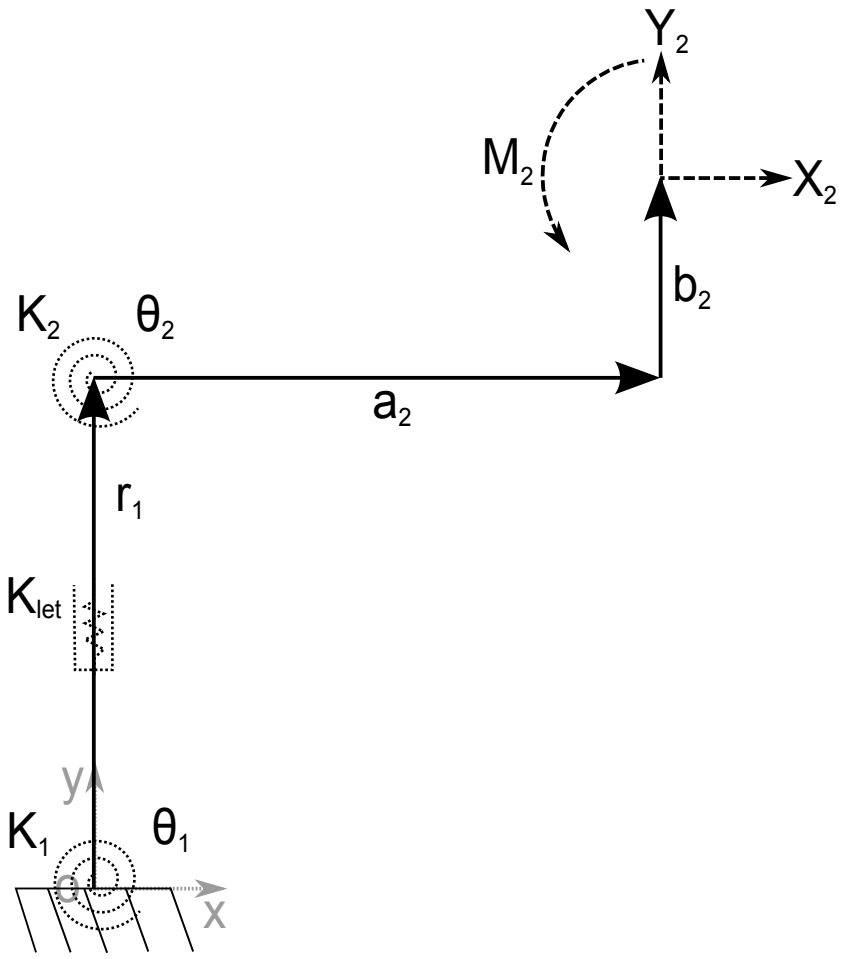

Figure 2.6: Pseudo-rigid-body model for the virtual work analysis of flexion-extension.

2.3.2 Principle of Virtual Work

The principle of virtual work is a method of obtaining the equilibrium position of a mechanism without the need to calculate the reaction forces at each joint. This method produces an equation for each degree of freedom of the mechanism and the resulting set of equations can be solved simultaneously. The principle of virtual work is used here to determine the force-deflection relationship of the pseudo-rigid-body model of the mechanism.

The implant uses torsion of the horizontal beams to allow for greater range of motion for each joint. The combined motion of compression and rotation of the implant complicates the model of the motion. The compression of the joint essentially changes the link lengths during the

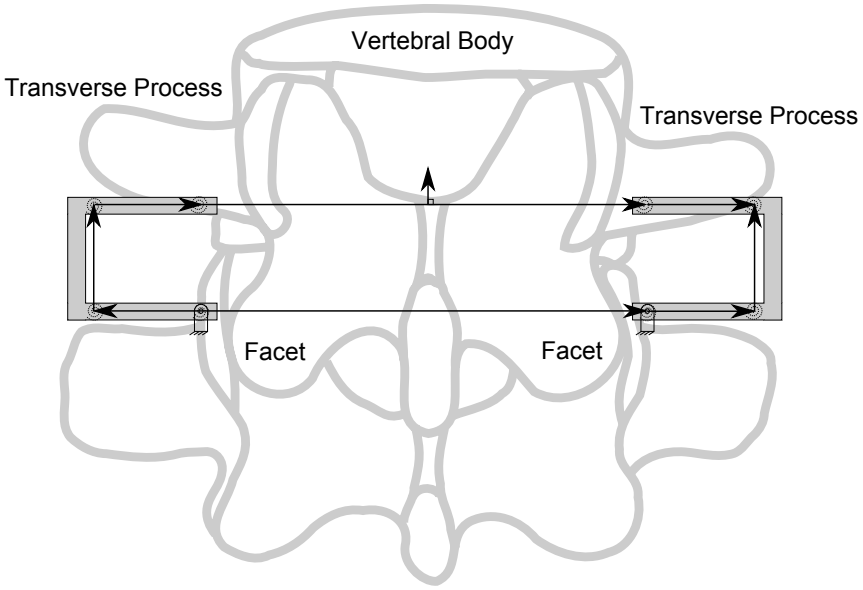

Figure 2.7: Lateral bending pseudo-rigid-body model with spine.

motion. Models were derived for flexion-extension (forward-backward bending) and for lateral bending (side to side bending).

2.3.3 Flexion-Extension Model

Figure 2.6 represents the related geometry and forces for the virtual work calculations for flexion-extension. The springs represented by K_1 and K_2 are the torsional segments of the LET joint. The spring constant, K_{let}, represents the extension/compression of the LET joint. All angles are define from the positive x-axis.

The vector form of the applied forces, $\vec{F}$, from the anatomical structure is

$$\vec{F} = X_2\hat{i} + Y_2\hat{j} \tag{2.1}$$

where X_2 and Y_2 are the applied forces in the x-direction and y-direction, as shown in Figure 2.6. Also the spring force due to K_{let} will be treated as an external force in the direction of $\vec{r}_1$. The vector form of this force, $\vec{F}_s$ is

$$\vec{F}_s = F_s\cos\theta_1\hat{i} + F_s\sin\theta_1\hat{j} \tag{2.2}$$

where F_s is defined as a function of r_1 and the initial length, r_{10}, through the following equations

$$K_{let} = \frac{2\gamma K_\Theta EI}{L_{tl}} \tag{2.3}$$

$$\Theta = \sin^{-1}\left(\frac{r_1 - r_{10}}{2\gamma L_{tl}}\right) \tag{2.4}$$

$$F_s = \frac{-4K_{let}\Theta}{\cos\Theta\gamma L_{tl}} \tag{2.5}$$

where K_{let} is the spring constant associated with the axial stiffness of the LET joint. K_Θ, γ, E, I and L_{tl} are the stiffness coefficient, characteristic radius factor, modulus of elasticity, moment of inertia and length of the torsion beam. The pseudo-rigid-body angle is Θ. The location of these forces with respect to the defined origin are

$$\begin{aligned} \vec{z} = \quad & (r_1\cos\theta_1 + a_2\cos\theta_2 - b_2\sin\theta_2)\,\hat{i} \\ & + (r_1\sin\theta_1 + a_2\sin\theta_2 + b_2\cos\theta_2)\,\hat{j} \end{aligned} \tag{2.6}$$

$$\vec{z}_s = r_1\cos\theta_1\,\hat{i} + r_1\sin\theta_1\,\hat{j} \tag{2.7}$$

The subscript "s" refers to the axial spring approximation of the LET joint.

A generalized coordinate is needed for each of the three degrees of freedom. The generalized coordinates, q_i, are selected as

$$q_i = r_1,\ \theta_1,\ \theta_2 \tag{2.8}$$

Differentiating the first position vector with respect to a generalized coordinate gives

$$\begin{aligned} \frac{d\vec{z}}{dq_i} = \quad & \left(\cos\theta_1\frac{dr_1}{dq_i} - r_1\sin\theta_1\frac{d\theta_1}{dq_i}\right. \\ & \left. -a_2\sin\theta_2\frac{d\theta_2}{dq_i} - b_2\cos\theta_2\frac{d\theta_2}{dq_i}\right)\hat{i} \\ + \quad & \left(\sin\theta_1\frac{dr_1}{dq_i} + r_1\cos\theta_1\frac{d\theta_1}{dq_i}\right. \\ & \left. +a_2\cos\theta_2\frac{d\theta_2}{dq_i} - b_2\sin\theta_2\frac{d\theta_2}{dq_i}\right)\hat{j} \end{aligned} \tag{2.9}$$

$$\begin{aligned}\frac{d\vec{z}_s}{dq_i} = & \quad (\cos\theta_1 \frac{dr_1}{dq_i} - r_1 \sin\theta_1 \frac{d\theta_1}{dq_i})\hat{i} \\ & + \quad (\sin\theta_1 \frac{dr_1}{dq_i} + r_1 \cos\theta_1 \frac{d\theta_1}{dq_i})\hat{j} \end{aligned} \tag{2.10}$$

Differentiating the first position vector with respect to each of the generalized coordinates results in

$$\frac{d\vec{z}}{dr_1} = \cos\theta_1 \hat{i} + \sin\theta_1 \hat{j} \tag{2.11}$$

$$\frac{d\vec{z}}{d\theta_1} = -r_1 \sin\theta_1 \hat{i} + r_1 \cos\theta_1 \hat{j} \tag{2.12}$$

$$\begin{aligned}\frac{d\vec{z}}{d\theta_2} = & \quad (-a_2 \sin\theta_2 - b_2 \cos\theta_2)\hat{i} \\ & + \quad (a_2 \cos\theta_2 - b_2 \sin\theta_2)\hat{j} \end{aligned} \tag{2.13}$$

$$\frac{d\vec{z}_s}{dr_1} = \cos\theta_1 \hat{i} + \sin\theta_1 \hat{j} \tag{2.14}$$

$$\frac{d\vec{z}_s}{d\theta_1} = -r_1 \sin\theta_1 \hat{i} + r_1 \cos\theta_1 \hat{j} \tag{2.15}$$

$$\frac{d\vec{z}_s}{d\theta_2} = 0\hat{i} + 0\hat{j} \tag{2.16}$$

The virtual work due to the applied forces, δW_f, is calculated by taking the dot product of the force vector and the virtual displacement as

$$\delta W_f = \vec{F} \cdot d\vec{z} \tag{2.17}$$

$$\begin{aligned}\delta\vec{W}_f = & \quad (X_2 \cos\theta_1 + Y_2 \sin\theta_1)\delta r_1 \\ & + \quad (-X_2 r_1 \sin\theta_1 + Y_2 r_1 \cos\theta_1)\delta\theta_1 \\ & + \quad [X_2(-a_2 \sin\theta_2 - b_2 \cos\theta_2) \\ & \quad + Y_2(a_2 \cos\theta_2 - b_2 \sin\theta_2)]\delta\theta_2 \end{aligned} \tag{2.18}$$

The virtual work due to $\vec{F}_s$, δW_s, is

$$\delta \vec{W}_s = F_s \delta r_1 \tag{2.19}$$

The applied moment, M_2, and the torsion due to torsional springs, T_1 and T_2, are expressed in vector form as

$$T_1 = -K_1(\theta_1 - \theta_{10})\hat{k} \tag{2.20}$$

$$T_2 = -K_2[(\theta_2 - \theta_{20}) - (\theta_1 - \theta_{10})]\hat{k} \tag{2.21}$$

$$M_2 = M_2\hat{k} \tag{2.22}$$

where θ_{10} and θ_{20} are the initial angles of the mechanism. The virtual work due to these moments, δW_m, is

$$\delta \vec{W}_m = (T_1 - T_2)\delta\theta_1 + (T_2 + M_2)\delta\theta_2 \tag{2.23}$$

The total virtual work is calculated by summing the virtual work terms. Employing the principle of virtual work (the virtual work is zero if and only if the system is in equilibrium) results in the following three equations (one equation for each degree of freedom):

$$(X_2 \cos\theta_1 + Y_2 \sin\theta_1 + F_s)\delta r_1 = 0 \tag{2.24}$$

$$(-X_2 r_1 \sin\theta_1 + Y_2 r_1 \cos\theta_1 + T_1 - T_2)\delta\theta_1 = 0 \tag{2.25}$$

$$\begin{aligned}[X_2(-a_2 \sin\theta_2 - b_2 \cos\theta_2) \\ + Y_2(a_2 \cos\theta_2 - b_2 \sin\theta_2) \\ + T_2 + M_2]\delta\theta_2 = 0\end{aligned} \tag{2.26}$$

Solving these equations simultaneously results in the equilibrium position of the device.

2.3.4 Lateral Bending Model

The pseudo-rigid-body model for lateral bending with applied forces and torque locations is shown in Figure 2.7 and Figure 2.8. The vector form of the applied forces is

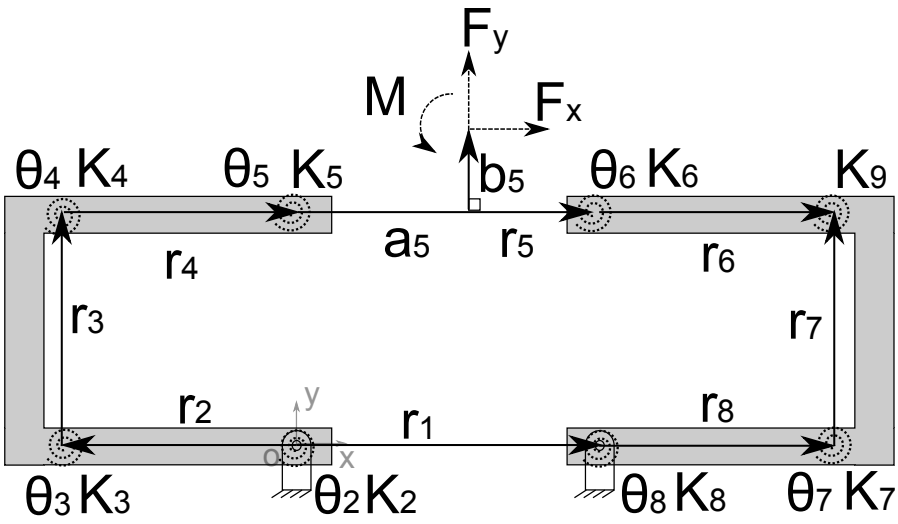

Figure 2.8: Lateral bending pseudo-rigid-body model.

$$\vec{F} = F_x\hat{i} + F_y\hat{j} \tag{2.27}$$

The location of this force with respect to the defined origin is

$$\begin{aligned}\vec{z} = \quad & (r_2\cos\theta_2 + r_3\cos\theta_3 + r_4\cos\theta_4 \\ & + a_5\cos\theta_5 - b_5\sin\theta_5)\hat{i} \\ + \quad & (r_3\sin\theta_3 + r_4\sin\theta_4 \\ & + a_5\sin\theta_5 + b_5\cos\theta_5)\hat{j}\end{aligned} \tag{2.28}$$

A generalized coordinate is needed for each degree of freedom. The generalized coordinates q_i are

$$q_i = \theta_i, (i = 2,3,4,5,6) \tag{2.29}$$

Differentiating the position vector, $\vec{z}$, with respect to a generalized coordinate gives:

$$\begin{aligned}\frac{d\vec{z}}{dq_i} = \quad & [-r_2\sin\theta_2\frac{d\theta_2}{dq_i} - r_3\sin\theta_3\frac{d\theta_3}{dq_i} \\ & -r_4\sin\theta_4\frac{d\theta_4}{dq_i} - (a_5\sin\theta_5 + b_5\cos\theta_5)\frac{d\theta_5}{dq_i}]\hat{i} \\ + \quad & [r_2\cos\theta_2\frac{d\theta_2}{dq_i} + r_3\cos\theta_3\frac{d\theta_3}{dq_i} \\ & +r_4\cos\theta_4\frac{d\theta_4}{dq_i} + (a_5\cos\theta_5 - b_5\sin\theta_5)\frac{d\theta_5}{dq_i}]\hat{j} \end{aligned} \tag{2.30}$$

Differentiating the position vector with respect to each specific q_i results in

$$\frac{d\vec{z}}{d\theta_2} = -r_2\sin\theta_2\hat{i} + r_2\cos\theta_2\hat{j} \tag{2.31}$$

$$\frac{d\vec{z}}{d\theta_3} = -r_3\sin\theta_3\hat{i} + r_3\cos\theta_3\hat{j} \tag{2.32}$$

$$\frac{d\vec{z}}{d\theta_4} = -r_4\sin\theta_4\hat{i} + r_4\cos\theta_4\hat{j} \tag{2.33}$$

$$\begin{aligned}\frac{d\vec{z}}{d\theta_5} = \quad & (-a_5\sin\theta_5 - b_5\cos\theta_5)\hat{i} \\ + \quad & (a_5\cos\theta_5 - b_5\sin\theta_5)\hat{j} \end{aligned} \tag{2.34}$$

$$\frac{d\vec{z}}{d\theta_6} = 0\hat{i} + 0\hat{j} \tag{2.35}$$

Calculating the virtual work, δW_f, due to the applied forces by taking the dot product of the force vector and the virtual displacements gives

$$\begin{aligned}\delta\vec{W}_f = \quad & (-F_x r_2\sin\theta_2 + F_y r_2\cos\theta_2)\delta\theta_2 \\ + \quad & (-F_x r_3\sin\theta_3 + F_y r_3\cos\theta_3)\delta\theta_3 \\ + \quad & (-F_x r_4\sin\theta_4 + F_y r_4\cos\theta_4)\delta\theta_4 \\ + \quad & [F_x(-a_2\sin\theta_5 - b_5\cos\theta_5) \\ & +F_y(a_5\cos\theta_5 - b_5\sin\theta_5)]\delta\theta_5 \\ + \quad & 0\delta\theta_6 \end{aligned} \tag{2.36}$$

Expressing the applied moments in vector form as

$$
\begin{aligned}
T_2 &= -K_2(\theta_2-\theta_{20})\hat{k} && (2.37)\\
T_3 &= -K_3[(\theta_2-\theta_{20})-(\theta_3-\theta_{30})]\hat{k} && (2.38)\\
T_4 &= -K_4[(\theta_3-\theta_{30})-(\theta_4-\theta_{40})]\hat{k} && (2.39)\\
T_5 &= -K_5[(\theta_4-\theta_{40})-(\theta_5-\theta_{50})]\hat{k} && (2.40)\\
T_6 &= -K_6[(\theta_5-\theta_{50})-(\theta_6-\theta_{60})]\hat{k} && (2.41)\\
T_7 &= -K_7[(\theta_8-\theta_{80})-(\theta_7-\theta_{70})]\hat{k} && (2.42)\\
T_8 &= -K_8(\theta_8-\theta_{80})\hat{k} && (2.43)\\
T_9 &= -K_9[(\theta_7-\theta_{70})-(\theta_6-\theta_{60})]\hat{k} && (2.44)\\
M &= M\hat{k} && (2.45)
\end{aligned}
$$

The resulting virtual work due to torques is given by

$$
\begin{aligned}
\delta\vec{W}_m = \quad & (T_2+T_3+T_7\frac{d\theta_8}{d\theta_2}-T_7\frac{d\theta_7}{d\theta_2}\\
& +T_8\frac{d\theta_8}{d\theta_2}+T_9\frac{d\theta_7}{d\theta_2})\delta\theta_2\\
+ \quad & (-T_3+T_4+T_7\frac{d\theta_8}{d\theta_3}-T_7\frac{d\theta_7}{d\theta_3}\\
& +T_8\frac{d\theta_8}{d\theta_3}+T_9\frac{d\theta_7}{d\theta_3})\delta\theta_3\\
+ \quad & (-T_4+T_5+T_7\frac{d\theta_8}{d\theta_4}-T_7\frac{d\theta_7}{d\theta_4}\\
& +T_8\frac{d\theta_8}{d\theta_4}+T_9\frac{d\theta_7}{d\theta_4})\delta\theta_4\\
+ \quad & (M-T_5+T_6+T_7\frac{d\theta_8}{d\theta_5}-T_7\frac{d\theta_7}{d\theta_5}\\
& +T_8\frac{d\theta_8}{d\theta_5}+T_9\frac{d\theta_7}{d\theta_5})\delta\theta_5\\
+ \quad & (-T_6-T_9+T_7\frac{d\theta_8}{d\theta_6}-T_7\frac{d\theta_7}{d\theta_6}\\
& +T_8\frac{d\theta_8}{d\theta_6}+T_9\frac{d\theta_7}{d\theta_6})\delta\theta_6
\end{aligned}
\qquad (2.46)
$$

The total virtual work equations with forces and moments is

$$\delta\vec{W} = \delta\vec{W}_f + \delta\vec{W}_m = 0 \tag{2.47}$$

Additional terms must be defined to complete the derivation. The vector loop equation for the device is

$$\vec{Z} : \vec{r_2} + \vec{r_3} + \vec{r_4} + \vec{r_5} + \vec{r_6} = \vec{r_1} + \vec{r_7} + \vec{r_8} \tag{2.48}$$

The x component of the vector loop is

$$\begin{aligned} \vec{X} \quad : \quad & r_2\cos\theta_2 + r_3\cos\theta_3 + r_4\cos\theta_4 + \\ & cos\theta_5 + r_6\cos\theta_6 \\ & = r_1 + r_7\cos\theta_7 + r_8\cos\theta_8 \end{aligned} \tag{2.49}$$

Differentiating the x components with respect to a generalized coordinate gives

$$\begin{aligned} \frac{d\vec{X}}{dq} \quad : \quad & -r_2\sin\theta_2\frac{d\theta_2}{d\theta_q} - r_3\sin\theta_3\frac{d\theta_3}{d\theta_q} - \\ & r_4\sin\theta_4\frac{d\theta_4}{d\theta_q} - r_5\sin\theta_5\frac{d\theta_5}{d\theta_q} - r_6\sin\theta_6\frac{d\theta_6}{d\theta_q} \\ & = -r_7\sin\theta_7\frac{d\theta_7}{d\theta_q} - r_8\sin\theta_8\frac{d\theta_8}{d\theta_q} \end{aligned} \tag{2.50}$$

The y component of the vector loop is

$$\begin{aligned} \vec{Y} \quad : \quad & r_2\sin\theta_2 + r_3\sin\theta_3 + r_4\sin\theta_4 + \\ & r_5\sin\theta_5 + r_6\sin\theta_6 \\ & = r_7\sin\theta_7 + r_8\sin\theta_8 \end{aligned} \tag{2.51}$$

Differentiating the y components with respect to the generalized coordinate is

$$\begin{aligned}\frac{d\vec{Y}}{dq} \quad : \quad & r_2\cos\theta_2\frac{d\theta_2}{d\theta_q} + r_3\cos\theta_3\frac{d\theta_3}{d\theta_q} + \\ & r_4\cos\theta_4\frac{d\theta_4}{d\theta_q} + r_5\cos\theta_5\frac{d\theta_5}{d\theta_q} + r_6\cos\theta_6\frac{d\theta_6}{d\theta_q} \\ & = r_7\cos\theta_7\frac{d\theta_7}{d\theta_q} + r_8\cos\theta_8\frac{d\theta_8}{d\theta_q}\end{aligned} \tag{2.52}$$

The x and y components differentiated with respect to a single generalized coordinate, θ_i, are

$$\frac{d\vec{X}}{d\theta_i} : -r_i\sin\theta_i = -r_7\sin\theta_7\frac{d\theta_7}{d\theta_i} - r_8\sin\theta_8\frac{d\theta_8}{d\theta_i} \tag{2.53}$$

$$\frac{d\vec{Y}}{d\theta_i} : r_i\cos\theta_i = r_7\cos\theta_7\frac{d\theta_7}{d\theta_i} + r_8\cos\theta_8\frac{d\theta_8}{d\theta_i} \tag{2.54}$$

Solving equation (2.53) for $\frac{d\theta_8}{d\theta_i}$

$$\frac{d\theta_8}{d\theta_i} = \frac{r_i\sin\theta_i - r_7\sin\theta_7\frac{d\theta_7}{d\theta_i}}{r_8\sin\theta_8} \tag{2.55}$$

Solving for $\frac{d\theta_7}{d\theta_i}$ by substituting equation (2.55) into equation (2.54) results in

$$\frac{d\theta_7}{d\theta_i} = \frac{r_i(\cos\theta_i - \sin\theta_i\cot\theta_8)}{r_7(\cos\theta_7 - \sin\theta_7\cot\theta_8)} \tag{2.56}$$

and substituting into equation (2.55) gives $\frac{d\theta_8}{d\theta_i}$ as

$$\frac{d\theta_8}{d\theta_i} = \frac{r_i(\sin\theta_i\cos\theta_7 - \cos\theta_i\sin\theta_7)}{r_8(\sin\theta_7\cos\theta_8 - \sin\theta_8\cos\theta_7)} \tag{2.57}$$

2.4 Model Verification

The flexion-extension virtual work model was verified against finite element analysis results and tests from a physical prototype. The lateral bending virtual work model was verified with a finite element model. The lateral bending model used a torsion beam length, L_{tl}, that is 10 times that listed in the dimension table for the LET joint (Table 2.1). The flexion-extension geom-

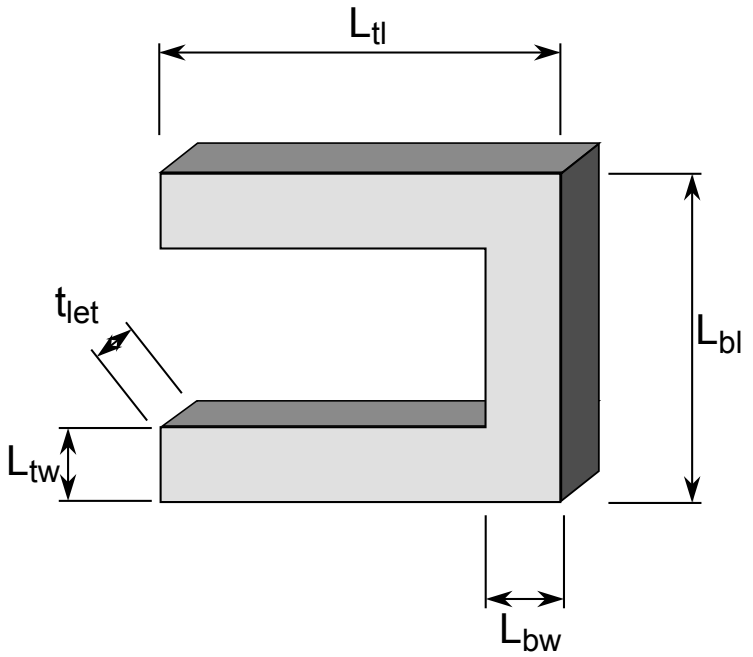

Figure 2.9: LET joint dimensions.

Table 2.1: Model Parameters

LET Joint		Flexion-Extension		Lateral Bending	
t_{let}	0.66 mm	r_1	28.45 mm	r_1	49.24 mm
L_{bw}	6.86 mm	a_2	42.15 mm	r_2	134.51 mm
L_{tw}	7.11 mm	b_2	22.00 mm	r_3	21.34 mm
L_{tl}	16.51 mm			a_5	24.62 mm
L_{bl}	21.34 mm			b_5	10.67 mm
K_Θ	2.17				
γ	0.85				
E	110 GPa				

etry emphasizes the combined torsion (horizontal beams) and bending (vertical beams). While an increase of torsion beam length for the lateral bending model emphasizes the fixed-guided beam motion and ensures beam theory is applicable. The specific device dimensions used for the purpose of model verification are listed in the table and shown in Figure 2.9.

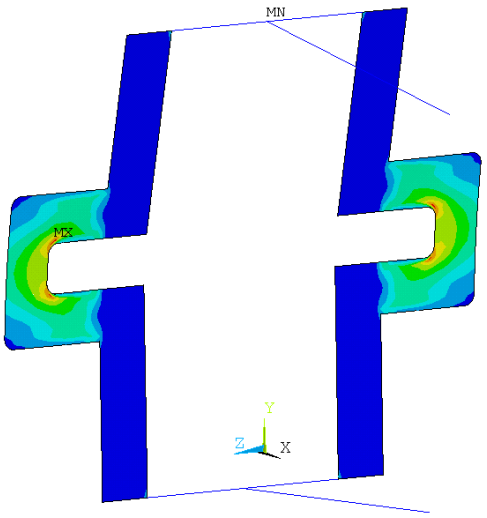

Figure 2.10: von Mises stress distribution.

2.4.1 Finite Element Model

The finite element model was constructed and analyzed in ANSYS. For flexion-extension the boundary conditions are pinned at the force application location and the corresponding bottom pin location. The LET joint where deformation occurs was modeled as a shell element. Relatively stiff beam elements were used as structural beams to simulate the stiffness of the test fixture. The lateral bending FEA used a simplified geometry of just the flexible elements of the LET joint joined at the midline. At this midline the bottom beams were fully constrained while the top beams were deflected vertically from the midline. The von Mises stress distribution for the flexion-extension model can be seen in Figure 2.10.

2.4.2 Physical Prototype Tests

The physical test was constructed in accordance with a modified ASTM standard F1717 test. The fixture consists of an aluminum block to secure a pivot pin which then connects the UHMWPE blocks in which the device had been secured (Fig. 2.11). The device was allowed to rotate around the pivot pin on both the top and the bottom. The device was deflected and the loads were recorded using a tensile test device (Instron- model 1321).

The test results for the flexion-extension model are shown in Fig. 2.12. There is some hysteresis exhibited in the measured data indicating a small amount of friction in the test system.

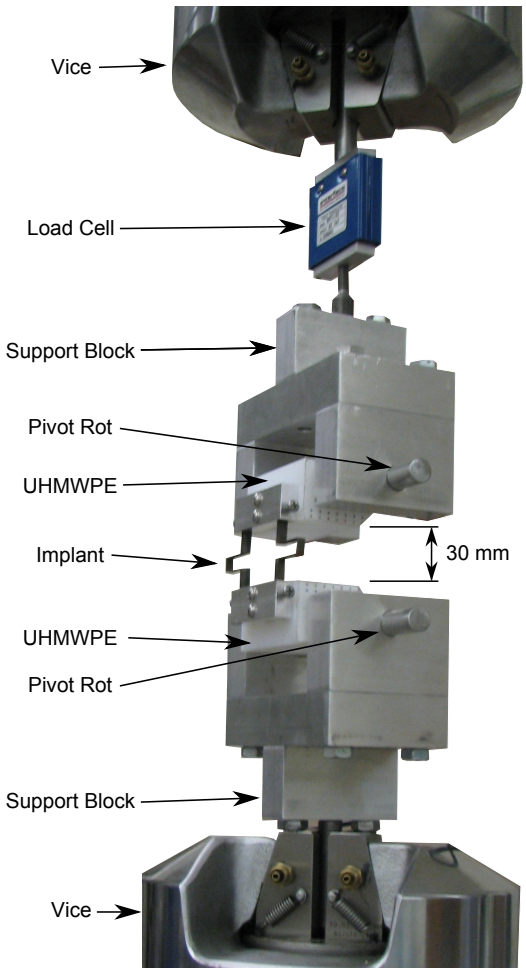

Figure 2.11: Testing fixture (F1717).

Similarly, the results for the lateral bending model are shown in Fig. 2.13. Good correlation is shown between the experimental results, the finite element results, and the closed-form equations. Variation of the closed-form equations can be accounted for by the fact that the radius at the corners was not accounted for in the virtual work model.

2.5 Conclusions

A model of the FlexSuRe™ spinal implant has been derived based on the principle of virtual work. The model has been verified against physical testing and finite element analysis. The model will be useful in designing FlexSuRe™ implants and will facilitate the quick design of devices for given biological parameters.

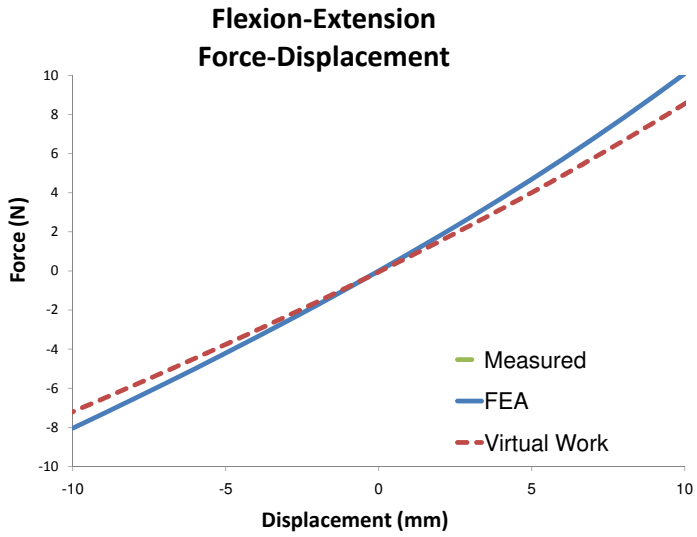

Figure 2.12: Flexion-extension: measured, FEA and virtual work results.

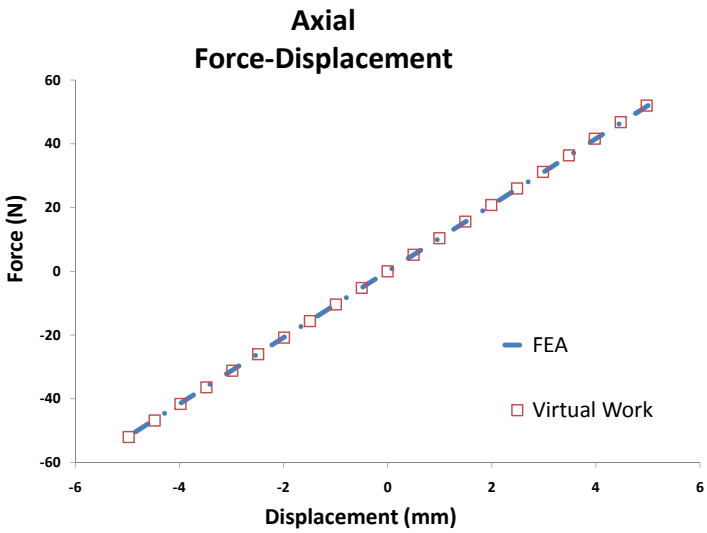

Figure 2.13: Lateral bending: FEA and virtual work results.

CHAPTER 3. COMBINED IMPLANT AND DISC MODEL

The implant is designed to function in conjunction with biologic structures and tissues. The combined implant-tissue system presents a unique force-deflection response. The goal of the implant is to cause this combined force-deflection response to approach that of the normal spine.

The stiffness of the disc and biologic structures is estimated by stiffnesses obtained from literature and integrated into the mathematical analysis. Further data and correlations need to be investigated to describe the stiffness of the biologic structures as they degrade.

A force-deflection model was derived resulting in a set of simultaneous equations of which the solution is the equilibrium position of the device and anatomical structure.

The combined flexion-extension model is depicted in Figure 3.1. The combined lateral bending model is shown in Figure 3.2.

3.1 Disc and Implant in Flexion-Extension

The vector form of the applied forces, $\vec{F}_c$, is

$$\vec{F}_c = X_c\hat{i} + Y_c\hat{j} \tag{3.1}$$

where X_c and Y_c are the applied forces the magnitude of which is correlated with the force-deflection relationship obtained experimentally in the x-direction and y-direction, as shown in Figure 3.1. The location of these forces with respect to the defined origin are

$$\begin{aligned} \vec{z}_c = \quad & (r_1\cos\theta_1 + a_2\cos\theta_2 + c_2\sin\theta_2 - d_2\cos\theta_2\hat{i} \\ + \quad & (r_1\sin\theta_1 + a_2\sin\theta_2 - c_2\cos\theta_2 - d_2\sin\theta_2)\hat{j} \end{aligned} \tag{3.2}$$

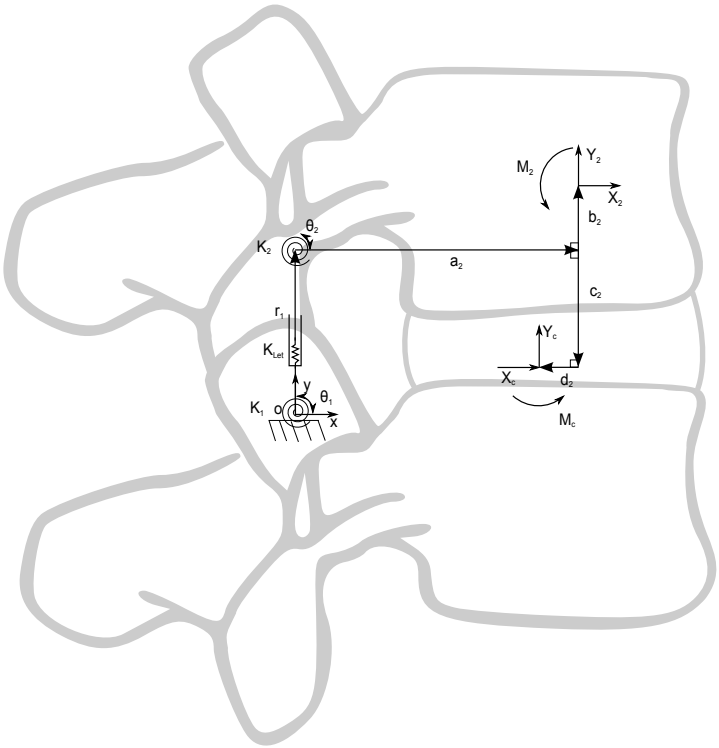

Figure 3.1: Device and disc flexion-extension model with equivalent forces shown.

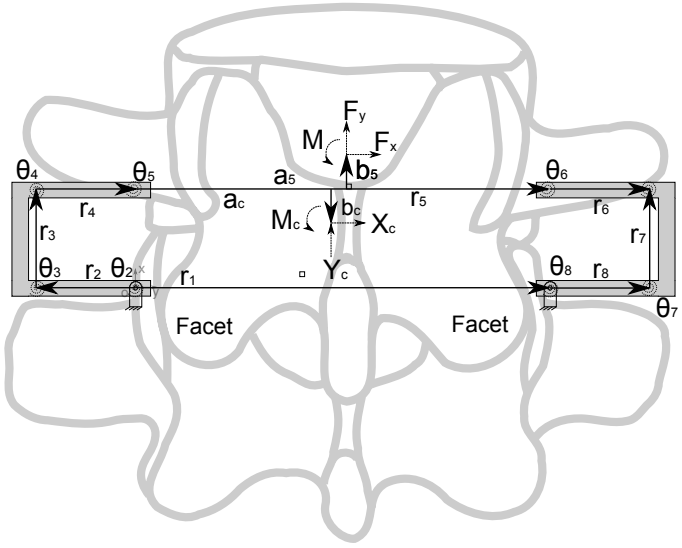

Figure 3.2: Lateral bending model for combined device and disc with equivalent forces shown.

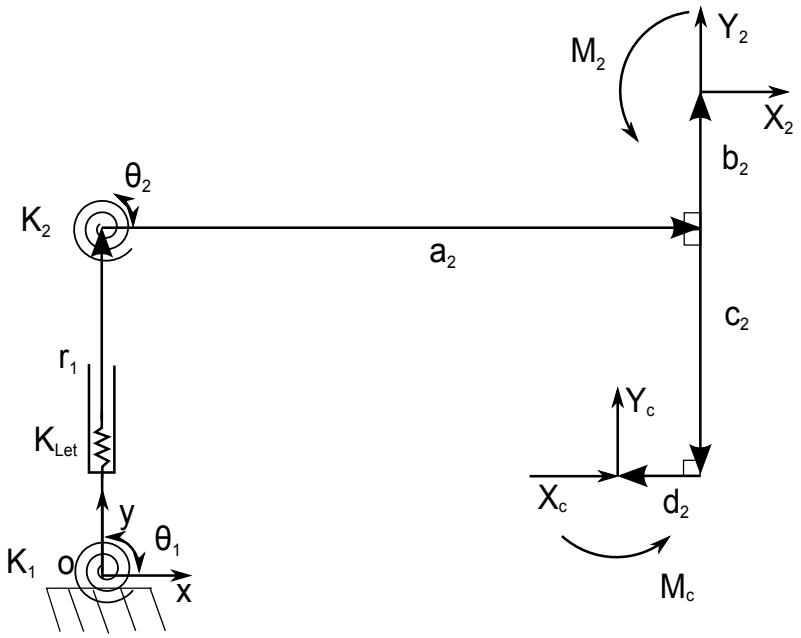

Figure 3.3: Flexion-extension device and disc equivalent forces.

This combined analysis uses the same generalized coordinates as in the virtual work model of the device alone. Differentiating the first position vector with respect to a generalized coordinate gives

$$\begin{aligned}\frac{d\vec{z}_c}{dq_i} = \quad & [\cos\theta_1 \frac{dr_1}{dq_i} - r_1 \sin\theta_1 \frac{d\theta_1}{dq_i} + \\ & (-a_2 \sin\theta_2 + c_2 \cos\theta_2 + d_2 \sin\theta_2)\frac{d\theta_2}{dq_i}]\hat{i} \\ + \quad & [\sin\theta_1 \frac{dr_1}{dq_i} + r_1 \cos\theta_1 \frac{d\theta_1}{dq_i} + \\ & (a_2 \cos\theta_2 + c_2 \sin\theta_2 - d_2 \cos\theta_2)\frac{d\theta_2}{dq_i}]\hat{j} \end{aligned} \tag{3.3}$$

Differentiating the position vector, $\vec{z}_c$, with respect to each of the generalized coordinates results in

$$\frac{d\vec{z}_c}{dr_1} = \cos\theta_1 \hat{i} + \sin\theta_1 \hat{j} \tag{3.4}$$

$$\frac{d\vec{z}_c}{d\theta_1} = -r_1 \sin\theta_1 \hat{i} + r_1 \cos\theta_1 \hat{j} \tag{3.5}$$

$$\frac{d\vec{z}_c}{d\theta_2} = \begin{array}{l}(-a_2\sin\theta_2 + c_2\cos\theta_2 + d_2\sin\theta_2)\,\hat{i} \\ +\ (a_2\cos\theta_2 + c_2\sin\theta_2 - d_2\cos\theta_2)\ \hat{j}\end{array} \tag{3.6}$$

The virtual work due to the applied forces estimates of the disc, δW_{fc}, is calculated by taking the dot product of the force vector and the virtual displacement as

$$\delta W_{fc} = \vec{F}_c \cdot d\vec{z}_c \tag{3.7}$$

$$\begin{aligned}\delta\vec{W}_{fc} = \quad & (X_c\cos\theta_1 + Y_c\sin\theta_1)\delta r_1 \\ + \quad & (-X_c r_1\sin\theta_1 + Y_c r_1\cos\theta_1)\delta\theta_1 \\ + \quad & [X_c(-a_2\sin\theta_2 + c_2\cos\theta_2 + d_2\sin\theta_2) \\ & + Y_c(a_2\cos\theta_2 + c_2\sin\theta_2 - d_2\cos\theta_2)]\delta\theta_2\end{aligned} \tag{3.8}$$

The applied moment estimate of disc torque, M_c, expressed in vector form as

$$M_c = M_c\hat{k} \tag{3.9}$$

The virtual work due to this moments, δW_c, is

$$\delta\vec{W}_c = M_c\delta\theta_2 \tag{3.10}$$

The additional terms to be added to the virtual work equations from the mechanism alone derivation are

$$(X_c\cos\theta_1 + Y_c\sin\theta_1)\delta r_1 = 0 \tag{3.11}$$

$$(-X_c r_1\sin\theta_1 + Y_c r_1\cos\theta_1)\delta\theta_1 = 0 \tag{3.12}$$

$$\begin{aligned}&[X_c(-a_2\sin\theta_2 + c_2\cos\theta_2 + d_2\sin\theta_2) \\ &+ Y_c(a_2\cos\theta_2 + c_2\sin\theta_2 - d_2\cos\theta_2) \\ &+ M_c]\delta\theta_2 = 0\end{aligned} \tag{3.13}$$

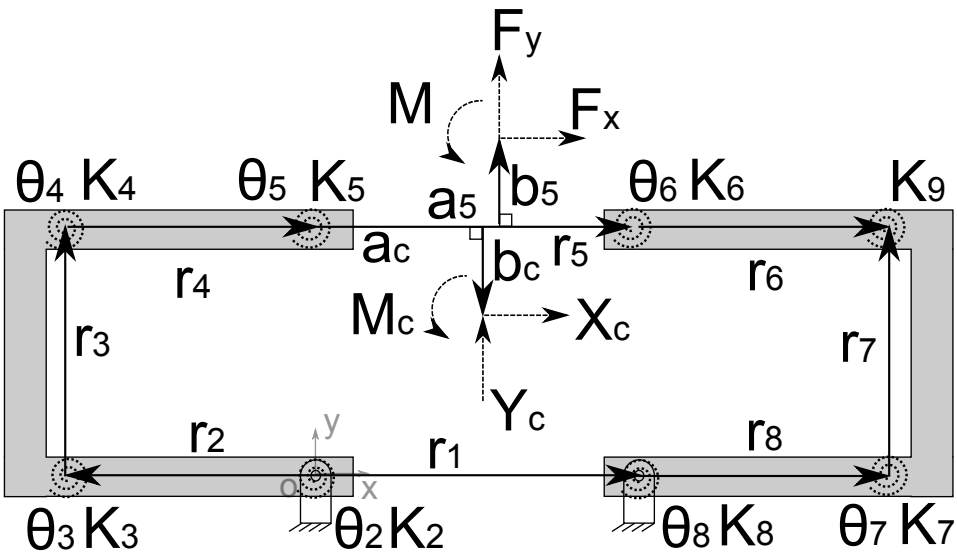

Figure 3.4: Lateral bending device and disc equivalent forces.

The combined system of equations is:

$$(X_2\cos\theta_1 + Y_2\sin\theta_1 + F_s + X_c\cos\theta_1 + Y_c\sin\theta_1)\delta r_1 = 0 \tag{3.14}$$

$$(-X_2 r_1\sin\theta_1 + Y_2 r_1\cos\theta_1 + T_1 - T_2 - X_c r_1\sin\theta_1 + Y_c r_1\cos\theta_1)\delta\theta_1 = 0 \tag{3.15}$$

$$\begin{aligned}&[X_2(-a_2\sin\theta_2 - b_2\cos\theta_2) + X_c(-a_2\sin\theta_2 + c_2\cos\theta_2 + d_2\sin\theta_2)\\ &+ Y_2(a_2\cos\theta_2 - b_2\sin\theta_2) + Y_c(a_2\cos\theta_2 + c_2\sin\theta_2 - d_2\cos\theta_2)\\ &+ T_2 + M_2 + M_c]\delta\theta_2 = 0\end{aligned} \tag{3.16}$$

Solving these equations simultaneously results in the estimated equilibrium position of the device in combination with the anatomical structure.

3.2 Lateral Bending with Disc Derivation

The pseudo-rigid-body model for lateral bending with applied forces and torque locations is shown in Figure 3.4. The vector form of the applied forces is

$$\vec{F}_c = X_c\hat{i} + Y_c\hat{j} \tag{3.17}$$

The location of this force with respect to the defined origin is

$$\begin{aligned}\vec{z}_c = \quad & (r_2\cos\theta_2 + r_3\cos\theta_3 + r_4\cos\theta_4 \\ & + a_c\cos\theta_5 + b_c\sin\theta_5)\hat{i} \\ + \quad & (r_3\sin\theta_3 + r_4\sin\theta_4 \\ & + a_c\sin\theta_5 - b_c\cos\theta_5)\hat{j}\end{aligned} \tag{3.18}$$

Differentiating the position vector, $\vec{z}_c$, with respect to a generalized coordinate gives

$$\begin{aligned}\frac{d\vec{z}_c}{dq_i} = \quad & [-r_2\sin\theta_2\frac{d\theta_2}{dq_i} - r_3\sin\theta_3\frac{d\theta_3}{dq_i} \\ & - r_4\sin\theta_4\frac{d\theta_4}{dq_i} + (-a_c\sin\theta_5 + b_c\cos\theta_5)\frac{d\theta_5}{dq_i}]\hat{i} \\ + \quad & [r_2\cos\theta_2\frac{d\theta_2}{dq_i} + r_3\cos\theta_3\frac{d\theta_3}{dq_i} \\ & + r_4\cos\theta_4\frac{d\theta_4}{dq_i} + (a_c\cos\theta_5 + b_c\sin\theta_5)\frac{d\theta_5}{dq_i}]\hat{j}\end{aligned} \tag{3.19}$$

The same generalized coordinates as the previous lateral bending derivation are used in this combined derivation. Differentiating the position vector with respect to each specific q_i results in

$$\frac{d\vec{z}_c}{d\theta_2} = -r_2\sin\theta_2\hat{i} + r_2\cos\theta_2\hat{j} \tag{3.20}$$

$$\frac{d\vec{z}_c}{d\theta_3} = -r_3\sin\theta_3\hat{i} + r_3\cos\theta_3\hat{j} \tag{3.21}$$

$$\frac{d\vec{z}_c}{d\theta_4} = -r_4\sin\theta_4\hat{i} + r_4\cos\theta_4\hat{j} \tag{3.22}$$

$$\frac{d\vec{z}_c}{d\theta_5} = (-a_c \sin\theta_5 + b_c \cos\theta_5)\hat{i} + (a_c \cos\theta_5 + b_c \sin\theta_5)\hat{j} \tag{3.23}$$

$$\frac{d\vec{z}_c}{d\theta_6} = 0\hat{i} + 0\hat{j} \tag{3.24}$$

Calculating the virtual work, δW_{fc}, due to the applied forces by taking the dot product of the force vector and the virtual displacements gives

$$\delta W_{fc} = \vec{F}_c \cdot d\vec{z}_c \tag{3.25}$$

$$\begin{aligned}\delta\vec{W}_{fc} = & (-X_c r_2 \sin\theta_2 + Y_c r_2 \cos\theta_2)\delta\theta_2 \\ + & (-X_c r_3 \sin\theta_3 + Y_c r_3 \cos\theta_3)\delta\theta_3 \\ + & (-X_c r_4 \sin\theta_4 + Y_c r_4 \cos\theta_4)\delta\theta_4 \\ + & [X_c(-a_c \sin\theta_5 + b_c \cos\theta_5) \\ & + Y_c(a_c \cos\theta_5 + b_c \sin\theta_5)]\delta\theta_5 \\ + & 0\delta\theta_6\end{aligned} \tag{3.26}$$

Expressing the applied moments in vector form as

$$M_c = M_c\hat{k} \tag{3.27}$$

The resulting virtual work due to M_c is

$$\delta\vec{W}_{mc} = M_c\delta\theta_5 \tag{3.28}$$

The total virtual work equations with forces and moments with the estimate of the force-displacement relationships of the anatomical structure is

$$\delta\vec{W} = \delta\vec{W}_f + \delta\vec{W}_m + \delta\vec{W}_{fc} + \delta\vec{W}_{mc} = 0 \tag{3.29}$$

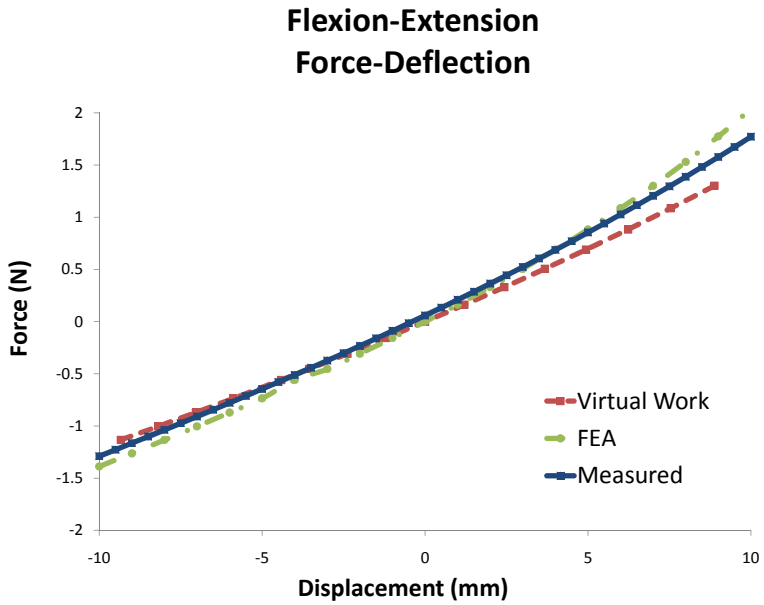

Figure 3.5: Pinned-pinned force-deflection.

3.3 New Geometry Validation

The model was validated by using the finite element method and physical testing. The close correlation between these three methods is shown in Figure 3.5 for the pinned-pinned test. The results from the three methods for the compression/extension test are shown in Figure 3.6. The new geometry of the FlexSuRe™ is in Table 3.3.

3.4 Cadaveric Testing

To evaluate the model using the combined device and biologic tissue model, the device was implanted into a single FSU and tested. First, the FSU was exercised in the flexion-extension and lateral bending loading conditions using a custom spinal testing machine. Second, the FlexSure™ was fixed to the vertebral bodies using pedicle screws and again exercised in flexion-extension and lateral bending. A follower load was used for an approximation of physiologic loading [14]. Pictures of the device implanted on the FSU were taken with a calibration ruler to

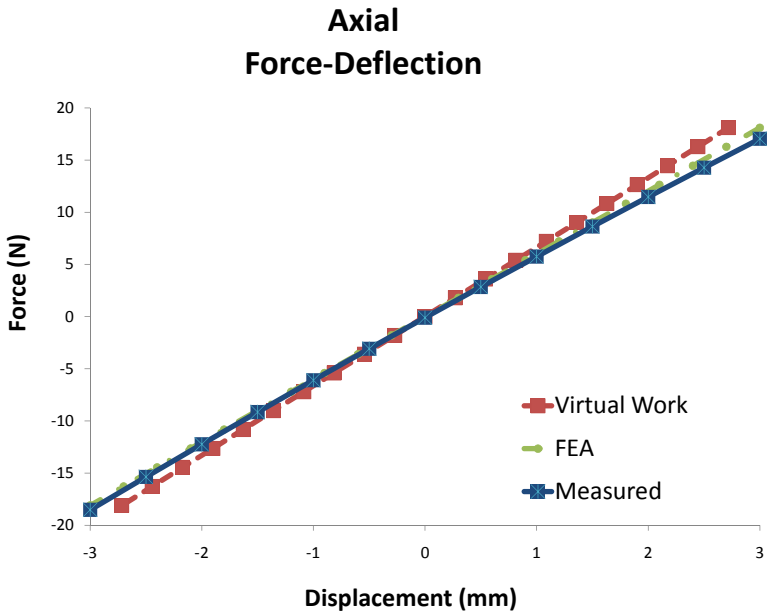

Figure 3.6: Axial force-deflection.

Table 3.1: Combined Model Parameters

FlexSuRe™		Flexion-Extension		Lateral Bending	
t_{let}	4 mm	r_1	8.5 mm	r_1	61.6 mm
L_{bw}	1.5 mm	a_2	31.3 mm	r_2	17.4 mm
L_{tw}	0.5 mm	b_2	0 mm	r_3	8.5 mm
L_{tl}	22 mm	c_2	18.93 mm	a_5	27.65 mm
L_{bl}	8.5 mm	d_2	0 mm	b_5	0 mm
K_Θ	2.17			a_{disc}	27.65 mm
γ	0.85			b_{disc}	9.47 mm
E	110 GPa				

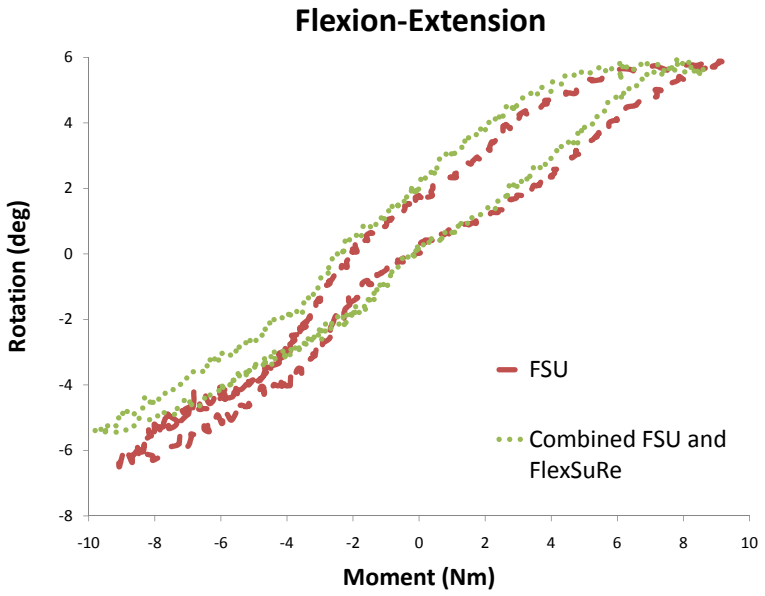

Figure 3.7: Flexion-extension cadaveric testing results.

extract relevant dimensions for numerical validation. These dimensions relevant to the validation are listed in Table 3.3.

3.4.1 Cadaveric Results and Model Comparison

The force-deflection relationships for the combined biologic structure and the FlexSuRe™ implant are shown in Figure 3.7 for flexion-extension and in Figure 3.8 for lateral bending. These curves were then divided into flexion and extension components and linearized with a 3rd degree polynomial. These polynomial lines were used as inputs to the virtual work model as the stiffness of the biologic structures. For the physical testing, a follower load of 444.8 N (100 lbs) was used to compress the FSU. Other inputs to the virtual work model include component shear and compression stiffness values listed in, Table 3.2 [1]. The results of the predicted and the measured combined systems are shown in Figure 3.9 for flexion-extension and Figure 3.10 for lateral bending.

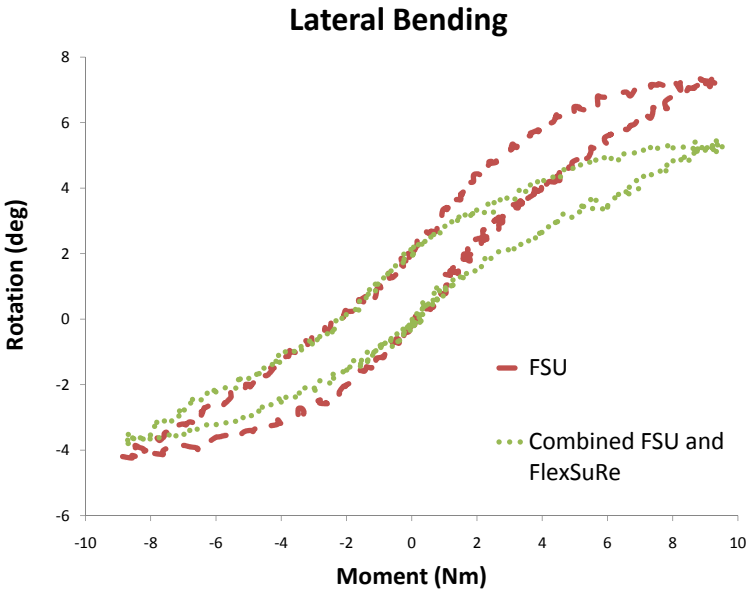

Figure 3.8: Lateral bending cadaveric testing results.

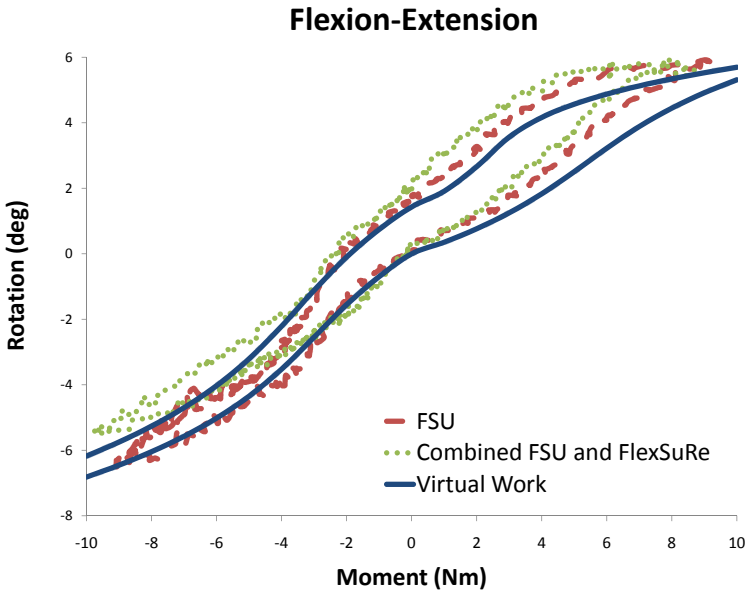

Figure 3.9: Flexion-extension cadaveric testing results and virtual work estimate.

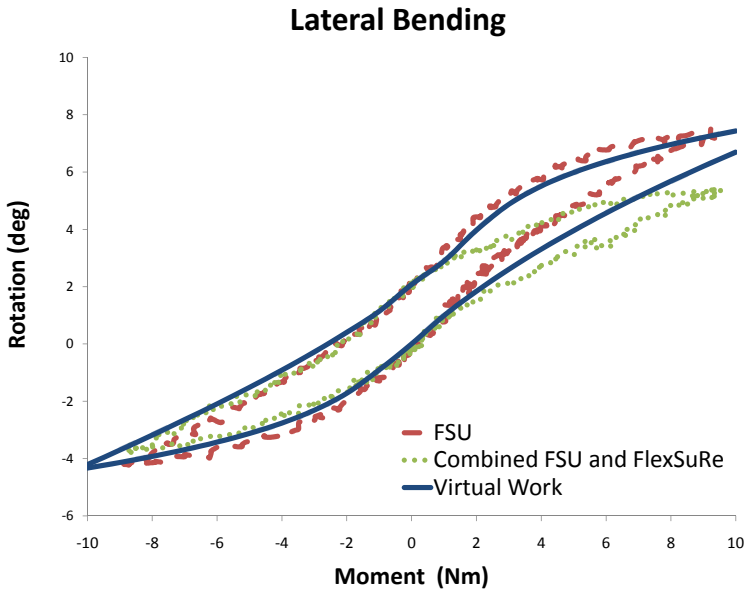

Figure 3.10: Lateral bending cadaveric testing results and virtual work estimate.

Table 3.2: Disc Estimate Parameters [1]

Disc Estimate	
Compression	1430 N/mm
Posterior Shear	200 N/mm
Anterior Shear	330 N/mm
Lateral Shear	278 N/mm

3.5 Results Discussion

Further work is needed to provide a more definitive validation, but at this point the virtual work model is consistent with the physical test results. The disc used was four times stiffer than the disc the disc the prototype was designed for. This resulted in the prototype having only a small effect on the overall stiffness. The model predicted the change for the given disc stiffness.

CHAPTER 4. STRESS ANALYSIS

A conservative approach to estimating the stress in the spinal implant consists of calculating the stress at a theoretical location where all the stresses combine in their highest von Mises stress. There are four components of stress: shear stress on the horizontal member due to torsion of the beam or bending of the device τ_{bh}; stress in the vertical member due to bending of the device σ_{bv}; stress in the horizontal member due to axial extension or compression of the device, σ_{eh}; stress in the vertical member due to axial extension or compression of the device, σ_{ev}.

4.1 Torsion of the Horizontal Member

The maximum stress due to torsion of the horizontal member is at the center of the long side of the cross section as shown in Figure 4.1. The location could vary due to changing sizes of the members during geometrical changes. Both locations are included until final dimensions are found. Since the LET joint is symmetric, then T_i is equal to half the total torque, T_{tot}, given by

$$T_i = \frac{T_{tot}}{2} \tag{4.1}$$

Given that w is greater than or equal to t the following parameter is needed [18]

$$Q = \frac{w^2 t^2}{3w + 1.8t} \tag{4.2}$$

Using the previous equations, the shear stress due to bending of the horizontal member is given by:

$$\tau_{bh} = \frac{T_i}{Q} \tag{4.3}$$

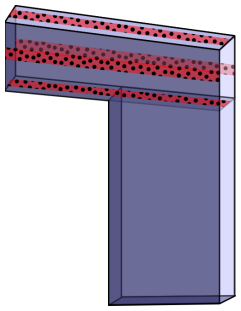

Figure 4.1: Locations of maximum stress for bending of horizontal member.

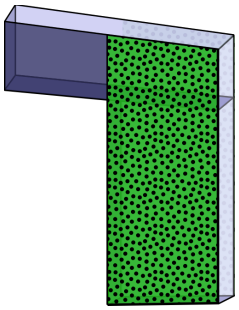

Figure 4.2: Locations of maximum stress for bending of vertical member.

4.2 Bending of the Vertical Member

The maximum stress is on the surfaces of the vertical member in the direction of the bending as shown in Figure 4.2. The stress on these surfaces due to bending of the vertical member is given by:

$$\sigma_{bv} = \frac{T_i c}{I} \tag{4.4}$$

where c is half the height of the beam and I is the second moment of inertia.

4.3 Extension-Compression of the Vertical Member

The maximum stress due to extension or compression is the same throughout the cross section of the member. The stress in the vertical member due to extension/compression, σ_{ev}, is given by:

$$\sigma_{ev} = \frac{F}{2A} \tag{4.5}$$

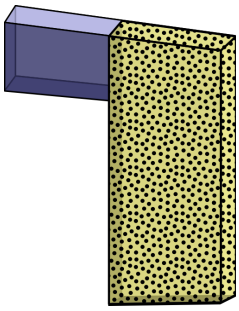

Figure 4.3: Locations of maximum stress for extension-compression of vertical member.

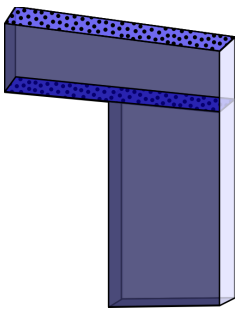

Figure 4.4: Locations of maximum stress for extension-compression of horizontal member.

where F is the applied force on the whole device and A is the cross sectional area of the vertical beam.

4.4 Bending of the Horizontal Member

The maximum stress due to bending is on the top and bottom surfaces of the horizontal member. Under this loading condition, this member is a fixed-guided beam. The following equation is the stress in the horizontal member due to bending:

$$\sigma_{eh} = \frac{FL_{tl}c}{4I} \tag{4.6}$$

The maximum displacement is given by

$$d_{max} = 2\gamma L_{tl} \sin\Theta \tag{4.7}$$

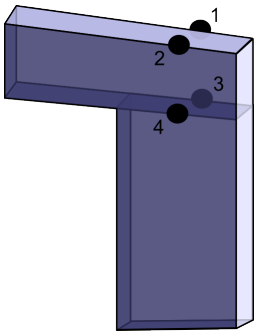

Figure 4.5: Locations of maximum stress.

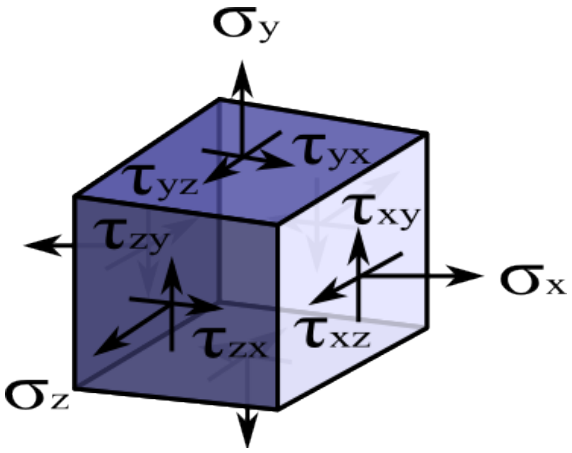

Figure 4.6: Stress element.

where Θ is found by solving

$$\cos\Theta\gamma L_{tl}F = 4k_{fg}\Theta \tag{4.8}$$

4.5 Combined Stress

In the pursuit of simplicity, a theoretical location where all the stresses are maximum is analyzed. Representative of these combinations are the four locations in Figure 4.5. These provide locations to visualize the stress combinations if the shear stress were maximum at these locations and the σ_{eh} was maximum also. A general stress element is depicted in Figure 4.6. The set of possible combinations of stress are shown in Figure 4.7.

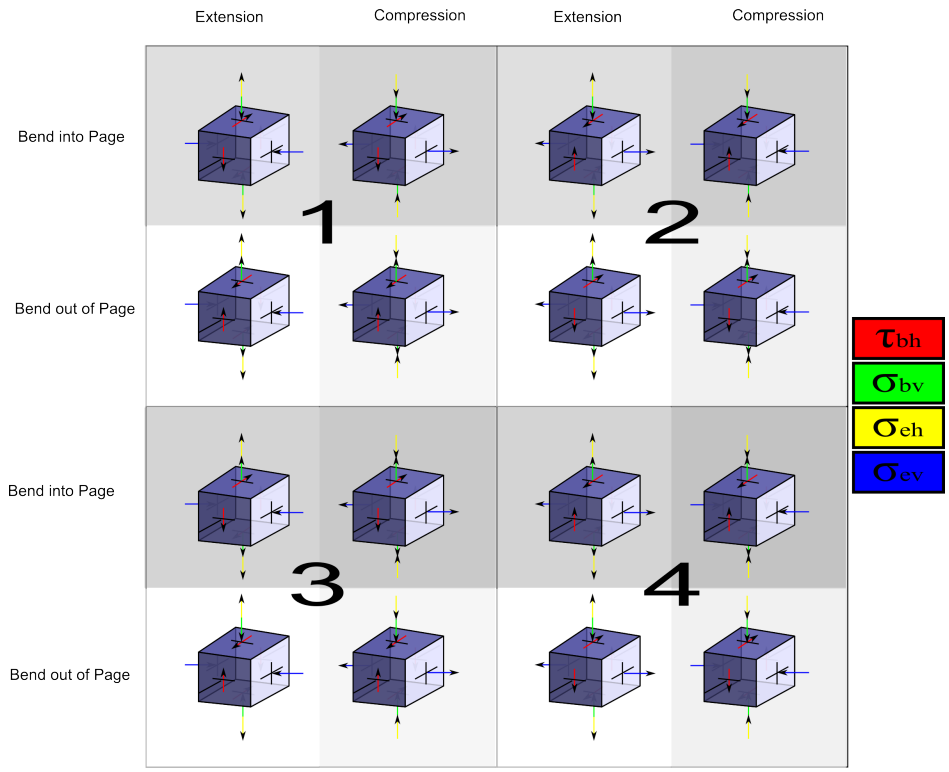

Figure 4.7: Possible stress combinations.

4.6 Mathematical Calculations

The general stress tensor is given by:

$$\begin{bmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix} \tag{4.9}$$

However, not all components are active in this analysis. The stress tensor with active components of stress is

$$\begin{bmatrix} \sigma_{eh} & 0 & 0 \\ 0 & \sigma_{bv} + \sigma_{ev} & \tau_{bh} \\ 0 & \tau_{bh} & 0 \end{bmatrix} \tag{4.10}$$

To calculate the possible combinations of stress an array of possible combinations, positive and negative, is multiplied by the magnitude of stress calculated from the stress equations given previously. The principle stresses the eigenvalues of the stress tensor. The von Mises stress, also known

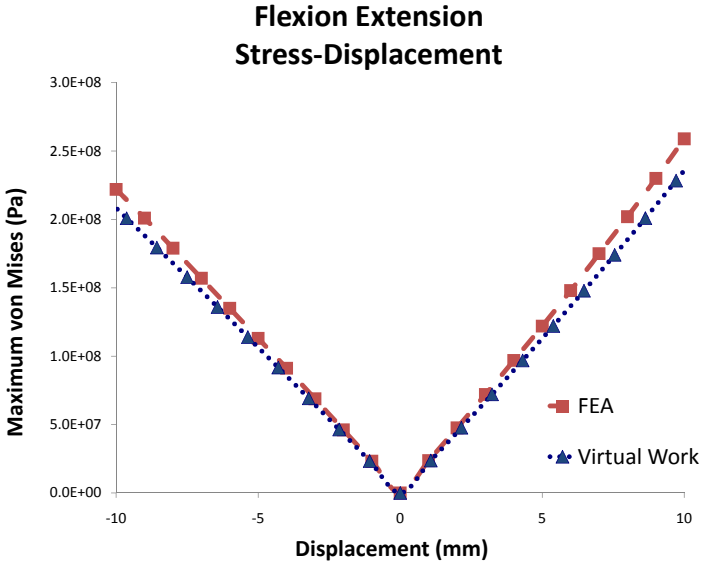

Figure 4.8: Flexion-extension stress-displacement comparison.

as Max Distortion Energy Criterion, is given by

$$\sigma_v = \sqrt{\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2}} \tag{4.11}$$

where σ_1, σ_2 and σ_3 are the principle stresses.

4.7 Model Comparison

A comparison of the FEA stress results and the stress estimates of the derived stress equations were performed. The stress-displacement relationship for the flexion-extension loading configuration is shown in Figure 4.8. Similarly, the relationship for axial loading condition is shown in Figure 4.9.

Good correlation is shown in the two loading conditions. One source of error for the flexion-extension relationship arises from fillets included in the FEA model and not accounted for in the derived estimation of the stress. Other sources error include the purely fixed-guided beam

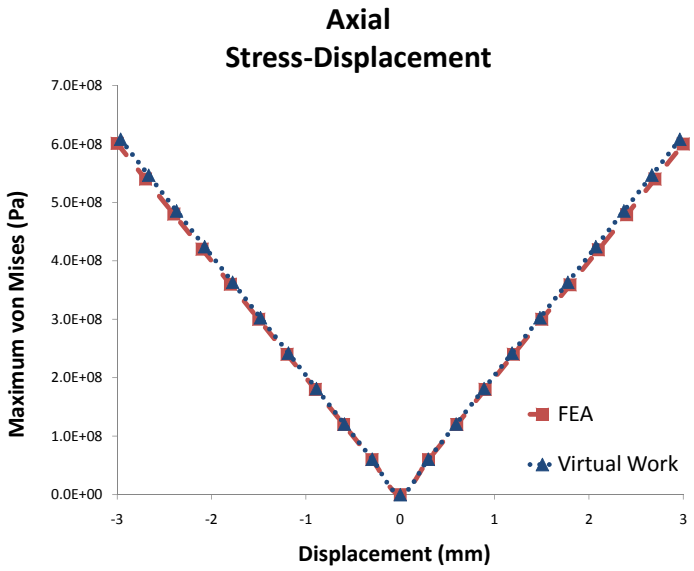

Figure 4.9: Axial stress-displacement comparison.

estimate in the derived equations. The stress equations provide a quick approach for estimating the stress in the spinal implant.

CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

This thesis has presented a compliant-mechanisms-based spinal implant that shows promise for achieving many objectives of spinal implant design which others have not been able to attain. The proposed implant is fully compliant and thus has the potential to eliminate wear due to articulating surfaces, enhance spinal stability by returning the spine to normal motion patterns, and integrate lifting or distraction forces and moments to reduce nerve root impingement.

Presented herein is also the mathematical model describing the motion of the device alone and in conjunction with an estimate of anatomical structure stiffness. Normally this combined stiffness is a measured quantity and not easily predicted. The model has been confirmed by close correlation between finite element methods, physical device testing and cadaveric testing. This three degree-of-freedom kinematic model, in combination with biologic structure and motion restoration parameters, will be a valuable tool in implant design and clinical applications for pain reduction. The work here introduces and provides a foundation for development of the FlexSuRe™ spinal implant.

5.2 Recommendations

Further work is needed to fully define the behavior change that would be most advantageous to reduce pain. Much discussion relates to whether or not a non-degenerate kinematic signature should be restored or if simply making it stiffer or more stable is adequate. Also, an estimate of the stiffness that is lost due to early stage degeneration needs to be obtained and correlated to measurements that can be obtained in vivo. Other factors must be defined, such as how much distraction load should the FSU experience to facilitate pain reduction and possible healing.

The model assumes that the device is planar and not implanted at an angle which protrudes posteriorly. During implantation it may occur or be advantageous to implant the device at a slight posterior angle. The implications of this angle change have not been explored. The bone/screw interface should be evaluated, particularly to understand how much torque the pedicle screws experience and the ability of the bone/screw interface to withstand such torque. An possible extension of this device is an embodiment with the ability to provide a multilevel effect or a design that facilitates a "daisy chain" of devices. Further cadaveric testing is needed, as is wear particle testing to confirm that there are no significant wear particles produced.

REFERENCES

[1] Frei, H., Oxland, T. R., Rathonyi, G. C., and Nolte, L., 2001. "The effect of nucleotomy on lumbar spine mechanics in compression and shear loading.." *Spine October 1, 2001,* **26**(19).

[2] Mulholland, R., and Sengupta, D., 2002. "Rationale, principles and experimental evaluation of the concept of soft stabilization." *European Spine Journal,* **11**(0), October, pp. S198–S205.

[3] Sengupta, 2005. Dynamic stabilization devices in the treatment of low back pain, Oct.

[4] Lehmann, T. R., Spratt, K. F., Tozzi, J. E., Weinstein, J. N., Reinarz, S. J., el-Khoury, G. Y., and Colby, H., 1987. "Long-term follow-up of lower lumbar fusion patients." *Spine,* **12**(2), Mar., pp. 97–104 PMID: 2954220.

[5] Hilibrand, A. S., and Robbins, M., 2004. "Adjacent segment degeneration and adjacent segment disease: the consequences of spinal fusion?." *The Spine Journal: Official Journal of the North American Spine Society,* **4**(6 Suppl), Dec., pp. 190S–194S PMID: 15541666.

[6] Schnake, K., Putzier, M., Haas, N., and Kandziora, F., 2006. "Mechanical concepts for disc regeneration." *European Spine Journal,* **15**(0), pp. 354–360.

[7] Ong, K., Lau, E., Ianuzzi, A., Villarraga, M., and Kurtz, S. M., 2008. Future demand in spinal fusions: U.s. projections to 2030 75th Annual Meeting of the American Associations of Orthopaedic Surgeons, March 5-9.

[8] Anderson, P. A., Rouleau, J. P., Bryan, V. E., and Carlson, C. S., 2003. "Wear analysis of the bryan cervical disc prosthesis." *Spine,* **28**(20), Oct., pp. S186–94 PMID: 14560190.

[9] McKee, J., 2009. "Engineering the future of implants." *AAOS Now*(20), March.

[10] O'Leary, P., Nicolakis, M., Lorenz, M. A., Voronov, L. I., Zindrick, M. R., Ghanayem, A., Havey, R. M., Carandang, G., Sartori, M., Gaitanis, I. N., Fronczak, S., and Patwardhan, A. G. "Response of charit total disc replacement under physiologic loads: prosthesis component motion patterns." *The Spine Journal: Official Journal of the North American Spine Society,* **5**(6), pp. 590–9.

[11] Upasani, V. V., Farnsworth, C. L., Tomlinson, T., Chambers, R. C., Tsutsui, S., Slivka, M. A., Mahar, A. T., and Newton, P. O., 2009. "Pedicle screw surface coatings improve fixation in nonfusion spinal constructs." *Spine,* **34**(4), pp. 335–343.

[12] Howell, L. L., 2001. *Compliant Mechanisms*. Wiley-IEEE.

[13] Hallab, N. J., Cunningham, B. W. M., and Jacobs, J. J., 2003. "Spinal implant Debris-Induced osteolysis.." *Spine October 15, 2003*.

[14] Patwardhan, A. G., Havey, R. M. B., Meade, K. P., Lee, B. B., and Dunlap, B. B., 1999. "A follower load increases the Load-Carrying capacity of the lumbar spine in compression. [Miscellaneous article]." *Spine May 15, 1999,* **24**(10).

[15] Panjabi, M., Oxland, T., Yamamoto, I., and Crisco, J., 1994. "Mechanical behavior of the human lumbar and lumbosacral spine as shown by three-dimensional load-displacement curves." *J Bone Joint Surg Am,* **76**(3), Mar., pp. 413–424.

[16] Schmidt, T. A., An, H. S., Lim, T. H., Nowicki, B. H., and Haughton, V. M., 1998. "The stiffness of lumbar spinal motion segments with a high-intensity zone in the anulus fibrosus." *Spine,* **23**(20), Oct., pp. 2167–2173 PMID: 9802156.

[17] Bergmark, A., 1989. "Stability of the lumbar spine – a study in mechanical engineering." *Acta Orthopaedica Scandinavica,* **60**(3 supp 230), p. 1.

[18] Jacobsen, J. O., Chen, G., Howell, L. L., and Magleby, S. P., 2009. "Lamina emergent torsional (LET) joint." *Mechanism and Machine Theory,* **44**(11), Nov., pp. 2098–2109.

APPENDIX A. FLEXION-EXTENSION MATLAB CODE

This appendix is the Matlab code for the virtual work calculation and optimization to find the equilibrium position of the FlexSuRe™ with and without the biologic estimate for flexion-extension. This code consists of several functions which are separated by sections.

A.1 Main.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Main driver of the force-deflection program
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [] = Main()
clear
clc
%Input the Dimension parameters
[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,a_2,b_2,c_2,d_2]...
    =Dimensions_0;

%Plot the cross section of the half LET
CrossSection();

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%guesses for the varaibles to be solved for
%start with the initial values
gr_1=r_10;          %m
gtheta_1=theta_10;  %rad
gtheta_2=theta_20;  %rad
```

```
%Initial guesses in a vector
x0=[gr_1,gtheta_1,gtheta_2];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%The initial positions (X,Y) of the torsion springs, force app loc, and
%other features
X1=gr_1*cos(gtheta_1);
Y1=gr_1*sin(gtheta_1);

X2=gr_1*cos(gtheta_1)+a_2*cos(gtheta_2);
Y2=gr_1*sin(gtheta_1)+a_2*sin(gtheta_2);

X3=gr_1*cos(gtheta_1)+a_2*cos(gtheta_2)-b_2*sin(gtheta_2);
Y3=gr_1*sin(gtheta_1)+a_2*sin(gtheta_2)+b_2*cos(gtheta_2);

X4=gr_1*cos(gtheta_1)+a_2*cos(gtheta_2)+c_2*sin(gtheta_2);
Y4=gr_1*sin(gtheta_1)+a_2*sin(gtheta_2)-c_2*cos(gtheta_2);

X5=gr_1*cos(gtheta_1)+a_2*cos(gtheta_2)+c_2*sin(gtheta_2)...
    -d_2*cos(gtheta_2);
Y5=gr_1*sin(gtheta_1)+a_2*sin(gtheta_2)-c_2*cos(gtheta_2)...
    -d_2*sin(gtheta_2);

PX=[0,X1,X2,X3,X4,X5];
PY=[0,Y1,Y2,Y3,Y4,Y5];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%Used for find the change in length of the orgin to the
%force app location   %used with another piece after deflected positions
%are calculated
X_disp_0=X3-a_2;
Y_disp_0=Y3+b_2;
r_disp_0=sqrt(X_disp_0^2+Y_disp_0^2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Loop for solving system of equations at various loadings
%Loadsteps=100;
%MaxForce=-1; %Newtons
%MaxTorque=-10:1:10; %Newton-meters
%for i=1:Loadsteps
%for i=1:numel(MaxTorque)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%M_2=[-10:1:10];
M_2=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
X_2=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
Y_2=[-1.3654,-1.2412,-1.1152,-0.98704,-0.85637,-0.72293,-0.58638,...
    -0.44621,-0.30209,-0.15353,0,0.1591,0.32444,0.49679,0.67701,...
    0.86611,1.0653,1.2757,1.4991,1.7374,1.9927];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:numel(M_2)
    %M_2(i)=i*MaxForce/Loadsteps; % Newtons
    %M_2(i)=MaxTorque(i); % Newton meter

    options = optimset('Algorithm','active-set','Display','off');
```

```
x=fsolve(@(x) Virtual(x,X_2(i),Y_2(i),M_2(i)),x0);

r_1(i)=x(1);
theta_1(i)=x(2);
theta_2(i)=x(3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%The initial positions (X,Y) of the torsion springs,
%force app loc, and
%other features
DefX1(i)=r_1(i)*cos(theta_1(i));
DefY1(i)=r_1(i)*sin(theta_1(i));

DefX2(i)=r_1(i)*cos(theta_1(i))+a_2*cos(theta_2(i));
DefY2(i)=r_1(i)*sin(theta_1(i))+a_2*sin(theta_2(i));

DefX3(i)=r_1(i)*cos(theta_1(i))+a_2*cos(theta_2(i))...
    -b_2*sin(theta_2(i));
DefY3(i)=r_1(i)*sin(theta_1(i))+a_2*sin(theta_2(i))...
    +b_2*cos(theta_2(i));

DefX4(i)=r_1(i)*cos(theta_1(i))+a_2*cos(theta_2(i))...
    +c_2*sin(theta_2(i));
DefY4(i)=r_1(i)*sin(theta_1(i))+a_2*sin(theta_2(i))...
    -c_2*cos(theta_2(i));

DefX5(i)=r_1(i)*cos(theta_1(i))+a_2*cos(theta_2(i))...
    +c_2*sin(theta_2(i))-d_2*cos(theta_2(i));
DefY5(i)=r_1(i)*sin(theta_1(i))+a_2*sin(theta_2(i))...
    -c_2*cos(theta_2(i))-d_2*sin(theta_2(i));
```

```
PointsX=[0,DefX1(i),DefX2(i),DefX3(i),DefX4(i),DefX5(i)];
PointsY=[0,DefY1(i),DefY2(i),DefY3(i),DefY4(i),DefY5(i)];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%The vector from the origin to
X_disp(i)=DefX3(i)-a_2;
Y_disp(i)=DefY3(i)+b_2;
r_disp(i)=sqrt(X_disp(i)^2+Y_disp(i)^2);
r_3_disp(i)=r_disp(i)-r_disp_0;
r_3_disp_mm=(r_3_disp.*1000)';
DeflectedY3=((DefY3-Y3)*1000)';

%Center of rotation location function
[interX,interY] = CenterRotation(r_1,theta_1,theta_2,i);
CorX(i)=interX;
CorY(i)=interY;

%Animation (pseudo) of the deflection of the
%construct as it is loaded
figure(1)
subplot(2,2,2)
    plot(PX*100,PY*100,PointsX*100,PointsY*100,'-o',...
        CorX*100,CorY*100,'*')
        %AXIS ([-1 5 -1 5])
        AXIS equal
        xlabel('cm')
        ylabel('cm')
        grid on
```

```
            legend('Original','Deformed','IAR','Location','NorthWest')
            %AXIS([XMIN XMAX YMIN YMAX])

    %Stress calculation in StressCalc.m
    [max_stress(i),SF_yield(i),SF_endurance(i)] =...
        StressCalc(r_1(i),theta_2(i));

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Find the degenerate torque of disc alone in DiscTorque.m
theta=[-10:1:10];
theta=theta.*(pi/180);
for j=1:numel(theta)
    torque(j)=DiscTorque(theta(j));
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This takes away the inital center of rotation
%point which is at the origin
%and sets it equal to the second point
CorX(1)=CorX(2);
CorY(1)=CorY(2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Intact and Degenerate cases
%Data in FSU.m file
```

```
[NaturalFSU] = FSU();
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%PLOTS

%Torque Rotation Curves
%subplot(2,2,1)
%    plot(NaturalFSU(:,1),-NaturalFSU(:,2).*(180/pi),'-',...
%        torque,-theta.*(180/pi),'-o',...
%        -M_2,-(theta_2-gtheta_2).*(180/pi),'-*')
%        ylabel('Rotation (deg)')
%        xlabel('Torque (Newton-meter)')
%        AXIS([-10 10 -10 10])
%        grid on
%        legend('Natural','Degenerate','Assembly','Location','SouthEast')

subplot(2,2,1)
    plot(NaturalFSU(:,1),-NaturalFSU(:,2).*(180/pi),'-',...
        -M_2,-(theta_2-gtheta_2).*(180/pi),'-*')
        ylabel('Rotation (deg)')
        xlabel('Torque (Newton-meter)')
        AXIS([-10 10 -10 10])
        grid on
        legend('Natural','Assembly','Location','SouthEast')



%Side view locations
subplot(2,2,2)
```

```
    plot(PX*1000,PY*1000,PointsX*1000,PointsY*1000,...
        '-o',CorX*1000,CorY*1000,'*')
        %AXIS ([-10 50 -10 60])
        AXIS equal
        xlabel('mm')
        ylabel('mm')
        grid on
        legend('Original','Deformed','IAR','Location','NorthWest')
        %AXIS([XMIN XMAX YMIN YMAX])

%Theta_2 Yeild and Endurance
subplot(2,2,3)
    plot(-(theta_2-theta_20)*180/pi,SF_yield,'-*',...
        -(theta_2-theta_20)*180/pi,SF_endurance,'-o')
        xlabel('\Delta theta2 (deg)')
        ylabel('SF')
        grid on
        legend('Yield','Endurance')

%r_1 Yeild and Endurance
subplot(2,2,4)
    plot((r_1-r_10)*1000,SF_yield,'-*',...
        (r_1-r_10)*1000,SF_endurance,'-o')
        xlabel('\Delta r1 (mm)')
        ylabel('SF')
        grid on
        legend('Yield','Endurance')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Useful data to output
```

```
    %disp=(r_1-r_10)'.*1000
    %torqueproduced = M_2'
    %rotation = ((theta_2-gtheta_2).*(180/pi))'
   % r_3_disp_mm;
   DeflectedY3
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.2 Virtual.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Virtual work equations to be solved
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Q = Virtual(CurrentPosition,X_2,Y_2,M_2)

%Parse the solution variables
r_1=CurrentPosition(1);
theta_1=CurrentPosition(2);
theta_2=CurrentPosition(3);

%Input the Dimension parameters
[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,a_2,b_2,c_2,d_2]...
    = Dimensions_0;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%To conform with boundry conditions of Physical testing
%angle of applied force in direction of r1
%X_2=Y_2*cos(theta_1);
%Y_2=Y_2*sin(theta_1);
%M_2=0;
%strict x y direction forces
```

```
%X_2=0;
%Y_2=0;
%M_2=-M_2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%!!!!! Spring Constants  From Joey Stuff
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%2D Model One of these is the slit design is used and two of each if the
%intact design is used
%when checking with Joey's remeber we have two let joints he has one
keq = TorsionSpring(L_tw,L_bw,L_bl,L_tl,t_let);
K_1 =keq*2;  %multiply by 2 because you split it in two top bottom
K_2=K_1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Moments and Forces of Springs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
T_1 = -K_1*(theta_1-theta_10);

T_2 = -K_2*((theta_2-theta_20)-(theta_1-theta_10));

F_s=LinearSpring(L_tw,L_tl,t_let,r_1,r_10);

%F_s=F_s;

%F_s=0;
%T_2=0;
%T_1=0;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Disc approximations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Fcx=ShearX(r_1,theta_1,theta_2); %Shear- anterior to posterior of disc
%Fcy=Compression(r_1,theta_1,theta_2);   %Compression of disc
%Mc=DiscTorque(theta_2);     %Torque rotation of disc
Fcx=0;
Fcy=0;
Mc=0;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Solve System of Equations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Q1=X_2*cos(theta_1)+Y_2*sin(theta_1)...     %Force Appl Forces
    +F_s...                                 %Force due to spring r_1 (LET)
    +Fcx*cos(theta_1)+Fcy*sin(theta_1);     %Forces due to disc approx.

Q2=-X_2*r_1*sin(theta_1)+Y_2*r_1*cos(theta_1)... %Force Appl Forces
    -Fcx*r_1*sin(theta_1)+Fcy*r_1*cos(theta_1)...%Forces disc approx.
    +T_1-T_2;                                    %Torques

Q3=X_2*(-a_2*sin(theta_2)-b_2*cos(theta_2))...      %Force Appl Forces
    +Y_2*(a_2*cos(theta_2)-b_2*sin(theta_2))...
    +T_2+M_2...                                    %Torques
    ...                                             %Forces disc approx.
    +Fcx*(-a_2*sin(theta_2)+c_2*cos(theta_2)+d_2*sin(theta_2))...
    +Fcy*(a_2*cos(theta_2)+c_2*sin(theta_2)-d_2*cos(theta_2))...
    +Mc;                                            %Moment disc approx
```

```
Q=[Q1;Q2;Q3];    %System of equations to be solved
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.3 Dimensions_0.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Initial input dimesions of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,...
    theta_20,a_2,b_2,c_2,d_2]=Dimensions_0

%Prototype Dimesions
L_bw=1.5/1000;    %m
L_tw=0.5/1000;   %m
L_tl=22/1000-L_bw;    %m
L_bl=8.5/1000-L_tw;  %m
t_let=4/1000;    %m

%Intial Position
r_10=L_bl+L_tw;     %m
theta_10=90*pi/180; %rad
theta_20=0*pi/180;  %rad

a_2=(5/1000+.03934+t_let/2);
b_2=.022;
c_2=18.9347/1000;   %m
d_2=0/1000;         %m

end
```

```
%%%%%%%%%%%%%%%%%%%%%%Prototype Dimesions
% L_bw=1.5/1000;    %m
% L_tw=.5/1000 ; %m
% %L_tl=22/1000-L_bw ;  %m
% L_tl=19/1000;
% L_bl=8.5/1000-L_tw ;  %m
% t_let=4/1000;    %m

% %Intial Position
% r_10=L_bl+L_tw; %m
% theta_10=90*pi/180; %rad
% theta_20=0*pi/180;  %rad
% a_2=(50-12.8/2)/1000;  %m
% b_2=(43-12.8/2)/1000;  %m
% c_2=9.56/1000;  %m
% d_2=5.87/1000;  %m

%%%%%%%%%%%%%%%%%%Manufactured F1717 Dimensions
% L_bw=0.006858;
% L_tw=0.007112;
% T=0.01651;
% L_tl=(T-L_bw)*10;
% L_bl=0.021336-L_tw;
% %t_let=0.000635;
% t_let=.00066;
% %Intial Position
% r_10=L_bl+L_tw; %m
% theta_10=90*pi/180; %rad
% theta_20=0*pi/180;  %rad
% a_2=0/1000;  %m
```

```
% b_2=.1/1000;  %m
% c_2=9.56/1000;  %m
% d_2=5.87/1000;  %m

% %%%%%%%%%%%%%Joseph Jacobsen Thesis Comparison dimensions
% L_bw=.005;
% L_tw=.0049;
% T=.0277;
% L_tl=(T-L_bw);
% L_bl=.0128-L_tw;
% t_let=.00081;
% %Intial Position
% r_10=L_bl+L_tw; %m
% theta_10=90*pi/180; %rad
% theta_20=0*pi/180;  %rad
% a_2=0/1000;  %m
% b_2=.1/1000;  %m
% c_2=9.56/1000;  %m
% d_2=5.87/1000;  %m
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.4 MaterialProps.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Material Properties Definitions
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [E,G,Sut,Syield,SF_des,Sendurance] = MaterialProps()

% input the material properties
E=110*10^9;  %Pa  titanium
nu=.29;
```

```
G=E/(2*(1+nu));

Sut = 2.20*10^8; %Ultimate strength in Pa
Syield = 377*10^6; %Yeild Strength in Pa
Sendurance = .55*Sut;

SF_des = 1.1; % Safety Factor
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.5 LinearSpring.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Linear spring estimate of the axial compression of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function F_s = LinearSpring(L_tw,L_tl,t_let,r_1,r_10)

[E,G,Sut,Syield,SF_des,Sendurance]=MaterialProps();

gamma_ce=0.85;
%Ktheta=2.65;
Ktheta=2.17;
I_ce=(t_let*(L_tw^3))/12
K_let=2*gamma_ce*Ktheta*E*I_ce/L_tl; % N_m/rad
THETA = asin((r_1-r_10)/(2*.85*L_tl));
F_s=(-4*K_let*THETA/(cos(THETA)*.85*L_tl));
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.6 TorsionSpring.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Torsional stiffness of the horizontal beams of the LET joint
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function keq = TorsionSpring(L_tw,L_bw,L_bl,L_tl,t_let)
[E,G,Sut,Syield,SF_des,Sendurance]=MaterialProps();
K_torsion=max(L_tw,t_let)*min(L_tw,t_let)^3*...
    (1/3-.21*min(L_tw,t_let)/max(L_tw,t_let)*...
    (1-min(L_tw,t_let)^4/(12*max(L_tw,t_let)^4)));
Kt=K_torsion*G/L_tl;
                %Li is the length of the torsional segment
               %G is the modulous of rigidity
               %Ki is analogous to J the polar moment of inertia for a
               %circular cross section
Ib=(L_bw*t_let^3)/12 ;
Kb=E*Ib/L_bl ;
                %E modulous of elasticity
               %Ib is the beams moment of Inertia
               %Lb is the length of the segment in bending
keq=(2*Kt^2*Kb)/(Kt^2+2*Kt*Kb);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.7 CenterRotation.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Calculate the Center of Rotation of the horizontal beam
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [interX,interY] = CenterRotation(r_1,theta_1,theta_2,i)
[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,...
    a_2,b_2,c_2,d_2]= Dimensions_0;
```

```
if i>=2
    Ax1=r_1(i-1)*cos(theta_1(i-1));
    Ay1=r_1(i-1)*sin(theta_1(i-1));

    Ax2=r_1(i)*cos(theta_1(i));
    Ay2=r_1(i)*sin(theta_1(i));

    Bx1=r_1(i-1)*cos(theta_1(i-1))+a_2*cos(theta_2(i-1));
    By1=r_1(i-1)*sin(theta_1(i-1))+a_2*sin(theta_2(i-1));

    Bx2=r_1(i)*cos(theta_1(i))+a_2*cos(theta_2(i));
    By2=r_1(i)*sin(theta_1(i))+a_2*sin(theta_2(i));

    A=[Ax2-Ax1,Ay2-Ay1];
    B=[Bx2-Bx1,By2-By1];

    magA=sqrt(A(1)^2+A(2)^2);
    magB=sqrt(A(1)^2+A(2)^2);

    unitA = A./magA;
    unitB = B./magB;

    midA = [(Ax1+Ax2)/2,(Ay1+Ay2)/2];
    midB = [(Bx1+Bx2)/2,(By1+By2)/2];

    slopeA = A(2)/A(1);
    slopeB = B(2)/B(1);
    inv_slopeA=-1/slopeA;
    inv_slopeB=-1/slopeB;
    Xguess = .3;
```

```
    P1 = [Xguess+midA(1),inv_slopeA*Xguess+midA(2)];
    P2 = [Xguess+midB(1),inv_slopeB*Xguess+midB(2)];

    AX= [Ax1,Ax2,midA(1),P1(1)];
    AY= [Ay1,Ay2,midA(2),P1(2)];
    BX= [Bx1,Bx2,midB(1),P2(1)];
    BY= [By1,By2,midB(2),P2(2)];

    x1=midA(1);
    x2=P1(1);
    x3=midB(1);
    x4=P2(1);

    y1=midA(2);
    y2=P1(2);
    y3=midB(2);
    y4=P2(2);

    interY=((x1*y2-y1*x2)*(y3-y4)-(x3*y4-x4*y3)*(y1-y2))/...
        ((x1-x2)*(y3-y4)-(y1-y2)*(x3-x4));
    interX=((x1*y2-y1*x2)*(x3-x4)-(x3*y4-x4*y3)*(x1-x2))/...
        ((x1-x2)*(y3-y4)-(y1-y2)*(x3-x4));
else
    interY=0;
    interX=0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.8 CrossSection.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Display the cross section of the LET joint
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function []= CrossSection()

[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,a_2,b_2,c_2,d_2]...
    =Dimensions_0;

P1x =0;
P1y =0;
P2x =L_tl+L_bw;
P2y = 0;
P3x =P2x;
P3y =L_bl+2*L_tw;
P4x =0;
P4y =P3y ;
P5x =0;
P5y =L_tw;
P6x =L_tl;
P6y =P5y;
P7x =P6x;
P7y =L_tw+L_bl;
P8x =0;
P8y =P7y;

X = [P1x,P2x,P3x,P4x,P8x,P7x,P6x,P5x];
Y=  [P1y,P2y,P3y,P4y,P8y,P7y,P6y,P5y];

figure(2)
```

```
    plot(X.*1000,Y.*1000,'-*')
        xlabel('mm')
        ylabel('mm')
        AXIS equal
Ymax=max(Y)
Xmax=max(X)
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.9 Compression.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Axial compression stiffness estimate of the disc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Fcy = Compression(r_1,theta_1,theta_2)

%input dimensions
[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,a_2,b_2,c_2,d_2]...
    = Dimensions_0;

Cy_0=r_10*sin(theta_10)+a_2*sin(theta_20)... %initial Position of disc
    -c_2*cos(theta_20)-d_2*sin(theta_20);   %estimate force app loc
Cy=r_1*sin(theta_1)+a_2*sin(theta_2)... %Deflected position of disc
    -c_2*cos(theta_2)-d_2*sin(theta_2);     %estimate force app loc

dispY=Cy-Cy_0; %displacement

Percentage=.8;   %adjust for weakening of collagen fibers.

Comp=1430*1000*Percentage;   %N/mm  1430N/mm*1000mm/m=N/m
```

```
Fcy=-Comp*dispY;  %force due to shear and 400N
                  %compresion through "approximate center"

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.10 ShearX.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Front and back shear force displacement relationship
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Fcx = ShearX(r_1,theta_1,theta_2)

[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,...
    a_2,b_2,c_2,d_2]= Dimensions_0;

Cx_0=r_10*cos(theta_10)+a_2*cos(theta_20)+...
    c_2*sin(theta_20)-d_2*cos(theta_20);
Cx=r_1*cos(theta_1)+a_2*cos(theta_2)+...
    c_2*sin(theta_2)-d_2*cos(theta_2);
dispX=Cx-Cx_0;

AnteriorShear=330*1000;   %N/mm  330N/mm*1000mm/m=N/m
PosteriorShear=200*1000;  %N/mm  200N/mm*1000mm/m=N/m

if dispX>0
    Fcx=-AnteriorShear*dispX;
else
    Fcx=-PosteriorShear*dispX;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.11 DiscTorque.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Torque rotation stiffness of the biologic disc estimate
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Mc = DiscTorque(theta_2)

[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,a_2,b_2,c_2,d_2]...
    = Dimensions_0;

rotz=theta_2-theta_20;

%Bottom
    %Extension
    A=6281.8;
    B=794.71;
    C=81.285;
    %Flexion
    D=12497;
    E=-1917.5;
    F=162.07;
%Top
    %Extension
    A=4674.5;
    B=729.85;
    C=81.51;
    %Flexion
    D=41913;
    E=-3297;
    F=133.59;
```

```
    if rotz<0
     Mc=-(A*rotz^3+B*rotz^2+C*rotz);
    else
        Mc=-(D*rotz^3+E*rotz^2+F*rotz);
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.12 FSU.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Torque-rotation of the "normal" case
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [NaturalFSU] = FSU()
%function [NaturalFSU,IVB_ALL_PLL] = FSU()
AngularDisp=(-10:1:10).*pi/180;

%Bottom
    %Extension
    A=6281.8;
    B=794.71;
    C=81.285;
    %Flexion
    D=12497;
    E=-1917.5;
    F=162.07;

%Top
    %Extension
    A=4674.5;
    B=729.85;
```

```
    C=81.51;
    %Flexion
    D=41913;
    E=-3297;
    F=133.59;

 for i=1:numel(AngularDisp)
    if AngularDisp(i)<0
     NaturalFSU(i,1)=-(A*AngularDisp(i)^3+...
            B*AngularDisp(i)^2+C*AngularDisp(i));
    else
        NaturalFSU(i,1)=-(D*AngularDisp(i)^3+...
        E*AngularDisp(i)^2+F*AngularDisp(i));
    end

    NaturalFSU(i,2)=AngularDisp(i);
 end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.13 StressCalc.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Stress Calculations of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [max_stress,SF_yield,SF_endurance] = StressCalc(r_1,theta_2)

[L_bw,L_tw,L_tl,L_bl,t_let,r_10,theta_10,theta_20,...
    a_2,b_2,c_2,d_2]=Dimensions_0;
[E,G,Sut,Syield,SF_des,Sendurance] = MaterialProps();

keq=TorsionSpring(L_tw,L_bw,L_bl,L_tl,t_let);%
```

```
%Stress Calculations
I_bend= (L_bw*t_let^3)/12;%
k_B=E*I_bend/L_bl;

gamma = 0.85;
%K_Theta = 2.65;
K_Theta=2.17;

Max_Theta_rad=theta_2-theta_20;
Torque = Max_Theta_rad*keq;%

I_ce =(t_let*(L_tw^3))/12;%
Kfg=2*gamma*K_Theta*E*I_ce/L_tl;

THETA = asin((r_1-r_10)/(2*gamma*L_tl));%
Faxial = 4*Kfg*THETA/(cos(THETA)*gamma*L_tl);%

Combinations = [1 1 1 1;
                1 1 1 -1;
                1 1 -1 1;
                1 1 -1 -1;
                1 -1 1 1;
                1 -1 1 -1;
                1 -1 -1 1;
                1 -1 -1 -1;
               -1 1 1 1;
               -1 1 1 -1;
               -1 1 -1 1;
               -1 1 -1 -1;
```

```
                -1 -1 1 1;
                -1 -1 1 -1;
                -1 -1 -1 1;
                -1 -1 -1 -1];

%% Stress Facet LET
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %tau_bh is the shear stress in the Horizontal member due to Bending
T_tau=(Torque)/2;
Q_torsion = (max(L_tw,t_let)^2*min(L_tw,t_let)^2)/...
    (3*max(L_tw,t_let)+1.8*min(L_tw,t_let));
tau_bh=T_tau/Q_torsion;

% %sigma_bv is the stress in the Vertical memeber due to Bending
sigma_bv=(Torque/2)*(t_let/2)/I_bend;

% %sigma_ev is the stress in the Vertical member
%due to Extension/compression
sigma_ev=Faxial/(2*(t_let*L_bw));

% %sigma_eh is the stress in Horizontal member due
%to Extension/compression
%sigma_eh=Faxial*gamma*L_tl*sin(THETA)*(L_tw/2)/(4*I_ce);
sigma_eh=(Faxial*L_tl*(L_tw/2))/(4*I_ce);
st=[];

for i=1:16

    s11 = Combinations(i,4)*sigma_ev;
```

```
    s22 = Combinations(i,2)*sigma_bv+Combinations(i,3)*sigma_eh;
    s33 = 0 ;
    s12 =0;
    s23 =Combinations(i,1)*tau_bh;
    s31 =0;
    st = [s11 , 0, 0; 0, s22, s23;  0, s23, 0 ];
    sigma_von(i)=sqrt(((s11-s22)^2+(s22-s33)^2+(s33-s11)^2+...
        6*(s12^2+s23^2+s31^2))/2);
end
max_stress=max(sigma_von)
SF_endurance=Sendurance/max_stress;
SF_yield=Syield/max_stress;

if SF_endurance>4
    SF_endurance = 4;
end
if SF_yield>4
    SF_yield=4;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

APPENDIX B. LATERAL BENDING MATLAB CODE

This appendix is the Matlab code for the virtual work calculation and optimization to find the equilibrium position of the FlexSuRe™ with and without the biologic estimate for lateral bending. This code consists of several functions which are separated by sections.

B.1 MainLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Main program driver for Lateral Bending
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
clc

[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,theta_20,...
    theta_30,theta_40,theta_50,theta_60]= DimensionsLAT();

r=[R1,R2,R3,R2,R1,R2,R3,R2];

CurrentPosition=[theta_20,theta_30,theta_40,theta_50,theta_60];

gtheta0 = [theta_30,pi-theta_20];

Theta78=fsolve(@(gtheta) ThetasLAT(gtheta,CurrentPosition,r),...
    gtheta0,optimset('Display','off'));
theta_70=Theta78(1);
theta_80=Theta78(2);
```

```
X2=r(2)*cos(theta_20);
Y2=r(2)*sin(theta_20);

X3=r(2)*cos(theta_20)+r(3)*cos(theta_30);
Y3=r(2)*sin(theta_20)+r(3)*sin(theta_30);

X4=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40);
Y4=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40);
%
Xa5=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +a_5*cos(theta_50);
Ya5=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +a_5*sin(theta_50);

Xb5=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +a_5*cos(theta_50)-b_5*sin(theta_50);
Yb5=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +a_5*sin(theta_50)+b_5*cos(theta_50);

Xadisc=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +a_disc*cos(theta_50);
Yadisc=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +a_disc*sin(theta_50);

Xbdisc=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +a_disc*cos(theta_50)-b_disc*sin(theta_50);
Ybdisc=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +a_disc*sin(theta_50)+b_disc*cos(theta_50);
```

```
X5=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +r(5)*cos(theta_50);
Y5=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +r(5)*sin(theta_50);

X6=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +r(5)*cos(theta_50)+r(6)*cos(theta_60);
Y6=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +r(5)*sin(theta_50)+r(6)*sin(theta_60);

X7=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +r(5)*cos(theta_50)+r(6)*cos(theta_60)-r(7)*cos(theta_70);
Y7=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +r(5)*sin(theta_50)+r(6)*sin(theta_60)-r(7)*sin(theta_70);

X8=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +r(5)*cos(theta_50)+r(6)*cos(theta_60)...
    -r(7)*cos(theta_70)-r(8)*cos(theta_80);
Y8=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +r(5)*sin(theta_50)+r(6)*sin(theta_60)-r(7)*sin(theta_70)...
    -r(8)*sin(theta_80);

X1=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)...
    +r(5)*cos(theta_50)+r(6)*cos(theta_60)-r(7)*cos(theta_70)...
    -r(8)*cos(theta_80)-r(1);
Y1=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)...
    +r(5)*sin(theta_50)+r(6)*sin(theta_60)...
    -r(7)*sin(theta_70)-r(8)*sin(theta_80);

PX=[0,X2,X3,X4,Xa5,Xb5,Xa5,Xadisc,Xbdisc,Xadisc,X5,X6,X7,X8,X1];
```

```
PY=[0,Y2,Y3,Y4,Ya5,Yb5,Ya5,Yadisc,Ybdisc,Yadisc,Y5,Y6,Y7,Y8,Y1];

x0=[theta_20,theta_30,theta_40,theta_50,theta_60];
Fy=[-19.76,-17.76,-15.78,-13.79,-11.81,-9.84,-7.87,...
    -5.90,-3.93,-1.97,0,1.97,3.93,5.90,7.87,9.84,11.81,...
    13.79,15.78,17.77,19.76];
Fx=0;
%M=[-10:1:10];
M=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
for i=1:numel(M)
    options = optimset('Algorithm','active-set','Display','off');
    x=fsolve(@(x) VirtualLAT(x,Fx,Fy(i),M(i),r),x0);

    theta_2(i)=x(1);
    theta_3(i)=x(2);
    theta_4(i)=x(3);
    theta_5(i)=x(4);
    theta_6(i)=x(5);

    DefYb5(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
        r(4)*sin(theta_4(i))+a_5*sin(theta_5(i))+b_5*cos(theta_5(i));

    DisplacementY(i)=-(Yb5-DefYb5(i));  %meters
   % figure(2)
   % plot(theta_5,M,'.-')
   % ylabel('torque (Newtons-Meter)')
   % xlabel('Displacement (mm)')
    %AXIS([-10 10 -10 10])
   % grid on
```

```
CurrentPosition=[theta_2(i),theta_3(i),theta_4(i),...
    theta_5(i),theta_6(i)];

gtheta0 = [theta_3(i),pi-theta_2(i)];

Theta78=fsolve(@(gtheta) ThetasLAT(gtheta,CurrentPosition,r),...
    gtheta0,optimset('Display','off'));
theta_7(i)=Theta78(1);
theta_8(i)=Theta78(2);

defX2(i)=r(2)*cos(theta_2(i));
defY2(i)=r(2)*sin(theta_2(i));

defX3(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i));
defY3(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i));

defX4(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i));
defY4(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i));

defXa5(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+a_5*cos(theta_5(i));
defYa5(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+a_5*sin(theta_5(i));

defXb5(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+a_5*cos(theta_5(i))-b_5*sin(theta_5(i));
defYb5(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+a_5*sin(theta_5(i))+b_5*cos(theta_5(i));
```

```
defXadisc(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+a_disc*cos(theta_5(i));
defYadisc(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+a_disc*sin(theta_5(i));

defXbdisc(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+a_disc*cos(theta_5(i))...
    -b_disc*sin(theta_5(i));
defYbdisc(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+a_disc*sin(theta_5(i))+...
    b_disc*cos(theta_5(i));

defX5(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))...
    +r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i));
defY5(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))...
    +r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i));

defX6(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))...
    +r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i))+...
    r(6)*cos(theta_6(i));
defY6(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i))+...
    r(6)*sin(theta_6(i));

defX7(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i))+...
    r(6)*cos(theta_6(i))-r(7)*cos(theta_7(i));
defY7(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i))+...
```

```
    r(6)*sin(theta_6(i))-r(7)*sin(theta_7(i));

defX8(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i))+...
    r(6)*cos(theta_6(i))-r(7)*cos(theta_7(i))-r(8)*cos(theta_8(i));
defY8(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i))+...
    r(6)*sin(theta_6(i))-r(7)*sin(theta_7(i))-r(8)*sin(theta_8(i));

defX1(i)=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i))+...
    r(6)*cos(theta_6(i))-r(7)*cos(theta_7(i))-...
    r(8)*cos(theta_8(i))-r(1);
defY1(i)=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i))+...
    r(6)*sin(theta_6(i))-r(7)*sin(theta_7(i))-...
    r(8)*sin(theta_8(i));

PointsX=[0,defX2(i),defX3(i),defX4(i),defXa5(i),...
    defXb5(i),defXa5(i),defXadisc(i),defXbdisc(i),...
    defXadisc(i),defX5(i),defX6(i),defX7(i),defX8(i),defX1(i)];
PointsY=[0,defY2(i),defY3(i),defY4(i),defYa5(i),...
    defYb5(i),defYa5(i),defYadisc(i),defYbdisc(i),...
    defYadisc(i),defY5(i),defY6(i),defY7(i),defY8(i),defY1(i)];

figure(3)
plot(PointsX,PointsY,'.-',PX,PY,'-.*')
AXIS equal
xlabel('m')
ylabel('m')
```

```
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Intact and Degenerate cases
%Data in FSULAT.m file
[NaturalFSU] = FSULAT();
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure(4)
    plot(-NaturalFSU(:,1),-NaturalFSU(:,2).*(180/pi),'-',...
        -M,-(theta_5-theta_50).*(180/pi),'-*')
        ylabel('Rotation (deg)')
        xlabel('Torque (Newton-meter)')
        %AXIS([-10 10 -10 10])
        grid on
        legend('Natural','Assembly','Location','SouthEast')

%%%%%%%%%%Outputs
disp=DisplacementY'*1000
%trans_Y_2=Y_2';
MomentTrans=-M';
RotationTrans=(-(theta_5-theta_50).*(180/pi))';

StressCalcLat(Fy)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.2 VirtualLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Virtual work equations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function Q = VirtualLAT(CurrentPosition,Fx,Fy,M,r)


[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,theta_20,...
    theta_30,theta_40,theta_50,theta_60]= DimensionsLAT();

%Parse the solution variable
 theta_2=CurrentPosition(1);
 theta_3=CurrentPosition(2);
 theta_4=CurrentPosition(3);
 theta_5=CurrentPosition(4);
 theta_6=CurrentPosition(5);

gtheta0 = [theta_3,pi-theta_2];
Theta78=fsolve(@(gtheta) ThetasLAT(gtheta,CurrentPosition,r),gtheta0,...
    optimset('Display','off'));

theta_7=Theta78(1);
theta_8=Theta78(2);

theta_70=theta_30;
theta_80=theta_20-pi;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
k = FixedGuidedLAT(L_tw,L_bw,L_bl,L_tl,t_let);

%Moments and Forces of Springs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
T_2 = -k*(theta_2-theta_20);
T_3 = -k*((theta_2-theta_20)-(theta_3-theta_30));
```

```
T_4 = -k*((theta_3-theta_30)-(theta_4-theta_40));
T_5 = -k*((theta_4-theta_40)-(theta_5-theta_50));
T_6 = -k*((theta_5-theta_50)-(theta_6-theta_60));
T_7 = -k*((theta_8-theta_80)-(theta_7-theta_70));
T_8 = -k*(theta_8-theta_80);
T_9 = -k*((theta_7-theta_70)-(theta_6-theta_60));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

allTheta=[0,theta_2,theta_3,theta_4,theta_5,theta_6,theta_7,theta_8];
[d7d2,d8d2]= KinematicCoefLAT(2,allTheta,r);
[d7d3,d8d3]= KinematicCoefLAT(3,allTheta,r);
[d7d4,d8d4]= KinematicCoefLAT(4,allTheta,r);
[d7d5,d8d5]= KinematicCoefLAT(5,allTheta,r);
[d7d6,d8d6]= KinematicCoefLAT(6,allTheta,r);

%Disc Estimate
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Fxdisc=ShearXLAT(theta_2,theta_3,theta_4,theta_5);
%Fydisc=CompressionLAT(theta_2,theta_3,theta_4,theta_5);
%Mdisc=-DiscTorqueLAT(theta_5);

Fxdisc=0;
Fydisc=0;
Mdisc=0;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%!!!!!Solve Equations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Q2=-(Fx+Fxdisc)*r(2)*sin(theta_2)+...
    (Fy+Fydisc)*r(2)*cos(theta_2)+T_2+T_3+...
```

```
    T_7*d8d2-T_7*d7d2+T_8*d8d2+T_9*d7d2;

Q3=-(Fx+Fxdisc)*r(3)*sin(theta_3)+...
    (Fy+Fydisc)*r(3)*cos(theta_3)-T_3+T_4+...
    T_7*d8d3-T_7*d7d3+T_8*d8d3+T_9*d7d3;

Q4=-(Fx+Fxdisc)*r(4)*sin(theta_4)+...
    (Fy+Fydisc)*r(4)*cos(theta_4)-T_4+T_5+...
    T_7*d8d4-T_7*d7d4+T_8*d8d4+T_9*d7d4;

Q5=(Fx*(-a_5*sin(theta_5)-b_5*cos(theta_5))+...
    Fy*(a_5*cos(theta_5)-...
    b_5*sin(theta_5)))+(Fxdisc*(-a_disc*sin(theta_5)-...
    b_disc*cos(theta_5))+Fydisc*(a_disc*cos(theta_5)-...
    b_disc*sin(theta_5)))+...
    M+Mdisc-T_5+T_6+T_7*d8d5-T_7*d7d5+T_8*d8d5+T_9*d7d5;

Q6=-T_6-T_9+T_7*d8d6-T_7*d7d6+T_8*d8d6+T_9*d7d6;

Q=[Q2;Q3;Q4;Q5;Q6];

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.3 DimensionsLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Input dimensions of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,...
    a_disc,b_disc,theta_20,theta_30,theta_40,theta_50,theta_60]=...
```

```
    DimensionsLAT()

L_bw=1.5/1000;
L_tw=0.5/1000;
L_tl=22/1000-L_bw;
L_bl=8.5/1000-L_tw;
t_let=4/1000;

gamma=0.85;
r_base=58.5258/1000;
R1=r_base+(1-gamma)*L_tl;
R2=gamma*L_tl;
R3=L_bl+L_tw;

a_5=R1/2;
b_5=0;  %m

a_disc=a_5;
b_disc=-9.46734/1000;

theta_20=180*pi/180;
theta_30=90*pi/180;
theta_40=0*pi/180;
theta_50=0*pi/180;
theta_60=0*pi/180;

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.4 MaterialPropsLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Material Properties of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [E,G,Sut,Syield,SF] = MaterialPropsLAT()

% input the material properties
E=110e9;  %Pa  titanium
nu=.29;
G=E/(2*(1+nu));
Sut = 2.20*10^8; %Ultimate strength in Pa
Syield = 140*10^6; %Yeild Strength in Pa
SF = 1.1; % Safety Factor
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.5 KinematicCoefLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Kinematic coeficients of the lateral bending derivation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [d7,d8] = KinematicCoefLAT(i,theta,r)

    if theta(8)==0
        theta(8)=.000000001;
    end

d7=(r(i)*cos(theta(i))-r(i)*sin(theta(i))*cot(theta(8)))/...
    (r(7)*cos(theta(7))-r(7)*sin(theta(7))*cot(theta(8)));
d8=(r(i)*sin(theta(i))-r(7)*sin(theta(7))*d7)/(r(8)*sin(theta(8)));
```

```
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.6 ThetasLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Find the remaining angles not initially defined
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function XY = ThetasLAT(gtheta,CurrentPosition,r)

%Parse the solution variable
theta_2=CurrentPosition(1);
theta_3=CurrentPosition(2);
theta_4=CurrentPosition(3);
theta_5=CurrentPosition(4);
theta_6=CurrentPosition(5);

theta_7=gtheta(1);
theta_8=gtheta(2);
X1=r(2)*cos(theta_2)+r(3)*cos(theta_3)+r(4)*cos(theta_4)+...
    r(5)*cos(theta_5)+r(6)*cos(theta_6)-r(1)-...
    r(7)*cos(theta_7)-r(8)*cos(theta_8);
Y1=r(2)*sin(theta_2)+r(3)*sin(theta_3)+r(4)*sin(theta_4)+...
    r(5)*sin(theta_5)+r(6)*sin(theta_6)-r(7)*sin(theta_7)-...
    r(8)*sin(theta_8);
XY=[X1;Y1];
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.7 FixedGuidedLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Spring constant for the fixed-guided beams
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function k = FixedGuidedLAT(L_tw,L_bw,L_bl,L_tl,t_let)
[E,G,Sut,Syield,SF]=MaterialPropsLAT();

gamma_ce=0.85;
%Ktheta=2.17;
Ktheta=2.65;
I_ce=(t_let*(L_tw^3))/12;

k=2*gamma_ce*Ktheta*E*I_ce/L_tl; % N_m/rad
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.8 CenterRotationLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculate the center of rotation of the center-top beam
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [interX,interY] = CenterRotationLAT(theta_2,...
    theta_3,theta_4,theta_5,i)

if i>=2
[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,...
    theta_20,theta_30,theta_40,theta_50,theta_60]= DimensionsLAT();
r=[R1,R2,R3,R2,R1,R2,R3,R2];
```

```
Ax1=r(2)*cos(theta_2(i-1))+r(3)*cos(theta_3(i-1))+...
    r(4)*cos(theta_4(i-1));
Ay1=r(2)*sin(theta_2(i-1))+r(3)*sin(theta_3(i-1))+...
    r(4)*sin(theta_4(i-1));

Ax2=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i));
Ay2=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i));

Bx1=r(2)*cos(theta_2(i-1))+r(3)*cos(theta_3(i-1))+...
    r(4)*cos(theta_4(i-1))+r(5)*cos(theta_5(i-1));
By1=r(2)*sin(theta_2(i-1))+r(3)*sin(theta_3(i-1))+...
    r(4)*sin(theta_4(i-1))+r(5)*sin(theta_5(i-1));

Bx2=r(2)*cos(theta_2(i))+r(3)*cos(theta_3(i))+...
    r(4)*cos(theta_4(i))+r(5)*cos(theta_5(i));
By2=r(2)*sin(theta_2(i))+r(3)*sin(theta_3(i))+...
    r(4)*sin(theta_4(i))+r(5)*sin(theta_5(i));

A=[Ax2-Ax1,Ay2-Ay1];
B=[Bx2-Bx1,By2-By1];

magA=sqrt(A(1)^2+A(2)^2);
magB=sqrt(A(1)^2+A(2)^2);

unitA = A./magA;
unitB = B./magB;

midA = [(Ax1+Ax2)/2,(Ay1+Ay2)/2];
```

```
midB = [(Bx1+Bx2)/2,(By1+By2)/2];

slopeA = A(2)/A(1);
slopeB = B(2)/B(1);
inv_slopeA=-1/slopeA;
inv_slopeB=-1/slopeB;

Xguess = .5;

P1 = [Xguess+midA(1),inv_slopeA*Xguess+midA(2)];
P2 = [Xguess+midB(1),inv_slopeB*Xguess+midB(2)];

AX= [Ax1,Ax2,midA(1),P1(1)];
AY= [Ay1,Ay2,midA(2),P1(2)];
BX= [Bx1,Bx2,midB(1),P2(1)];
BY= [By1,By2,midB(2),P2(2)];

x1=midA(1);
x2=P1(1);
x3=midB(1);
x4=P2(1);

y1=midA(2);
y2=P1(2);
y3=midB(2);
y4=P2(2);

interY=((x1*y2-y1*x2)*(y3-y4)-(x3*y4-x4*y3)*(y1-y2))/...
    ((x1-x2)*(y3-y4)-(y1-y2)*(x3-x4));
interX=((x1*y2-y1*x2)*(x3-x4)-(x3*y4-x4*y3)*(x1-x2))/...
```

```
    ((x1-x2)*(y3-y4)-(y1-y2)*(x3-x4));

figure(6)
plot(AX,AY,'-*',BX,BY,'-*',interX,interY,'*')

Axis equal
else
    interY=0;
    interX=0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.9 CompressionLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compression force deflection estimate of biologic disc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Fcy = CompressionLAT(theta_2,theta_3,theta_4,theta_5)

[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,...
    theta_20,theta_30,theta_40,theta_50,theta_60]= DimensionsLAT;

r=[R1,R2,R3,R2,R1,R2,R3,R2];

Cy_0=r(2)*sin(theta_20)+r(3)*sin(theta_30)+r(4)*sin(theta_40)+...
    a_disc*sin(theta_50)+b_disc*cos(theta_50);
Cy=r(2)*sin(theta_2)+r(3)*sin(theta_3)+r(4)*sin(theta_4)+...
    a_disc*sin(theta_5)+b_disc*cos(theta_5);

dispY=Cy-Cy_0;
Comp=1430*1000;    %N/mm  1430N/mm*1000mm/m=N/m
```

```
Fcy=-Comp*dispY;

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.10 ShearXLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Lateral shear of the device
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Fcx = ShearXLAT(theta_2,theta_3,theta_4,theta_5)

[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,...
    theta_20,theta_30,theta_40,theta_50,theta_60]= DimensionsLAT;

r=[R1,R2,R3,R2,R1,R2,R3,R2];

Cx_0=r(2)*cos(theta_20)+r(3)*cos(theta_30)+r(4)*cos(theta_40)+...
    a_disc*cos(theta_50)-b_disc*sin(theta_50);
Cx=r(2)*cos(theta_2)+r(3)*cos(theta_3)+r(4)*cos(theta_4)+...
    a_disc*cos(theta_5)-b_disc*sin(theta_5);

dispX=Cx-Cx_0;

Shear=278*1000;   %N/mm

if dispX>0
    Fcx=-Shear*dispX;
else
    Fcx=-Shear*dispX;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.11 DiscTorqueLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Disc torque-rotation of the biologic disc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Mc = DiscTorqueLAT(theta_5)

[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,...
    theta_20,theta_30,theta_40,theta_50,theta_60]= DimensionsLAT;

rotz=theta_5-theta_50;

    A=-690.17;
    B=337.1;
    C=52.74;

    D=16015;
    E=-1263.1;
    F=73.346;

    if rotz>0
     Mc=(A*abs(rotz)^3+B*abs(rotz)^2+C*abs(rotz));
    else
        Mc=-(D*abs(rotz)^3+E*abs(rotz)^2+F*abs(rotz));
    end

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.12 FSULAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Stiffness of the "normal" disc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [NaturalFSU] = FSULAT()
AngularDisp=(-10:1:10).*pi/180;

%Bottom
    %Extension
    A=19665;
    B=-627.1;
    C=51.353;
    %Flexion
    D=47.994;
    E=260.86;
    F=37.474;
%Top
    %Extension
    % A=-690.17;
    % B=337.1;
    % C=52.74;
    %Flexion
    % D=16015;
    % E=-1263.1;
    % F=73.346;

 for i=1:numel(AngularDisp)
    if AngularDisp(i)>0
     NaturalFSU(i,1)=(A*abs(AngularDisp(i))^3+...
            B*abs(AngularDisp(i))^2+C*abs(AngularDisp(i)));
```

```
    else
        NaturalFSU(i,1)=-(D*abs(AngularDisp(i))^3+...
            E*abs(AngularDisp(i))^2+F*abs(AngularDisp(i)));
    end
    NaturalFSU(i,2)=AngularDisp(i);
 end

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.13 StressCalcLAT.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Stress Calculations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [max_stress,SF_yield,SF_endurance] = StressCalcLAT(Fy)

%Valid for force placed at the center of the middle beam

%Two components
    %Fixed guided stress
     %axial stress on bending-verticle member
[L_bw,L_tw,L_bl,L_tl,t_let,a_5,b_5,R1,R2,R3,a_disc,b_disc,theta_20,...
    theta_30,theta_40,theta_50,theta_60]= DimensionsLAT();

%input dimensions and load
[E,G,Sut,Syield,SF]=MaterialPropsLAT();

Fy=Fy./2;

for j=1:numel(Fy)
```

```
I_fixed_guided=(t_let*L_tw^3)/12;

sigma_eh = ((Fy(j))*L_tl*(L_tw/2))/(2*I_fixed_guided);

 Combinations = [1 1 1 1;
                 1 1 1 -1;
                 1 1 -1 1;
                 1 1 -1 -1;
                 1 -1 1 1;
                 1 -1 1 -1;
                 1 -1 -1 1;
                 1 -1 -1 -1;
                -1 1 1 1;
                 -1 1 1 -1;
                 -1 1 -1 1;
               -1 1 -1 -1;
                 -1 -1 1 1;
                 -1 -1 1 -1;
              -1 -1 -1 1;
                -1 -1 -1 -1];
%
 % %sigma_ev is the stress in the Vertical member
 %due to Extension/compression
Faxial=Fy(j);
 sigma_ev=Faxial/(t_let*L_bw);
sigma_ev=0;
 st=[];

 sigma_bv=0;
```

```
 tau_bh=0;

 for i=1:16
%
     s11 = Combinations(i,4)*sigma_ev;
    s22 = Combinations(i,2)*sigma_bv+Combinations(i,3)*sigma_eh;
    s33 = 0 ;
    s12 =0;
    s23 =Combinations(i,1)*tau_bh;
    s31 =0;

    st = [s11 , 0, 0; 0, s22, s23;  0, s23, 0 ];

    sigma_von(i)=sqrt(((s11-s22)^2+(s22-s33)^2+(s33-s11)^2+...
        6*(s12^2+s23^2+s31^2))/2);
end

 max_stress(j)=max(sigma_von)

end
output_stress=max_stress'
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

APPENDIX C. ANSYS BATCH FILE

This appendix is the Ansys batch file for the finite element analysis of the FlexSuRe™ spinal implant. The load application location is applied to different nodes to correspond with testing boundary conditions.

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!Ansys Batch File for F1717 Type Structure
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
FINISH
/CLEAR,START
/PREP7
!!!!!!Variables
Theta=-00
pi=3.141592654
beta1=(90+Theta)*pi/180
beta2=(90+2*Theta)*pi/180
beta3=(90+3*Theta)*pi/180
spacer=5/1000  !!!
UHMWPE=.03934  !!!
forceapp =spacer+UHMWPE
midline=0.020
flair=00*pi/180
L_tl =22/1000     !!!!
L_tw = 0.5/1000   !!!
L_bl =  8.5/1000    !!!
L_bw = 1.5/1000   !!!
```

```
t_let=4/1000   !!!!
L_base=.022   !!!!!
upright_w=.005  !!!!
t_support=.00475
!!!!!!!!!!!!Beam Set up
ET,1,Shell281
ET,2,BEAM4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!material properties
MPTEMP,,,,,,,,
MPTEMP,1,0
MPDATA,EX,1,,110e9    !!Mod of elasitcity
MPDATA,PRXY,1,,0.29    !!poisson's
MPTEMP,,,,,,,,
MPTEMP,1,0
MPDATA,EX,2,,100e12   !Approximately Rigid
MPDATA,PRXY,2,,0.29

!!!!!!!!!!!!KeyPoints and Lines
!!Outer Edge
K,1,0,0,-midline,
K,2,-L_base*cos(beta1),L_base*sin(beta1),-midline,
K,3,-L_base*cos(beta1)-L_bl*cos(beta2),L_base*sin(beta1)+L_bl*sin(beta2),...
-midline
K,4,-L_base*cos(beta1)-L_bl*cos(beta2)-L_base*cos(beta3),L_base*sin(beta1)+...
L_bl*sin(beta2)+L_base*sin(beta3),-midline
K,5,0,0,midline,
K,6,-L_base*cos(beta1),L_base*sin(beta1),midline,
K,7,-L_base*cos(beta1)-L_bl*cos(beta2),L_base*sin(beta1)+L_bl*sin(beta2),...
midline
K,8,-L_base*cos(beta1)-L_bl*cos(beta2)-L_base*cos(beta3),L_base*sin(beta1)+...
```

```
L_bl*sin(beta2)+L_base*sin(beta3),midline
K,9,-L_base*cos(beta1)-L_tl*sin(flair),L_base*sin(beta1),...
-midline-L_tl*cos(flair)
K,10,-L_base*cos(beta1)-L_tl*sin(flair),L_base*sin(beta1),...
midline+L_tl*cos(flair)
K,11,kx(3)-L_tl*sin(flair),ky(3),-midline-L_tl*cos(flair)
K,12,kx(3)-L_tl*sin(flair),ky(3),midline+L_tl*cos(flair)
K,13,Kx(4),Ky(4),0
K,14,kx(4)+forceapp*sin(beta3),Ky(4)+forceapp*cos(beta3),0
K,15,0,0,0
!!!Right
K,16,Kx(1),Ky(1),Kz(1)+upright_w
K,17,Kx(2),Ky(2),Kz(2)+upright_w
K,18,-L_base*cos(beta1)-L_tw*cos(beta2),L_base*sin(beta1)+...
L_tw*sin(beta2),Kz(17)
K,19,-L_base*cos(beta1)-(L_bl-L_tw)*cos(beta2),L_base*sin(beta1)+...
(L_bl-L_tw)*sin(beta2),Kz(17)
K,20,Kx(3),Ky(3),Kz(3)+upright_w
K,21,Kx(4),Ky(4),Kz(4)+upright_w
K,22,Kx(18)-(L_tl-L_bw)*sin(flair),Ky(18),-midline-(L_tl-L_bw)*cos(flair)
K,23,Kx(19)-(L_tl-L_bw)*sin(flair),Ky(19),-midline-(L_tl-L_bw)*cos(flair)
K,24,Kx(18),Ky(18),-midline
K,25,Kx(19),Ky(19),-midline

!!! left
K,26,Kx(5),Ky(5),Kz(5)-upright_w
K,27,Kx(6),Ky(6),Kz(6)-upright_w
K,28,-L_base*cos(beta1)-L_tw*cos(beta2),...
L_base*sin(beta1)+L_tw*sin(beta2),Kz(27)
K,29,-L_base*cos(beta1)-(L_bl-L_tw)*cos(beta2),...
```

```
L_base*sin(beta1)+(L_bl-L_tw)*sin(beta2),Kz(27)
K,30,Kx(7),Ky(7),Kz(7)-upright_w
K,31,Kx(8),Ky(8),Kz(8)-upright_w
K,32,Kx(28)-(L_tl-L_bw)*sin(flair),Ky(28),midline+(L_tl-L_bw)*cos(flair)
K,33,Kx(29)-(L_tl-L_bw)*sin(flair),Ky(29),midline+(L_tl-L_bw)*cos(flair)
K,34,Kx(28),Ky(28),midline
K,35,Kx(29),Ky(29),midline
K,60,forceapp,0,0

Lstr,1,2
Lstr,3,4
Lstr,5,6
Lstr,7,8
Lstr,9,2
Lstr,6,10
Lstr,11,3
Lstr,7,12
Lstr,9,11
Lstr,10,12
Lstr,13,31
Lstr,13,14
Lstr,15,16
Lstr,16,1
Lstr,16,17
Lstr,17,18
Lstr,18,24
Lstr,24,22
Lstr,16,17
Lstr,19,20
Lstr,20,21
```

```
Lstr,22,23
Lstr,23,25
Lstr,19,25
Lstr,26,15
Lstr,4,21
Lstr,13,21
Lstr,20,3
Lstr,2,17
Lstr,2,24
Lstr,3,25
Lstr,8,31
Lstr,31,30
Lstr,7,30
Lstr,27,6
Lstr,26,27
Lstr,5,26
Lstr,27,28
Lstr,28,34
Lstr,6,34
Lstr,29,35
Lstr,35,7
Lstr,29,30
Lstr,33,35
lstr,32,33
lstr,32,34

Fillet_rad=1/1000
LFILLT,8,10,Fillet_rad, ,
LFILLT,6,10,Fillet_rad, ,
LFILLT,44,43,Fillet_rad, ,
```

```
LFILLT,44,45,Fillet_rad, ,
LFILLT,7,9,Fillet_rad, ,
LFILLT,5,9,Fillet_rad, ,
LFILLT,18,21,Fillet_rad, ,
LFILLT,21,22,Fillet_rad, ,
lstr,60,15

/PNUM,KP,1
/PNUM,LINE,0
/PNUM,AREA,0
/PNUM,VOLU,0
/PNUM,NODE,0
/PNUM,TABN,0
/PNUM,SVAL,0
/NUMBER,0
/PNUM,ELEM,0
/REPLOT
GPLOT

!!!!!!!AREAS
FLST,2,12,4
FITEM,2,30
FITEM,2,7
FITEM,2,50
FITEM,2,9
FITEM,2,51
FITEM,2,5
FITEM,2,29
FITEM,2,18
FITEM,2,52
```

```
FITEM,2,21
FITEM,2,53
FITEM,2,22

AL,P51X
AL,14,15,1,28
AL,17,16,29,28
AL,23,30,19,27
AL,27,2,25,20
AL,36,3,35,34
Al,39,38,37,34

FLST,2,12,4
FITEM,2,8
FITEM,2,41
FITEM,2,43
FITEM,2,48
FITEM,2,44
FITEM,2,49
FITEM,2,45
FITEM,2,39
FITEM,2,6
FITEM,2,47
FITEM,2,10
FITEM,2,46
AL,P51X

Al,40,42,41,33
AL,33,32,4,31
```

```
R,1,t_let,t_let,t_let,t_let, , ,
R,2,t_support,t_support,t_support,t_support, , ,

!!!!!!!!!!!!!!!!!!!!!!!!!!!!AREA
ASEL,s,AREA,,1
AATT,1,1,1,0,-1
ASEL,all
ASEL,s,AREA,,8
AATT,1,1,1,0,-1
ASEL,all
ASEL,s,AREA,,2
AATT,1,2,1,0,-1
ASEL,all
ASEL,s,AREA,,3
AATT,1,2,1,0, -1
ASEL,all
ASEL,s,AREA,,4
AATT,1,2,1,0,-1
ASEL,all
ASEL,s,AREA,,5
AATT,1,2,1,0, -1
ASEL,all
ASEL,s,AREA,,6
AATT,1,2,1,0,-1
ASEL,all
ASEL,s,AREA,,7
AATT,1,2,1,0, -1
ASEL,all
ASEL,s,AREA,,9
AATT,1,2,1,0, -1
```

```
ASEL,all
ASEL,s,AREA,,10
AATT,1,2,1,0,-1
ASEL,all
LESIZE,ALL, , ,10, ,1, , ,1,

MSHKEY,0
FLST,5,10,5,ORDE,2
FITEM,5,1
FITEM,5,-10
CM,_Y,AREA
ASEL, , , ,P51X
CM,_Y1,AREA
CHKMSH,'AREA'
CMSEL,S,_Y
AMESH,_Y1
CMDELE,_Y
CMDELE,_Y1
CMDELE,_Y2

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!Beam Properties
b=.005
h=.005
Izz=b*h*h*h/12
Iyy=h*b*b*b/12
A=b*h
tor_rigid=0

R,5,A,Izz,Iyy,b,h,0,
RMORE,0,tor_rigid,0,0,0,0,
```

```
LSEL,s,LINE,,11
LATT,2,5
Lsel,all
LSEL,s,LINE,,26
LATT,2,5
Lsel,all
LSEL,s,LINE,,24
LATT,2,5
Lsel,all
LSEL,s,LINE,,13
LATT,2,5
Lsel,all
LSEL,s,LINE,,12
LATT,2,5
Lsel,all
LSEL,s,LINE,,54
LATT,2,5
Lsel,all

*SET,Divisions,10
LESIZE,ALL, , ,Divisions, ,1, , ,1
FLST,2,6,5,ORDE,5
FITEM,2,11
FITEM,2,-13
FITEM,2,24
FITEM,2,26
FITEM,2,54
LMESH,P51X

YDisplacement = 10/1000
```

```
Loadsteps=10
!!!!!!Apply Loads
/SOL
DK,60,uy,0
DK,60,uz,0
DK,60,ux,0
dk,60,rotx,0
dk,60,roty,0
dk,60,rotz,0
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
*DO,mm,1,Loadsteps,1
DK,14,uy,(mm*yDisplacement)/Loadsteps
lswrite,mm
*ENDDO
nlgeom,on
lssolve,1,Loadsteps
FINISH
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```