

DESIGN OF COMPLIANT MECHANISM LATTICE STRUCTURES FOR
IMPACT ENERGY ABSORPTION

A Thesis

Submitted to the Faculty

of

Purdue University

by

Joel Christian Najmon

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Mechanical Engineering

December 2017

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Andres Tovar, Chair

Department of Mechanical Engineering

Dr. Jong Ryu

Department of Mechanical Engineering

Dr. Khosrow Nematollahi

Department of Mechanical Engineering

Approved by:

Dr. Sohel Anwar

Chair of the Graduate Program

This thesis work is dedicated to my beloved friends and family. I am especially grateful towards my parents, Ralph and Susan, and my siblings Seth, Lydia, and Mary, for their continual support and encouragement.

ACKNOWLEDGMENTS

I would like to give recognition to my advisor, Dr. Andres Tovar, for his guidance and teaching this past year. He has far exceeded my expectations as a mentor. I am so thankful that I've had the opportunity to do research with him and look forward to future work.

I would also like to thank Dr. Jong Ryu and Dr. Khosrow Nematollahi for their suggestions and advice to help improve my Master's thesis. I want to acknowledge the Sports Innovation Institute at IUPUI for their partial support of this work.

I would like to also acknowledge all those who helped me through this endeavor. Auturo Garica for his help with the initial code and design of compliant mechanisms. Jacob DeHart and Zeb Woods for their help with human modeling and helmet design inspiration. And to those who listened and provided technical feedback on my work: Ralph Najmon, Clayton Parisey, Joesph Derrick and my fellow labmates and research assistants.

Finally, my most sincere thanks to my family, friends, and church family for their steadfast support.

“I can do all things through Christ who strengthens me.” Philippians 4:13

TABLE OF CONTENTS

	Page
LIST OF TABLES	viii
LIST OF FIGURES	ix
ABBREVIATIONS	xiii
ABSTRACT	xv
1. INTRODUCTION	1
1.1 Types of Energy Absorbing Lattice Structures	1
1.1.1 Foams	1
1.1.2 Truss-based Lattice Structures	1
1.2 Compliant Mechanisms	3
1.2.1 Maximum Mutual Potential Energy Objective Function for Topol- ogy Optimization	4
1.2.2 Mechanical and Geometric Advantage	7
1.3 Compliant Mechanisms for Energy Absorption	7
1.4 Existing Helmet Liners	8
1.5 Motivation and Objectives	9
1.5.1 Motivation	9
1.5.2 Objectives	10
2. PROPOSED DESIGN OPTIMIZATION TOOL	11
2.1 Stage 1 - Optimality Criteria	12
2.1.1 Modifying Top88/Top3d	12
2.2 LS-PrePost Interfacer	12
2.2.1 The Structure of a .k File	13
2.2.2 Reading .k Files	14
2.3 Stage 2 - fmincon	14
2.3.1 Modular Program Structure	14
2.3.2 MATLAB's fmincon Optimization Function	15
2.3.3 Algorithms	16
2.3.4 Convergence Difficulties	17
2.4 Stage 3 - Method of Moving Asymptotes	18
2.4.1 Implementing the Method of Moving Asymptotes Algorithm	18
2.4.2 Displacement-Driven FEA	20
2.5 Program Options for Optimizer 2D and Optimizer 3D	20

	Page
3. DEPENDENCIES AND TRENDS OF MECHANICAL AND GEOMET- RIC ADVANTAGES IN COMPLIANT MECHANISMS	22
3.1 Proposed Seesaw Topology Optimization Problem	22
3.2 Selected Design Solutions for the Seesaw Problem	24
3.2.1 Human Intuition Designs	24
3.2.2 Topology Optimized Designs	24
3.3 Mechanism Advantage Analysis	29
3.3.1 Measuring Mechanical and Geometric Advantages	29
3.3.2 Mechanical Advantage Dependencies and Trends	29
3.3.3 Geometric Advantage Dependencies and Trends	31
3.4 Discussion	33
4. HELMET LINER DESIGN USING A LATTICE STRUCTURE OF POSI- TIVE AND NEGATIVE POISSON RATIO COMPLIANT MECHANISMS	35
4.1 Compliant Mechanism Synthesis	35
4.2 Mechanism Arrangements and Liner Architecture	38
4.3 Numerical Results	39
4.3.1 Testing Setup	40
4.3.2 Model Details	41
4.3.3 Performance Evaluation Criterion	41
4.3.4 Liner Arrangement Analysis	44
4.3.5 Rubber Hardness Analysis	48
4.3.6 Expanded Polypropylene Foam Liner Comparison	51
5. SUMMARY AND FINAL REMARKS	54
5.1 Summary	54
5.2 Limitations	56
5.3 Future Work	56
REFERENCES	58
A. OPTIMIZER PROGRAM DOCUMENTATION	61
A.1 Program Inputs	61
A.1.1 Input File	61
A.1.2 Material Properties	61
A.1.3 Optimization Parameters	62
A.1.4 Poisson's Ratio Problem Parameters	62
A.2 Program Options	63
A.2.1 Select Objective Function	63
A.2.2 Select Algorithm	64
A.2.3 Select Finite Element Method	65
A.2.4 Select Design Variable Initialization	65
A.2.5 Build Full Structure if the Problem is Symmetric?	65
A.2.6 Additional Options	66

	Page
A.3 Program Output	67
A.3.1 Runtime Figures and Command-Line Printouts	67
A.3.2 Saved Files	68
A.4 Creating a Problem Input File in LS-PrePost	69
A.4.1 Geometry Creation	69
A.4.2 Boundary Condition Creation	69
A.4.3 Load Case Creation	71
B. 2D EXAMPLES	72
B.1 Snail Problem - Minimum Compliance, 1 Load Case	72
B.2 Force Inverter Problem - Maximum Mutual Potential Energy	77
C. 3D EXAMPLES	84
C.1 Two Load Case Problem - Minimum Compliance, 2 Load Cases	84
C.2 Positive Poisson's Ratio Mechanism Problem - Maximum Mutual Po- tential Energy	89
D. LS-DYNA MATERIAL MODEL KEYCARD DETAILS	93
D.1 Material Model Parameters for the Helmet Liner	93
E. 3D VERSIONS OF THE POSITIVE AND NEGATIVE MECHANISMS	94
F. OPTIMIZER 2D MAIN MATLAB CODE	95

LIST OF TABLES

Table	Page
1.1 Topology optimization for cantilever beam problem using minimum compliance objective function.	5
1.2 Number of diagnosed concussions and the corresponding percentage of total NFL players. Source: NFL Play Safe Play Smart 2016 Injury Data. .	9
3.1 Optimizer 2D Input Parameters for Seesaw Problem Optimization	25
3.2 Optimized seesaw topologies to show volume fraction constraint masking. .	25
3.3 Optimized Seesaw Topologies using Force-driven and Displacement-driven FEA for various k-ratios ($k_{out} = 0.1$).	27
3.4 Optimized Seesaw Topologies using Force-driven and Displacement-driven FEA for various k-ratios ($k_{in} = 0.1$).	28
3.5 Mechanical Advantages of Human Intuition Designs	29
3.6 Mechanical Advantages of Topology Optimized Designs using Force-Driven and Displacement-Driven FEA for Various k-ratios	30
3.7 Geometric Advantages of Human Intuition Designs	32
3.8 Geometric Advantages of Topology Optimized Designs using Force-Driven and Displacement-Driven FEA for Various k-ratios	32
4.1 Weights for Performance Evaluation Criteria	43
4.2 Mechanism Arrangement Performance Values	46
4.3 Mechanism Arrangement Performance Normalized Values	46
4.4 Shore A Hardness Performance Values	50
4.5 Shore A Hardness Performance Normalized Values	50
4.6 EPP Foam vs. Proposed Design, Performance Comparison	52
D.1 Polycarbonate Shell Material Keycard	93
D.2 Impactor Material Keycard	93
D.3 Natural Rubber Material Keycard	93
D.4 Expanded Polypropylene Foam Material Keycard	93

LIST OF FIGURES

Figure	Page
1.1 Internal structure of open (left) and closed (right) cell foams. Source: Green Compliance Plus [2].	2
1.2 Body center cubic (left) and face center cubic (right) Bravais lattices of the cubic crystal system. Source: Wikipedia, Cubic Crystal System.	2
1.3 Lattice structures with straight-edge design (left) and spherical shell design (right). Source: [3, 4]	3
1.4 Compliant mechanism counterpart (right) of a crimping mechanism (left). Only half of the mechanism is shown due to symmetry. Source: Compliant Mechanisms Research, Brigham Young University.	4
1.5 A force inverter problem (top) solved (bottom) using topology optimization with a maximum MPE objective function.	6
1.6 Traditional football helmet schematic showing EPP foam liner and EVA comfort foam. Source: Silver State 55 helmets.	8
2.1 Example of a .k file. This .k file is for a square 3x3 shell element. Only the geometry has been created so only *ELEMENT_SHELL and *NODE keywords are seen.	13
2.2 Modular Program Structure Flowchart	15
3.1 Simple seesaw with center pivot.	23
3.2 Seesaw problem load and boundary conditions.	23
3.3 Human intuition designs. Top) left pivot seesaw. Middle) center pivot seesaw. Bottom) right pivot seesaw.	24
3.4 Mechanical Advantage vs. k-ratio for force-driven and displacement-driven FEA based topology optimized designs.	31
3.5 Geometric Advantage vs. k-ratio for force-driven and displacement-driven FEA based topology optimized designs.	33
3.6 Mechanical and geometric advantage trends with respect to change in k-ratio.	34
3.7 Mechanical and geometric advantage trends with respect to change in k-ratio.	34

Figure	Page
4.1 a) Positive mechanism and b) negative mechanism load and boundary conditions.	36
4.2 Optimized topology for a) positive mechanism and b) negative mechanism problems. c) Full positive mechanism topology. d) Full negative mechanism topology.	37
4.3 Liner designs. Top) positive arrangement liner. Middle) negative arrangement liner. Bottom) net zero liner.	38
4.4 Simulation of NOCSAE Drop Test Method ND 001 on a basic foam-based helmet to determine width of (top side) impact area.	39
4.5 Initial ballistic test setup for the positive arrangement liner	40
4.6 Acceleration vs. Time curves for positive, negative, and net zero liner arrangements.	44
4.7 Peak internal energy distributions for compliant mechanisms in positive, negative, and net zero liner arrangements.	45
4.8 Deformed liners at peak impactor displacement. Top) positive arrangement liner. Middle) negative arrangement liner. Bottom) net zero liner.	47
4.9 Acceleration vs. Time curves for 50A, 60A, 70A, and 80A rubber hardness liners.	48
4.10 Peak internal energy distributions for compliant mechanisms in 50A, 60A, 70A, and 80A rubber hardness liners.	49
4.11 Acceleration vs. Time curves for the proposed positive 70A liner and an EPP foam liner.	51
4.12 Acceleration vs. displacement simulation responses for EPP foam and compliant mechanisms array liner configurations.	53
5.1 Full EPP foam helmet drop test simulation.	57
A.1 Lines 19-21: Input File Code Segment	61
A.2 Lines 54-57: Material Properties Code Segment	61
A.3 Lines 60-65: Optimization Parameters Code Segment	62
A.4 Lines 68-71: Poisson's Ratio Problem Parameters Code Segment	63
A.5 Graphical Representation of Poisson's Ratio Problem Parameters	63
A.6 Lines 23-28: Select Objective Function Code Segment	64
A.7 Lines 30-38: Select Algorithm Code Segment	64

Figure	Page
A.8 Lines 40-43: Select Finite Element Method Code Segment	65
A.9 Lines 40-43: Select Finite Element Method Code Segment	65
A.10 Lines 50-52: Enabling Symmetrical Problem Build Code Segment	66
A.11 Lines 19-23: Disabling Volume Fraction Constraint for fmincon Algorithms (left: volume fraction constraint enabled, right: volume fraction constraint disabled)	66
A.12 Command-Line Sample Output	68
A.13 LS-PrePost: Shape Mesher Tool Interface	69
A.14 LS-PrePost: Entity Creation Tool Interface - Boundary Creation	70
B.1 Create 60 x 20 2D Mesh	72
B.2 Constrain X and Y Degrees of Freedom on Selected Nodes	73
B.3 Apply Loads in the X direction	74
B.4 Save As snail.k	74
B.5 Initial Command-Line Output for Snail Problem	75
B.6 Termination Command-Line Output for Snail Problem	76
B.7 Topology for Snail Problem	76
B.8 3D STL Iso-View of Snail Problem	77
B.9 Create 40 x 20 2D Mesh	78
B.10 Constrain Y Degrees of Freedom on Nodes Along the Top edge	79
B.11 Apply Inverter Loads in the X direction	80
B.12 Save As forceinverter.k	80
B.13 Initial Command-Line Output for Force Inverter Problem	81
B.14 Termination Command-Line Output for Force Inverter Problem	82
B.15 Topology for Force Inverter Problem	82
B.16 3D STL Iso-View of Force Inverter Problem	83
C.1 Create 60 x 60 x 4 3D Mesh	84
C.2 Constrain X, Y, and Z Degrees of Freedom on Left Face Nodes	85
C.3 Apply Two Opposing Load Cases in Y-Direction	86
C.4 Save As twoload.k	86

Figure	Page
C.5 Initial Command-Line Output for Two Load Case Problem	87
C.6 Termination Command-Line Output for Two Load Case Problem	88
C.7 Topology for Two Load Case Problem	88
C.8 3D STL Iso-View of Two Load Case Problem	89
C.9 Initial Command-Line Output for Positive Poisson's Ratio Mechanism Problem	90
C.10 Termination Command-Line Output for Positive Poisson's Ratio Mecha- nism Problem	91
C.11 Topology for Positive Poisson's Ratio Mechanism Problem	91
C.12 3D STL Iso-View of Positive Poisson's Ratio Mechanism Problem	92
E.1 3D version of positive mechanism.	94
E.2 3D version of negative mechanism.	94

ABBREVIATIONS

BCC:	Body Center Cubic
BFGS:	Broydon-Fletcher-Goldfarb-Shanno method
CTE:	Chronic Traumatic Encephalopathy
EPP:	Expanded Polypropylene
EPS:	Expanded Polystyrene
EVA:	Ethylene-Vinyl Acetate
FCC:	Face Center Cubic
FE:	Finite Element
FEA:	Finite Element Analysis
FEM:	Finite Element Method
GA:	Geometric Advantage
GUI:	Graphical User Interface
HIC:	Head Injury Criterion
IPA:	Interior-Point Algorithm
LSPP:	LS-PrePost
MA:	Mechanical Advantage
MMA:	Method of Moving Asymptotes
MPE:	Mutual Potential Energy
mTBI:	mild Traumatic Brain Injury
NFL:	National Football League
NOCSAE:	National Operating Committee on Standards for Athletic Equip- ment
OC:	Optimality Criterion
PIE:	Peak Internal Energy
SALi:	Shock Absorbing Liquids

SD:	Standard Deviation
SIMP:	Solid Isotropic Material with Penalization
SQP:	Sequential Quadratic Programming
TBI:	Traumatic Brain Injury
TPU:	Thermoplastic Polyurethane
TRRM:	Trust-Region-Reflective Method
VF:	Volume Fraction

ABSTRACT

Najmon, Joel Christian M.S.M.E., Purdue University, December 2017. Design of Compliant Mechanism Lattice Structures for Impact Energy Absorption. Major Professor: Andres Tovar.

Lattice structures have seen increasing use in several industries including automotive, aerospace, and construction. Lattice structures are lightweight and can achieve a wide range of mechanical behaviors through their inherent cellular design. Moreover, the unit cells of lattice structures can easily be meshed and conformed to a wide variety of volumes. Compliant mechanisms make suitable micro-structures for unit cells in lattice structures that are designed for impact energy absorption. The flexibility of compliant mechanisms allows for energy dissipation via straining of the members and also mitigates the effects of impact direction uncertainties.

Density-based topology optimization methods can be used to synthesize compliant mechanisms. To aid with this task, a proposed optimization tool, coded in MATLAB, is created. The program is built on a modular structure and allows for the easy addition of new algorithms and objective functions beyond what is developed in this study. An adjacent investigation is also performed to determine the dependencies and trends of mechanical and geometric advantages of compliant mechanisms. The implications of such are discussed.

The result of this study is a compliant mechanism lattice structure for impact energy absorption. The performance of this structure is analyzed through the application of it in a football helmet. Two types of unit cell compliant mechanisms are synthesized and assembled into three liner configurations. Helmet liners are further developed through a series of ballistic impact analysis simulations to determine the best lattice structure configuration and mechanism rubber hardness. The final liner

is compared with a traditional expanded polypropylene foam liner to appraise the protection capabilities of the proposed lattice structure.

1. INTRODUCTION

1.1 Types of Energy Absorbing Lattice Structures

With the ongoing development of additive manufacturing technologies over the past few decades, cellular lattice structures and other complex designs have seen an increasing rise in application [1]. One of the most prominent uses of cellular lattice structures is for energy absorbing applications. Cellular lattice structures are ideal for this, as they mimic the excellent energy absorbing capabilities of foam, with the prospect for topology optimization resulting in increased performance.

1.1.1 Foams

Polymer foams naturally have a cellular internal structure that easily deforms when compressed, absorbing and dissipating impact energy. In general, foams have two dominant types of internal structures: open cell and closed cell. Open cell foams have internal voids open to the atmosphere and absorb energy through the straining of internal members and displacing of internal air. Closed cell foams have closed internal voids and absorb energy through the straining of internal members and compression of internal air. Figure 1.1 shows the internal structures of open and closed cell foams [2].

1.1.2 Truss-based Lattice Structures

Truss-based lattice structures can be modeled after the internal structure seen in foams. Due to additive manufacturing limitations, modern lattice structures designed for impact energy absorption rely solely on internal member straining and buckling to absorb energy [3–5]. Hence these lattice structures only contain one of the energy absorbing mechanics of foam, however greater control on the design of the unit cell is

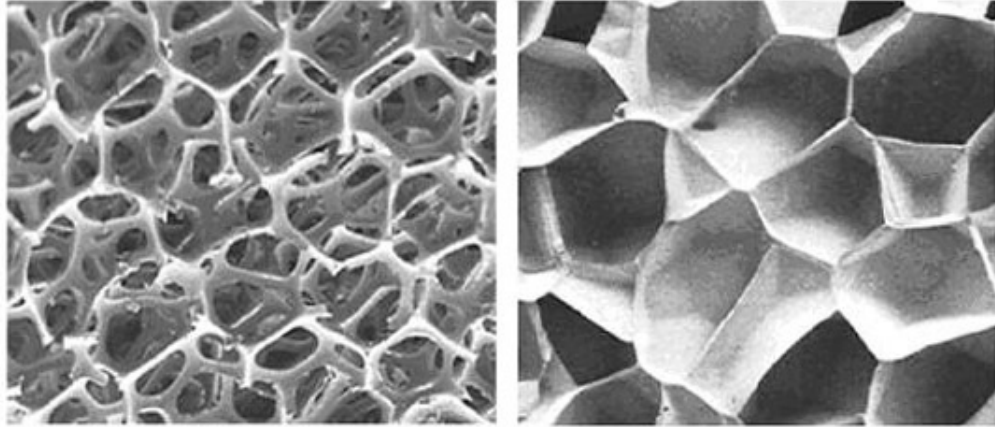


Figure 1.1. Internal structure of open (left) and closed (right) cell foams. Source: Green Compliance Plus [2].

now afforded. This is achieved through trusses that are interconnected within a unit cell. Truss orientation and placement is often in-between lattice points of crystalline cuboids such as body center cubic (BCC) and face centered cubic (FCC), shown in Figure 1.2 [4, 6, 7]. Arrangement of periodic spherical shell geometry has also been studied and follows the typical orientation and placement of truss members [4]. Unit cells are then patterned into the desired shapes using conformal methods [3]. Figure 1.3 shows examples of energy absorbing lattice structures.

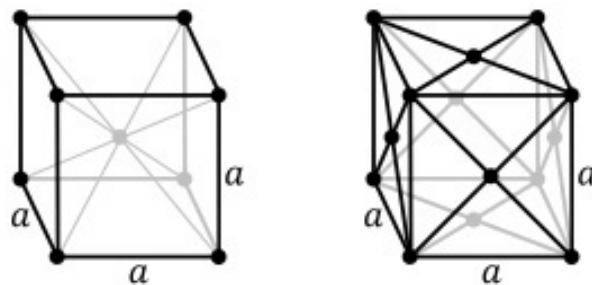


Figure 1.2. Body center cubic (left) and face center cubic (right) Bravais lattices of the cubic crystal system. Source: Wikipedia, Cubic Crystal System.

Unit cells can also be designed to deform with a positive or negative Poisson's ratio [8]. This type of design can be used to impose the straining of truss members through compression or tension forces, ensuring specific strains within the structure. Positive Poisson's ratio unit cells initiate truss buckling, while negative Poisson's ratio unit cells typically abstain from truss buckling [9].

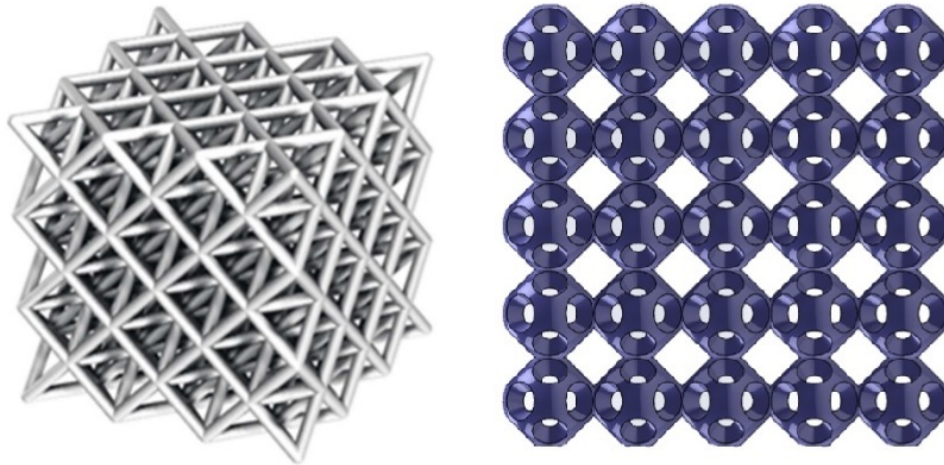


Figure 1.3. Lattice structures with straight-edge design (left) and spherical shell design (right). Source: [3,4]

Lattice structures for impact energy absorption have been studied before, however these lattices consisted of relatively simple topologies inspired by biological designs [10]. Further studies have been conducted involving energy absorption of origami-inspired compliant designs [11]. This study aims to further develop micro-structure design of lattice structures for impact energy absorption.

1.2 Compliant Mechanisms

Compliant mechanisms are mechanisms that transfer their displacement, force, or work through flexible members, rather than using rigid bodies and joints like a traditional mechanism [12]. One outstanding benefit of using compliant mechanisms

over a traditional mechanism is the significant reduction in the number of parts [13]. This is specifically desirable in the application of using compliant mechanisms in a lattice structure, as manufacturing is greatly simplified. Figure 1.4 shows an example of a compliant mechanism counterpart for a crimping mechanism.

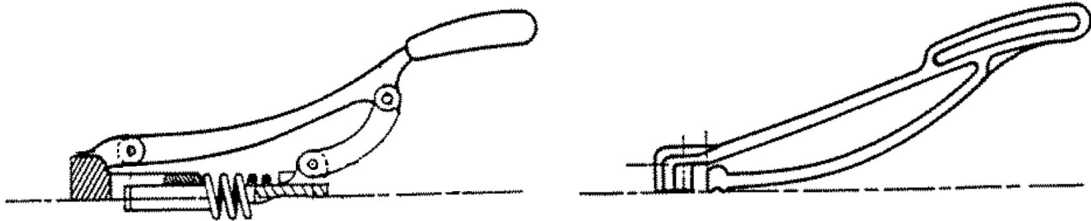


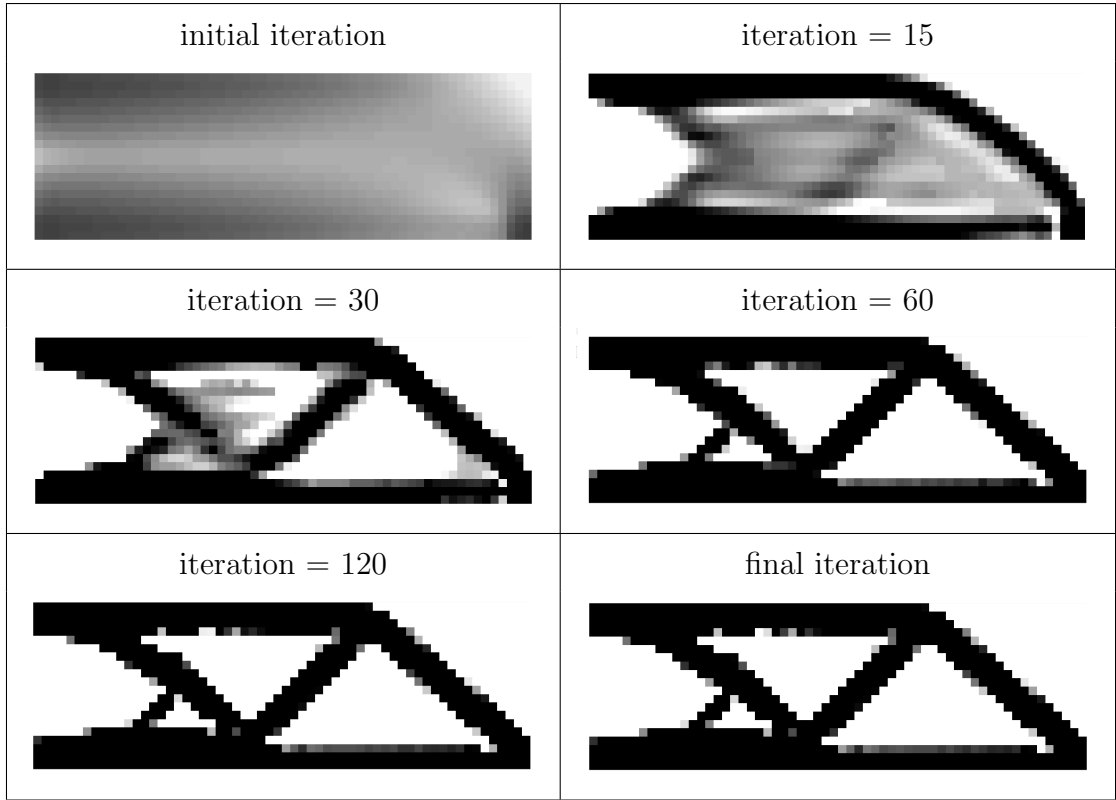
Figure 1.4. Compliant mechanism counterpart (right) of a crimping mechanism (left). Only half of the mechanism is shown due to symmetry. Source: Compliant Mechanisms Research, Brigham Young University.

Not all compliant mechanisms are made entirely of flexible material. Multi-material compliant mechanisms exist that feature rigid material for members and flexible material at connecting points [13, 14]. Flexible materials serve as the joint of the rigid-body members, similar to a traditional mechanism. This method of compliant mechanism synthesis preserves some of the output accuracy that is seen in traditional mechanisms at the cost of versatility for uncertainty in loads.

1.2.1 Maximum Mutual Potential Energy Objective Function for Topology Optimization

Density-based topology optimization is a popular method for structural design as it removes design biases associated with human intuition designs [15–17]. Densities of a given design domain are allocated to create an optimized design that minimizes or maximizes an objective function. Table 1.1 shows density allocation over optimization iterations to achieve an optimal topology for minimum displacement.

Table 1.1. Topology optimization for cantilever beam problem using minimum compliance objective function.



For compliant mechanism synthesis, a popular optimization objective is to maximize the mutual potential energy (MPE) of the structure [18]. This type of formulation has specified input and dummy output loads, with added corresponding external stiffnesses to model force actuators. The MPE is the displacement of the dummy output loaded nodes [19] and is given by

$$\text{MPE} = \mathbf{U}_2(\tilde{\mathbf{x}})^T \mathbf{K}(\tilde{\mathbf{x}}) \mathbf{U}_1(\tilde{\mathbf{x}}) \quad (1.1)$$

where $\tilde{\mathbf{x}}$ is the filtered design densities, \mathbf{U}_2 is the global displacement vector resulting from the dummy output load, \mathbf{U}_1 is the global displacement vector resulting from the input load, and \mathbf{K} is the global stiffness matrix. Optimization is usually done subject to a volume constraint of

$$v(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^T \mathbf{v} - \bar{v} \leq 0 \quad (1.2)$$

$$\mathbf{x} \in \mathcal{X}, \quad \mathcal{X} = \mathbf{x} \in \mathbb{R}^n : \mathbf{0} \leq \mathbf{x} \leq \mathbf{1} \quad (1.3)$$

where \mathbf{v} is element volume vector. Figure 1.5 shows an example force inverter problem solved using topology optimization with a maximum MPE objective function.

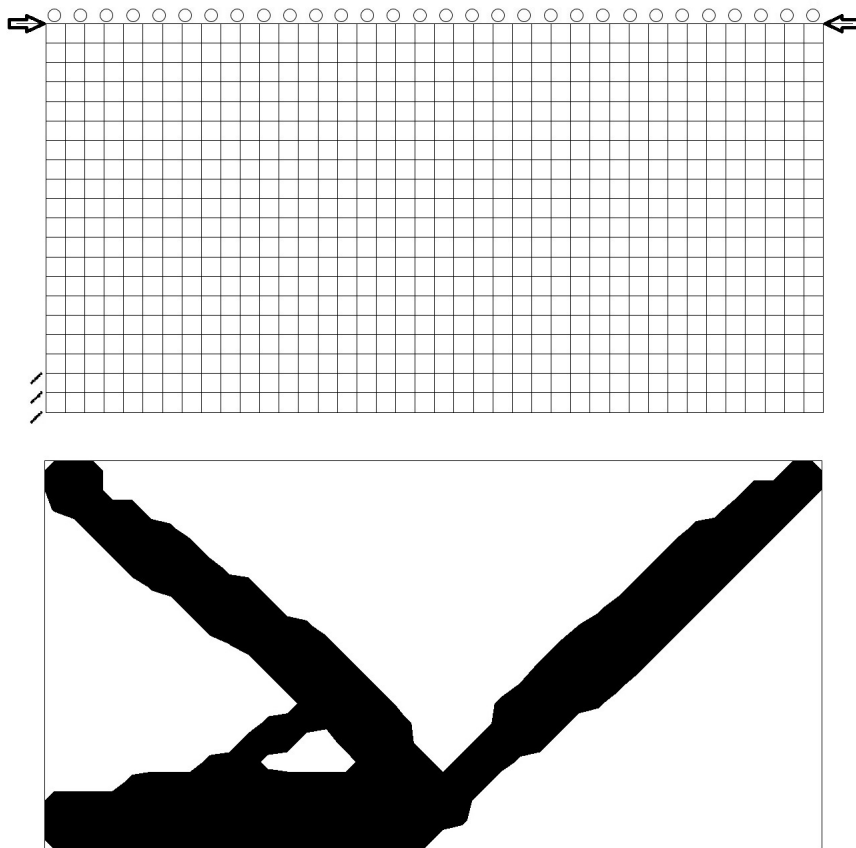


Figure 1.5. A force inverter problem (top) solved (bottom) using topology optimization with a maximum MPE objective function.

1.2.2 Mechanical and Geometric Advantage

Another characteristic of mechanisms is their mechanical or geometric advantage. The mechanical advantage (MA) is the ratio of output force to input force [20, 21] and is given by

$$\text{MA} = \frac{F_{out}}{F_{in}}. \quad (1.4)$$

The geometric advantage (GA) is the ratio of output displacement to input displacement [21] and is given by

$$\text{GA} = \frac{u_{out}}{u_{in}}. \quad (1.5)$$

1.3 Compliant Mechanisms for Energy Absorption

While most mechanism applications are to transfer displacement, force, or energy to an output port, compliant mechanisms can be utilized for impact energy absorption. The flexibility of compliant mechanisms affords this ability through the straining and buckling of its members.

Furthermore, compliant mechanisms can be tailored to behave with certain Poisson's ratios when compressed. Due to the inherent ambiguity of flexible members, it is challenging to optimize a compliant mechanism that satisfies a specific Poisson ratio, however an unspecified positive or negative Poisson ratio mechanism can still be achieved. These mechanisms can be arranged into lattice structures to absorb impact energy through the straining and buckling of internal flexible members through compressive and tensile forces between neighboring mechanisms.

1.4 Existing Helmet Liners

In this study, the developed lattice structure is used in the application of a sports helmet. The most commonly seen reusable sports liner is crushable expanded polypropylene (EPP) foam. For one-time-impact helmets, such as bike and horse riding helmets, a crushable damage-retaining, foam liner is used, typically expanded polystyrene (EPS). Other less common liners use thermoplastic polyurethane (TPU), shock absorbing liquids (SALi), air bladders, buckling columns, or non-Newtonian fluids [22]. The liner developed in this study aims at replacing a non-crushable foam EPP liner (football, baseball, lacrosse, etc). Figure 1.6 shows a schematic of the different protective components of a helmet including the ethylene-vinyl acetate (EVA) comfort foam.



Figure 1.6. Traditional football helmet schematic showing EPP foam liner and EVA comfort foam. Source: Silver State 55 helmets.

1.5 Motivation and Objectives

1.5.1 Motivation

In the recent years, the short and long term effects of sports related head injuries, like traumatic brain injuries (TBI) and mild traumatic brain injuries (mTBI), more commonly referred to as concussions, has become apparent. A recent study found that 99% of the brains of deceased National Football League (NFL) players (87% across all levels of play), were found to have chronic traumatic encephalopathy (CTE), which has been associated with memory and mood impairments and dementia [23]. Furthermore, athletes who received head injuries in the past are more susceptible to mTBIs, intensifying their effects [24]. Table 1.2 shows the number of diagnosed concussions and the corresponding percentage of total players in the NFL. These facts show the inadequacies in traditional approaches to helmet design, and call for disruptive technologies to escalate helmet safety and alleviate doubts behind head injury prone sports. This can be achieved by using topology optimized compliant mechanism lattice structures for helmet liners.

Table 1.2. Number of diagnosed concussions and the corresponding percentage of total NFL players. Source: NFL Play Safe Play Smart 2016 Injury Data.

Year	# of Diagnosed Concussions	% of total players (1696)
2012	261	15.4%
2013	229	13.5%
2014	206	12.1%
2015	275	16.2%
2016	244	14.4%

1.5.2 Objectives

The objective of this work can be characterized by three parts with a corresponding chapter for each part. The first part is the development of a compliant mechanism optimization tool. Ideally, the tool should be easy to use and have the adaptability for use with other density-based topology optimization objective functions to promote future academic development and learning.

The second contribution of this work is an investigation into the dependencies and trends behind mechanical and geometric advantages of compliant mechanism. This investigation was done to determine if resulting compliant mechanism topologies, optimized with certain parameters, are more likely to favor certain advantages. The investigation concludes with a discussion of the results and the implications they have for compliant mechanism synthesis.

The last part of this work is the development and design of compliant mechanism lattice structures for use in a helmet liner. The liner design starts with the synthesis of two types of compliant mechanisms. The first mechanism operates with a positive Poisson's ratio, where the Poisson's ratio, in this case, is defined as the displacement ratio between the input and output ports of the compliant mechanism. The second mechanism operates with a negative Poisson's ratio. These different mechanisms are assembled into three different liner arrangements and analyzed through a series of tests. The final liner design is compared against an EPP foam liner to appraise the protection capabilities of the proposed liner.

2. PROPOSED DESIGN OPTIMIZATION TOOL

The development of this compliant mechanism optimization code was done exclusively in MATLAB. While other more efficient high-level languages exist, MATLAB was chosen for its convenient and easy-to-use interface; ideal for development. Modifying existing topology optimization programs was straightforward, and was done in a manner that will hopefully encourage and facilitate future modification and development.

The development of the final code can be broken up into three stages each bringing a significant new feature to the program. Stages are defined by a new algorithm that was brought into the program, however other features were also added during these stages. The first stage of the code used Optimality Criteria (OC). In the next stage, several more algorithms are added through MATLAB's `fmincon` function. The third stage of the code, introduces the method of moving asymptotes (MMA) algorithm [25].

All of the features that were successfully implemented in the code, were kept and exist in the final version of the code. These features will be documented as they were added and summarized into a comprehensive list. Code development occurred first in the 2D variant of the code (Optimizer 2D). Once the new feature was working in Optimizer 2D, the 3D variant of the code (Optimizer 3D) was updated with a corresponding feature.

Consequently, both programs work the same way, and have very similar structures and build histories. For the sake of simplicity, descriptions of build stages will pertain directly to Optimizer 2D, however details that are only applicable to the Optimizer 3D program will still be expressed.

2.1 Stage 1 - Optimality Criteria

The first build of the code was created by modifying the existing topology optimization MATLAB programs top88 and top3d [18,26]. Import code was also added to read LS-PrePost's (LSPP) .k files containing geometry, loads, and boundary conditions of the optimization problem. This change allowed the user to set up the optimization problem with a graphical user interface (GUI), rather than manually entering nodal loads and supports.

2.1.1 Modifying Top88/Top3d

Top88 was modified to create a 2D compliant mechanism optimizer and top3d was modified to create a 3D counterpart [18,26]. This build kept the same optimization algorithm, OC, that the original two programs had. Modifications that were made followed those detailed in the corresponding paper of the top3d code [18].

The first of these was switching out the minimum compliance objective function for the maximum MPE objective function. Then the finite element (FE) code was modified to include two of the forces, an input load and an output (dummy) load. An external spring stiffness is also added to the output degrees of freedom.

2.2 LS-PrePost Interfacer

LS-PrePost was chosen as a complimentary GUI for the program for several reasons. 1) LSPP is free to download and free to use. 2) LSPP has all of the necessary modeling features needed for the program (geometry creation, meshing, loads, boundary conditions), and more. 3) Future editions of the code may use LS-PrePost's accompanying solver, LS-Dyna, to perform finite element analysis (FEA). This would result in longer solving time, at the gain of using LS-Dyna's robust non-linear solver.

At the time of writing, the interfacier only reads the problem data and interprets this into data that the MATLAB program can use. Due to the structure of the

FE solver in the program, the current version of the program only supports simple rectangular/cuboid shapes.

2.2.1 The Structure of a .k File

Modeling files are created in LSPP via keywords. As features of the model are created corresponding keywords are added to the .k file. Following the unique keyword headers are columns of data that define features and properties of that keyword. The location and order of this data is how LSPP and the interface code assign meaning to the numbers. An example of keyword text is shown in Figure 2.1.

```

$# LS-DYNA Keyword file created by LS-PrePost(R) V4.3 - 28Dec2016(09:00)
$# Created on Sep-17-2017 (15:26:11)
*KEYWORD
*PART
$#
shell_4p
$# pid secid mid eosid hgid grav adpopt tmid title
    1 0 0 0 0 0 0 0
*ELEMENT_SHELL
$# eid pid n1 n2 n3 n4 n5 n6 n7 n8
    1 1 5 6 2 1 0 0 0 0
    2 1 6 7 3 2 0 0 0 0
    3 1 7 8 4 3 0 0 0 0
    4 1 9 10 6 5 0 0 0 0
    5 1 10 11 7 6 0 0 0 0
    6 1 11 12 8 7 0 0 0 0
    7 1 13 14 10 9 0 0 0 0
    8 1 14 15 11 10 0 0 0 0
    9 1 15 16 12 11 0 0 0 0
*NODE
$# nid x y z tc rc
    1 0.0 0.0 0.0 0 0
    2 0.0 1.0 0.0 0 0
    3 0.0 2.0 0.0 0 0
    4 0.0 3.0 0.0 0 0
    5 1.0 0.0 0.0 0 0
    6 1.0 1.0 0.0 0 0
    7 1.0 2.0 0.0 0 0
    8 1.0 3.0 0.0 0 0
    9 2.0 0.0 0.0 0 0
   10 2.0 1.0 0.0 0 0
   11 2.0 2.0 0.0 0 0
   12 2.0 3.0 0.0 0 0
   13 3.0 0.0 0.0 0 0
   14 3.0 1.0 0.0 0 0
   15 3.0 2.0 0.0 0 0
   16 3.0 3.0 0.0 0 0
*END

```

Figure 2.1. Example of a .k file. This .k file is for a square 3x3 shell element. Only the geometry has been created so only *ELEMENT_SHELL and *NODE keywords are seen.

2.2.2 Reading .k Files

Data is read from .k files by searching the file for the unique keyword header. After the header is located the program reads the following rows and columns and breaks up the data into a corresponding array, which is further interpreted into numerical data for MATLAB.

2.3 Stage 2 - fmincon

The second build of the code added MATLAB's constrained nonlinear multi-variable function optimizer, `fmincon` [27]. Additional functionality was also added by converting the code to a modular format. This allows the optimizer program to solve optimization problems with different objectives by calling different functions (minimum compliance, maximum MPE, heat conduction, and other user-created functions).

Additional objective functions for minimum compliance (one or two loads) were created to test the robustness of the program and to aid with trouble shooting. Examples of using the optimizer with minimum compliance problems can be found in Appendix B and C.

2.3.1 Modular Program Structure

The main program file is responsible for calling all sub-functions to run. This is where the input .k file is specified. Instead of directly passing data between functions, data is saved to a temporary file on the hard drive. Sub-functions read and append only the variables that are needed for their function. This allows the code to run on computers with small amounts of ram, or allows higher-fidelity code to run on an equivalent computer. This method trades run-time for greater versatility. Figure 2.2 shows a flowchart of the program's structure.

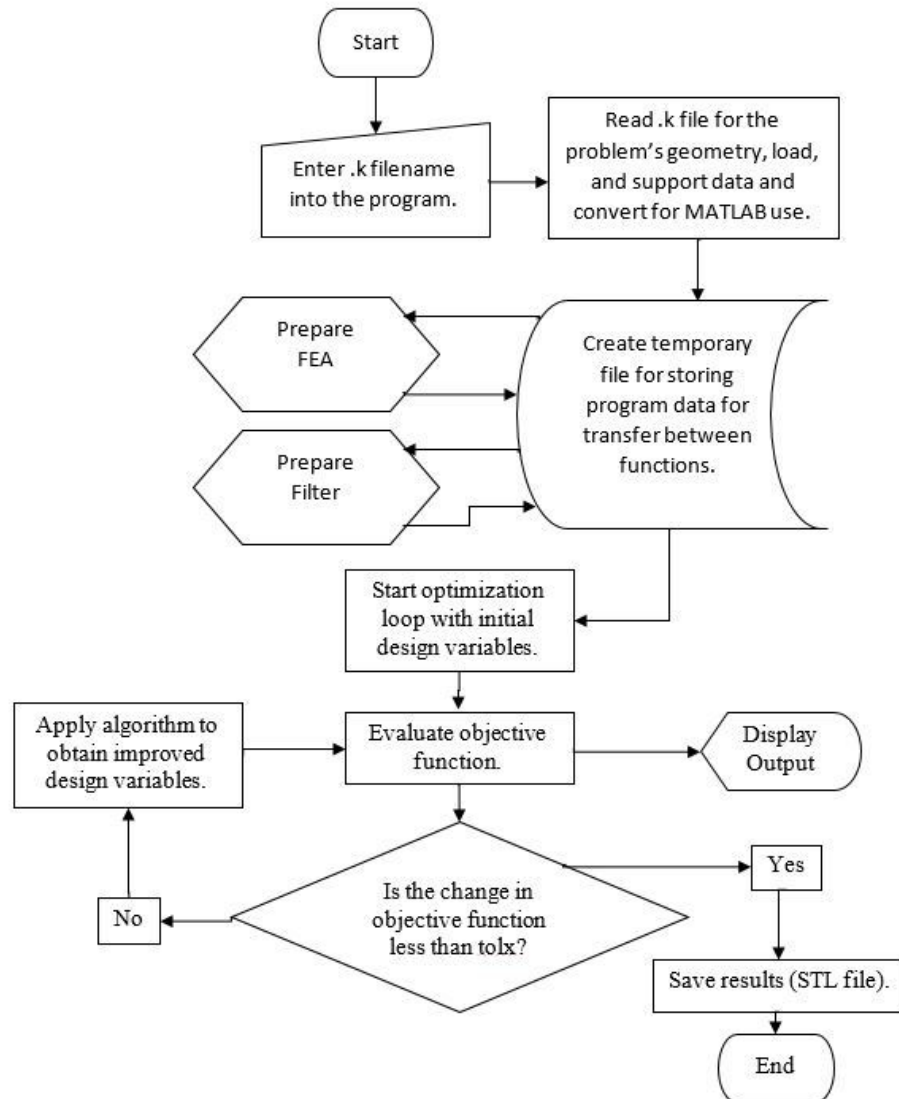


Figure 2.2. Modular Program Structure Flowchart

2.3.2 MATLAB's fmincon Optimization Function

Fmincon is a nonlinear programming solver with constraints. It finds the minimum of a problem specified by

$$\text{Minimize } f(\mathbf{x}) \text{ such that: } \begin{cases} c(\mathbf{x}) \leq 0 \\ ceq(\mathbf{x}) = 0 \\ \mathbf{A} \cdot \mathbf{x} \leq \mathbf{b} \\ \mathbf{A}eq \cdot \mathbf{x} = beq \\ \mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub} \end{cases} \quad (2.1)$$

where $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$ is an inequality constraint, $\mathbf{A}eq \cdot \mathbf{x} = beq$ is an equality constraint, and \mathbf{lb} and \mathbf{ub} are lower and upper bounds respectively [27].

The initial variables, objective function's value, c , the derivative of the objective function's value, dc/dx , optimization options, and linear and nonlinear constraints are specified when `fmincon` is called.

For binary density-based topology optimization, the variable constraints are:

$$\mathbf{x} \in \mathcal{X}, \mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{0} \leq \mathbf{x} \leq \mathbf{1}\} \quad (2.2)$$

$$\mathbf{xL} = nv \quad (2.3)$$

where, $\mathbf{x} = [x_1, x_2, \dots, x_n]$, is the densities of the elements, $\mathbf{L} = [1, 1, \dots, 1]^T$ is a vector of ones, n is the number of elements, and v is the volume fraction. The linear volume fraction constraint (2.3) is optional for compliant mechanism synthesis, however it is often included to limit material use and produce more aesthetically pleasing designs.

2.3.3 Algorithms

`Fmincon` can be run with five different algorithms. These algorithms include interior-point algorithm (IPA), trust-region-reflective method (TRRM), sequential quadratic programming (SQP), SQP-legacy, and active-set. For this study IPA and TRRM were primarily used.

Hessian approximation is not user-supplied and calculated by `fmincon` using the Broydon - Fletcher - Goldfarb - Shanno (BFGS) method [27]. It is worth noting that

Optimizer 2D and Optimizer 3D have not been tested extensively with all combinations of algorithms, objective functions, FE solvers, and problems, so some combinations might prove to be ill-suited for the applied problem. Recommendations for algorithms and other program options are given in the documentation of the code, Appendix A.

2.3.4 Convergence Difficulties

The IPA reconstructs the objective function incorporating linear constraints within the function. This produces a barrier function which leads to a vertical asymptote at $x = 0$, where the objective function cannot be driven past with relatively small changes in x . This effectively creates the constraint $c(\mathbf{x}) \geq 0$, when the algorithm's stepsize is sufficiently small.

Due to this behavior, there can be convergence difficulties when optimizing for maximum MPE for some compliant mechanism problems (cases where the negative objective function is minimized). In most of these cases, the objective function is initially a small positive number. As the program minimizes this value, the function reaches an asymptote at $f(\mathbf{x}) = 0$. Eventually, the change in the objective function becomes small enough to meet the convergence criterion and the optimizer concludes with a premature solution.

To address this problem, the initial densities were randomized. This method varies the starting point (amongst different attempts) of the function allowing some attempts to have their objective function's value move past zero and continue to minimize the function (or maximize the negative MPE in the case of compliant mechanism synthesis). Another approach is to lower the solution's convergence tolerance, which allows the change for further iterations to go beyond the asymptote.

This solution did not work for all compliant mechanism problems. The SQP, SQP-legacy, and active-set algorithms also produced similar convergence difficulties as the

IPA. Therefore, the trust-region-reflective method was selected as it's algorithm was sufficiently different from the others to not incur the same difficulties.

While TRRM was able to successfully solve complaint mechanism optimization problems, it had several disadvantages prompting to the implementation of MMA. The two most prominent drawbacks to TRRM are lengthy run-times and the inability to enforce linear constraints and boundary constraints at the same time, leading to the removal of the volume constraint.

2.4 Stage 3 - Method of Moving Asymptotes

In the third build of the code, MATLAB code for the Method of Moving Asymptotes written by Krister Svanberg was added as an algorithm option [25]. MMA was implemented to address convergence difficulties of using IPA and has several benefits over TRRM.

Since MMA, approximates the objective function by generating and solving a sub-problem, convergence problems caused by using a minimization solver with a maximization optimization problem (as was the case using IPA) are avoided. [25]. MMA was developed for structural optimization problems, hence it can efficiently evaluate computationally expensive objective functions. The method also is not limited to one constraint like TRRM, allowing the application of boundary constraints and volume fraction limits simultaneously [25].

2.4.1 Implementing the Method of Moving Asymptotes Algorithm

The MMA MATLAB code (mmasub and subsolve) by Krister Svanberg finds the minimum of a problem specified by

$$\text{find } \mathbf{x} \tag{2.4}$$

$$\text{minimize} \quad - \sum_{i=1}^n \left[\frac{(x_i^{(k)} - L_i^{(k)})^2}{x_i - L_i^{(k)}} \frac{\partial c}{\partial x_i} (\tilde{\mathbf{x}}^{(k)}) \right] \quad (2.5)$$

$$\text{subject to} \quad \tilde{\mathbf{x}}^T \mathbf{v} - \bar{v} \leq 0, \quad \mathbf{x} \in \mathcal{X}^{(k)} \quad (2.6)$$

$$\text{where} \quad \mathcal{X} = \mathbf{x} \in \mathcal{X} \mid 0.9L_i^{(k)} + 0.1x_i^{(k)} \leq x_i \leq 0.9U_i^{(k)} + 0.1x_i^{(k)}, i = 1, \dots, n. \quad (2.7)$$

$L_i^{(k)}$ and $U_i^{(k)}$ are the lower and upper asymptotes respectively and are updated with the following heuristic rule: [18, 25]

$$\text{when } k=1 \text{ and } k=2: \quad U_i^{(k)} + L_i^{(k)} = 2x_i^{(k)}, \quad U_i^{(k)} + L_i^{(k)} = 1 \quad (2.8)$$

$$\text{when } k \geq 3: \quad U_i^{(k)} + L_i^{(k)} = 2x_i^{(k)}, \quad U_i^{(k)} + L_i^{(k)} = \gamma_i^{(k)}. \quad (2.9)$$

$$\gamma_i^{(k)} = \begin{cases} 0.7, & (x_i^{(k)} - x_i^{(k-1)})(x_i^{(k-1)} - x_i^{(k-2)}) \leq 0 \\ 1.2, & (x_i^{(k)} - x_i^{(k-1)})(x_i^{(k-1)} - x_i^{(k-2)}) \geq 0 \\ 1.0, & (x_i^{(k)} - x_i^{(k-1)})(x_i^{(k-1)} - x_i^{(k-2)}) = 0 \end{cases} \quad (2.10)$$

Using a similar structure to the OC algorithm, an optimization while loop, which evaluates the objective function, calls `mmsub` and updates design variables for future iterations. The while loop is run until the change in the design is within the specified tolerance or the maximum number of loops is reached.

2.4.2 Displacement-Driven FEA

Additional functionality was also introduced through the added option of using a displacement-driven finite element method (FEM) over the traditional force-driven FEM. Enabling this option turns the input loads into input displacements (instead of input forces) and the output dummy loads to output dummy displacements (instead of output dummy forces). The displacement-driven FE solver works through the following steps: 1) Apply $\mathbf{U}_{external}$. 2) Find $\mathbf{F}_{external}$ that minimizes the 2-norm between $\mathbf{U}_{external}^* = \mathbf{K}^{-1}\mathbf{F}_{external}$ and $\mathbf{U}_{external}$. 3) Solve for $\mathbf{U}_{internal} = \mathbf{K}^{-1}\mathbf{F}_{external}$. For minimum compliance problems, this change has no effect on the resulting topology. However, for MPE problems, this option was found to have some effect on the resulting topologies of the optimizer. Essentially, the force-driven FE solver controls deformation force while the displacement-driven FE solver controls deformation displacement. The details behind the affects of the FE solver on MA and GA are further studied in the next chapter.

2.5 Program Options for Optimizer 2D and Optimizer 3D

The final program solves optimization problems in 2D and 3D allowing the user to choose options for the objective function, algorithm, FE solver, and variable initialization. The following is a comprehensive list of options for Optimizer 2D and Optimizer 3D.

Objective function options:

- Minimum Compliance, 1 Load
- Minimum Compliance, 2 Load
- Maximum Mutual Potential Energy

Algorithm options:

- Optimality Criteria
- Interior-Point

- Sequential Quadratic Programming
- Sequential Quadratic Programming - Legacy
- Trust-Region-Reflective
- Active-Set
- Method of Moving Asymptotes

Finite element solver options:

- Force-Driven Finite Element Analysis
- Displacement-Driven Finite Element Analysis

Variable initialization options:

- Volume Fraction Initialization
- Random Initialization

Full program documentation is given in Appendix A, with example problems in 2D and 3D given in Appendix B and C respectively. See Appendix F for the main file of the Optimizer 2D program and details on how to obtain the full code.

3. DEPENDENCIES AND TRENDS OF MECHANICAL AND GEOMETRIC ADVANTAGES IN COMPLIANT MECHANISMS

In this chapter, a study is carried out to determine the dependencies and trends of the MA and GA of compliant mechanisms. A topology optimization problem is proposed for the role of comparing topology solutions that were optimized with varying parameters. For reference, simple designs based on human intuition are also compared. The type of finite element solver (force-driven or displacement-driven) was chosen as a variable due to the difference between input and output load distinction. The second variable that was tested is the external load stiffness that is added to input and output ports of the mechanism. This variable was chosen for its correspondence with the symmetry of the problem. The effects of these parameters on resulting topology and the MA and GA are discussed. Results and conclusions of the study are also discussed.

3.1 Proposed Seesaw Topology Optimization Problem

The proposed topology optimization for use in this study was inspired by a simple seesaw, shown in Figure 3.1. This problem was chosen as it provides a simple illustration of the MA and GA dependencies on the topology of the problem. This illustration is created by specifying the left end of the seesaw as a downward facing input, the right end of the seesaw as an upward facing output, and by moving the pivot point.

For example, when the pivot point is in the center the MA and GA are 1 (for a rigid bar). The MA and GA can be directly changed by shifting the pivot point to the left or right.



Figure 3.1. Simple seesaw with center pivot.

Adapting this setup into an initial topology optimization problem is done by fixing the bottom of a 2:1 rectangle and specifying input and output loads in the middle of the left and right sides respectively. Figure 3.2 shows the loaded nodes and supports of the proposed design problem. All designs were optimized and analyzed with an 80 x 40 mesh.

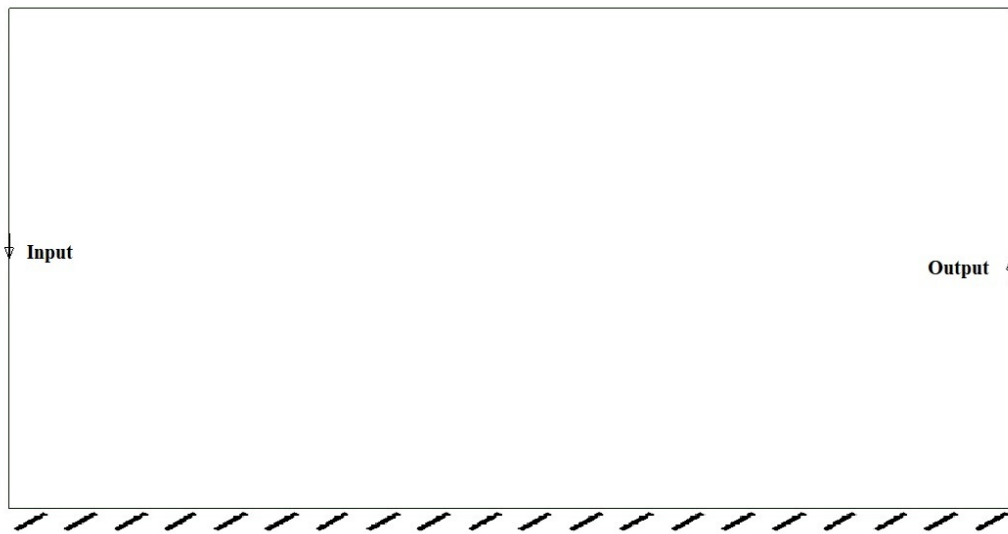


Figure 3.2. Seesaw problem load and boundary conditions.

3.2 Selected Design Solutions for the Seesaw Problem

3.2.1 Human Intuition Designs

A design based on human intuition was tested to gain perspective on the optimization tool's ability to optimize designs with improved MA and GA compared to traditional designs. The tested human intuition design has a seesaw-shaped topology as seen in Figure 3.1. Three designs are analyzed with pivot points located on the left, center, and right ends of the bar, shown in Figure 3.3

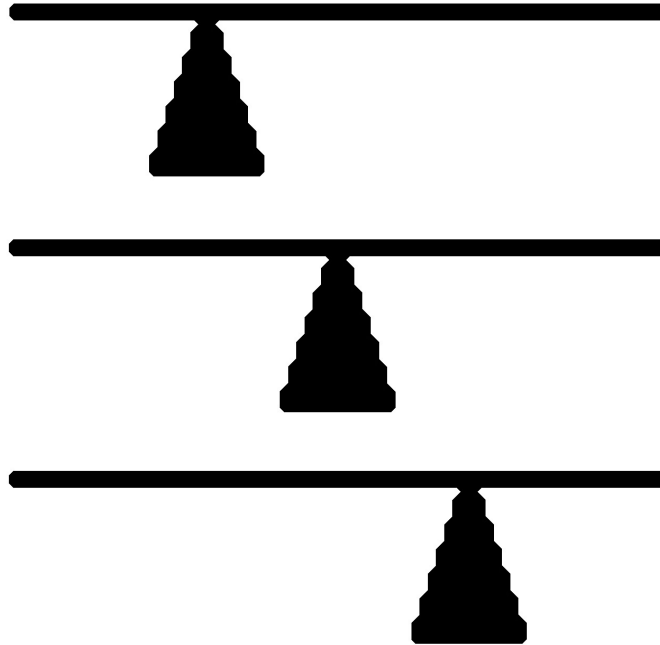


Figure 3.3. Human intuition designs. Top) left pivot seesaw. Middle) center pivot seesaw. Bottom) right pivot seesaw.

3.2.2 Topology Optimized Designs





All topology optimized compliant mechanism designs were synthesized with the Optimizer 2D program using the input parameters located in Table 3.1. The vol-

ume fraction (VF) constraint was removed as non-essential constraints can mask or hide MA or GA tailored topology. Table 3.2 shows the masking affect that the VF constraint can have on topologies.

Table 3.1. Optimizer 2D Input Parameters for Seesaw Problem Optimization

Objective Function:	Maximum MPE
Algorithm:	Optimality Criteria
Volume Fraction:	Removed
Density Initialization:	0.5
penalization:	3
minimum radius:	3

Table 3.2. Optimized seesaw topologies to show volume fraction constraint masking.

VF	Force-Driven FEA	Displacement-Driven FEA
VF = 0.3		
No VF		

Optimizer 2D allows the use of a traditional force-driven FEA solver or a displacement-driven FEA solver. This is the first parameter that was changed when generating optimized designs. The second parameter that was changed is the external stiffness added to loaded nodes. This can be visualized as an extra spring element added to

the input and output nodes and is only applicable in the applied force's direction. This spring is added to model the force actuator. These added external stiffness are related by the k-ratio which is given by

$$\text{k-ratio} = \frac{k_{out}}{k_{in}} \quad | \quad k_{in} \neq 0 \quad (3.1)$$

where k_{in} is the external stiffness added to the input node and k_{out} is the external stiffness added to the output node. Generally the k-ratio is 0.1/0.1. This typically results in symmetrical designs, however unequal external stiffnesses can be applied to obtain non-symmetrical topologies. Tables 3.3 and 3.4 shows topologies optimized with force-driven and displacement-driven FEA for varying k-ratios. To prevent extremely stiff structures, one of the external stiffnesses was always maintained at 0.1, while varying the other external stiffnesses.

Observing the topology results, there are a couple initial observations that can be made. First, force-driven FEA and displacement-driven FEA produce vastly different topologies (when not limited by a volume fraction). Displacement-driven FE solved topologies show significant changes as the k-ratio is changed. Conversely, force-driven FEA solved topologies show some morphing, but overall maintain a similar structure.

As expected, when the k-ratio is 1:1, the topology is symmetrical. When the k-ratio is not 1:1 the topology becomes skewed towards the side with the larger added stiffness. Surprisingly, when using the displacement-driven FE solver for large disparities between the added stiffnesses the topology morphs into a simple seesaw shape. Another interesting observation is the 6-bar mechanism that results from using the force-driven FE solver.

Table 3.3. Optimized Seesaw Topologies using Force-driven and Displacement-driven FEA for various k-ratios ($k_{out} = 0.1$).











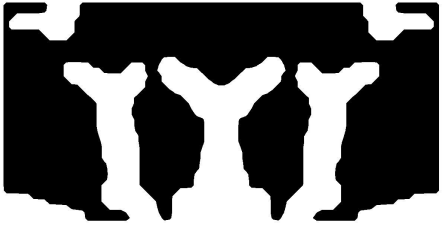









k-ratio	Force-Driven FEA	Displacement-Driven FEA
$\frac{0.1}{10} = 0.01$		
$\frac{0.1}{5} = 0.02$		
$\frac{0.1}{1} = 0.1$		
$\frac{0.1}{0.5} = 0.2$		
$\frac{0.1}{0.1} = 1$		

Table 3.4. Optimized Seesaw Topologies using Force-driven and Displacement-driven FEA for various k-ratios ($k_{in} = 0.1$).

k-ratio	Force-Driven FEA	Displacement-Driven FEA
$\frac{0.1}{0.1} = 1$		
$\frac{0.5}{0.1} = 5$		
$\frac{1}{0.1} = 10$		
$\frac{5}{0.1} = 50$		
$\frac{10}{0.1} = 100$		

3.3 Mechanism Advantage Analysis

3.3.1 Measuring Mechanical and Geometric Advantages

The MA is measured by removing the output load and constraining the output load's degree of freedom. For this case, the right output load is removed and replaced with a constraint in the y-direction. A single iteration of FEA is performed on the design, solving for the reaction force on the added constraint. This reaction force can be divided by the input force to obtain the MA of the compliant mechanism.

Measuring GA is straightforward. The loads and boundary conditions are maintained from the topology optimization. FEA is performed to solve for the input and output displacements. These are divided to obtain the GA of the compliant mechanism.

3.3.2 Mechanical Advantage Dependencies and Trends

The MA for the human intuition and topology optimized designs are given in Tables 3.5 and 3.6. To better visualize the effects of the FE solver and k-ratio on MA, Figure 3.4 plots the MA vs. the k-ratio for the force-driven and displacement-driven FE solvers.

Table 3.5. Mechanical Advantages of Human Intuition Designs

Human Intuition Design	Mechanical Advantage
Left Pivot Seesaw	6.58×10^{-5}
Center Pivot Seesaw	4.86×10^{-5}
Right Pivot Seesaw	6.59×10^{-5}

The human intuition designs, or simple seesaw designs, have very bad mechanical advantage when compared with the topology optimized designs. This poor performance is because the rigid bar and joint mechanism of a traditional seesaw was directly

Table 3.6. Mechanical Advantages of Topology Optimized Designs using Force-Driven and Displacement-Driven FEA for Various k-ratios

k-ratio	Force-Driven FEA	Displacement-Driven FEA
0.01	0.280	0.035
0.02	0.281	0.049
0.1	0.297	0.101
0.2	0.313	0.144
1.0	0.390	0.354
5.0	0.463	0.338
10.0	0.472	0.270
50.0	0.480	0.141
100.0	0.479	0.098

translated to a flexible compliant mechanism counterpart. The bar does not have the rigidity to effectively transfer force from one end to the next.

The topology optimized designs have significantly better MA than the human intuition designs. Observing the MA of the force-driven FE solved designs, an increase in the MA is seen when shifting the external stiffness from the input to the output, with the greatest increase occurring at a k-ratio of 1. On the other hand, the displacement-driven FE solved designs, see a maximum at a k-ratio of 1 and subsequently decreases with increasing external stiffness difference. It is worth noting that at every k-ratio tested, the force-driven FEA topology had higher MA.

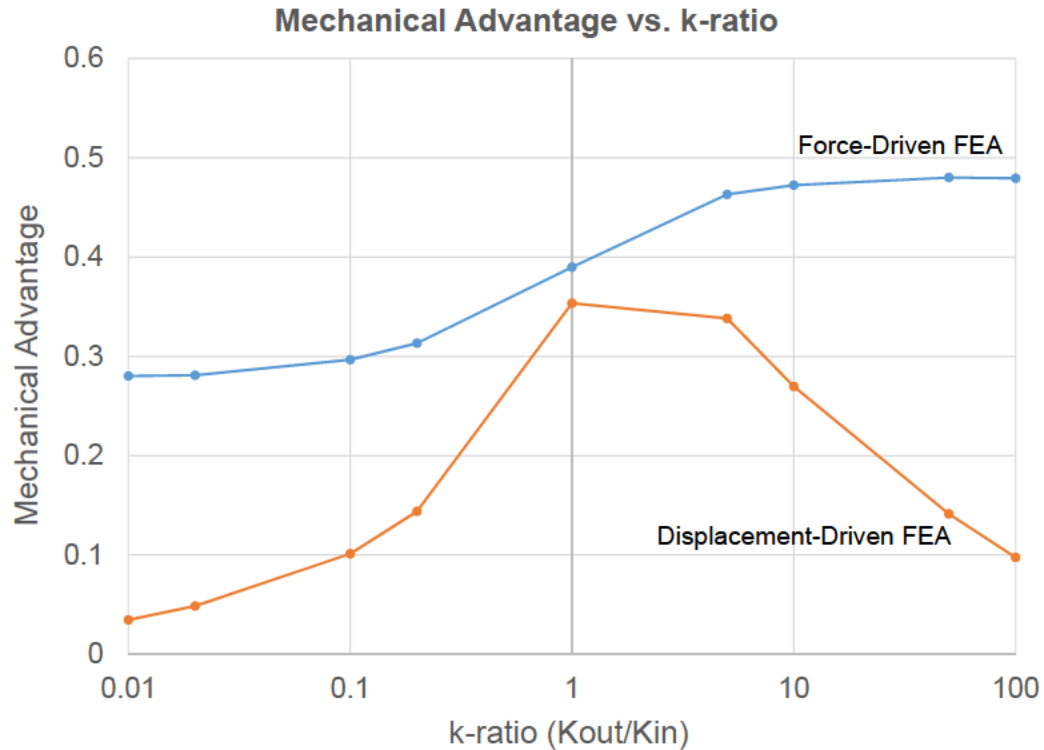


Figure 3.4. Mechanical Advantage vs. k-ratio for force-driven and displacement-driven FEA based topology optimized designs.

3.3.3 Geometric Advantage Dependencies and Trends

The GA for the human intuition and topology optimized designs are given in Tables 3.7 and 3.8. To better visualize the effects of the FE solver and k-ratio on GA, Figure 3.5 plots the GA vs. the k-ratio for the force-driven and displacement-driven FE solvers.

It is readily apparent from the tables that if k-ratio is 1, then the GA is also 1. Additionally, most of the human intuition designs had geometric advantages of approximately 1. This is again due to the flexible nature of the designs. Even so, the geometric advantage of the human intuition designs were still higher than half of the topology optimized designs.

Table 3.7. Geometric Advantages of Human Intuition Designs

Human Intuition Design	Geometric Advantage
Left Pivot Seesaw	1.001
Center Pivot Seesaw	1.000
Right Pivot Seesaw	0.999

Table 3.8. Geometric Advantages of Topology Optimized Designs using Force-Driven and Displacement-Driven FEA for Various k-ratios

k-ratio	Force-Driven FEA	Displacement-Driven FEA
0.01	1.708	2.818
0.02	1.711	2.946
0.1	1.586	2.651
0.2	1.481	2.402
1.0	1.000	1.000
5.0	0.684	0.422
10.0	0.630	0.380
50.0	0.584	0.340
100.0	0.584	0.355

Along those lines, it can be observed from Table 3.8 that the geometric advantage is better for unequal external stiffnesses that are weighted in favor of the input load. Conversely, designs with external stiffnesses weighted in favor of the output load displayed significantly worse geometric advantage. Displacement-driven FEA based designs showed the best GA for k-ratios less than 1, holding significantly higher values than force-driven FEA based designs. However for k-ratios greater than 1, the opposite is true, with force-driven FEA based designs maintaining the highest GA.

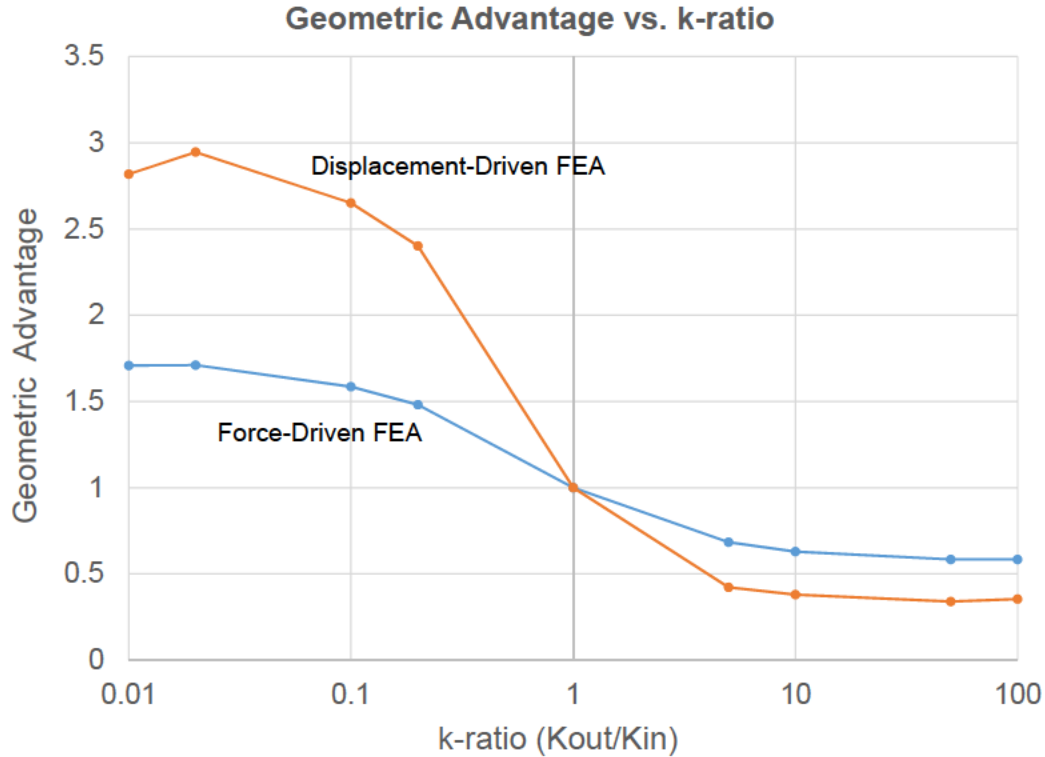


Figure 3.5. Geometric Advantage vs. k-ratio for force-driven and displacement-driven FEA based topology optimized designs.

3.4 Discussion

From these results, the MA of a compliant mechanism is primarily dependent on the type of FE solver used in the topology optimization iterations. Using a traditional force-driven FEA was shown to produce the highest mechanical advantages for corresponding k-ratios. The MA is also improved by increasing the external stiffness added to the output load relative to the input load's external stiffness. Figure 3.6 shows the trends of MA and GA with respect to changes in the stiffness of the structure (pivot position or k-ratio variation). Furthermore, it can also be concluded that topologies produced when using force-driven FEA are less subject to change when different k-ratios are applied.

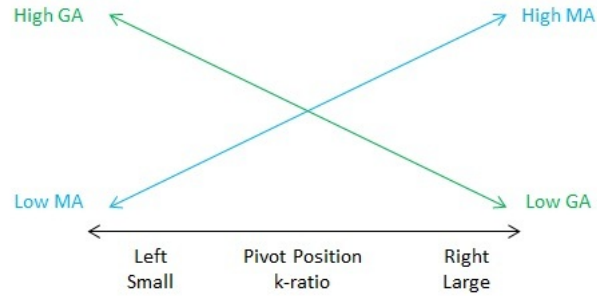


Figure 3.6. Mechanical and geometric advantage trends with respect to change in k-ratio.

When the k-ratio is 1, the resulting topologies are symmetric and consequently have a GA of 1. The GA of a compliant mechanism is maximized when the external stiffness added to the input load is higher than the relative output load's external stiffness. The GA can be further improved by using a displacement-driven FEA solver with topology optimization.

For the compliant mechanism lattice structure, a k-ratio of 1 was chosen (0.1/0.1) as a symmetrical structure is best suited to handle the uncertainty in impact locations and orientations. The force-driven FEA solver was selected for use as it provides a higher MA compared to the displacement-driven FEA. Figure 3.7 shows a flow chart for selecting what type of solver and k-ratio to use for high MA or GA.

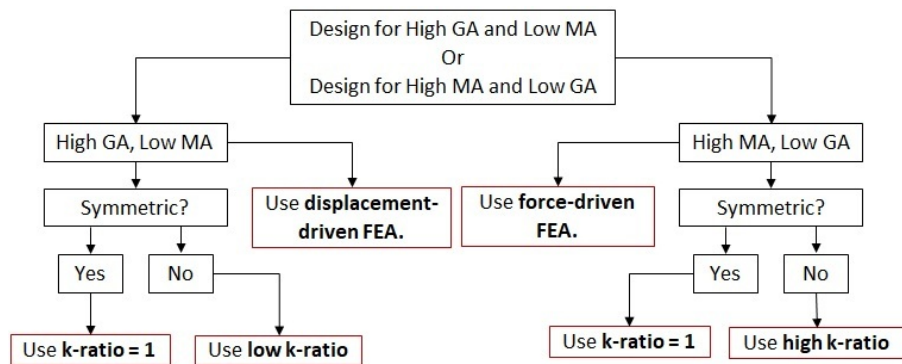


Figure 3.7. Mechanical and geometric advantage trends with respect to change in k-ratio.

4. HELMET LINER DESIGN USING A LATTICE STRUCTURE OF POSITIVE AND NEGATIVE POISSON RATIO COMPLIANT MECHANISMS

This chapter presents a study into the protective capabilities of a compliant mechanism array that dissipates impact energy through straining of the compliant mechanisms. Optimizer 2D is used to synthesis two types of compliant mechanisms. The first mechanism has a positive Poisson's ratio. The second mechanism operates has a negative Poisson's ratio. For this study, the Poisson's ratio of compliant mechanisms is defined as the displacement ratio between the input and output ports. These different mechanisms are assembled into three different liner arrangements. Liner implementation into the helmet is done by embedding the liner between an inner and outer polycarbonate shell, replacing the traditional EPP foam liner of a standard sports helmet. The liners are further developed through a series of ballistic impact tests to determine the final mechanism arrangement and ideal Shore A rubber hardness. The final liner is compared against an EPP foam liner to appraise the protection capabilities of the proposed liner.

4.1 Compliant Mechanism Synthesis

There are two primary designs for compliant mechanisms for use in the protective liner. Both function by redirecting an incoming radial force to a tangential direction. The first design for the compliant mechanism redirects forces out from the point of impact. Holistically, this gives the mechanism a positive Poisson's ratio. This mechanism will be referred to as the positive mechanism. The second design redirects the incoming force inwards, translating to a negative Poisson's ratio for the mechanism. This mechanism will be referred to as the negative mechanism.

Applying symmetric boundary conditions, only one-quarter of the mechanism needs to be optimized. Each problem was ran with a 30 x 30 mesh. For the positive mechanism, an input load was specified in the upper left corner and a dummy output load was specified in the lower right corner of the design domain. Similarly, for the negative mechanism, an input load was defined in the upper right corner with a dummy output load specified in the lower right corner of the design domain. Each load was given a width of 10% of the edge length, to achieve a large enough connection area so that implementation into the liner was robust. Roller supports were applied to the nodes along the left and bottom edges of the design domain for both mechanisms. Figure 4.1 shows the loaded nodes, load orientations, and supports for the positive and negative mechanisms.

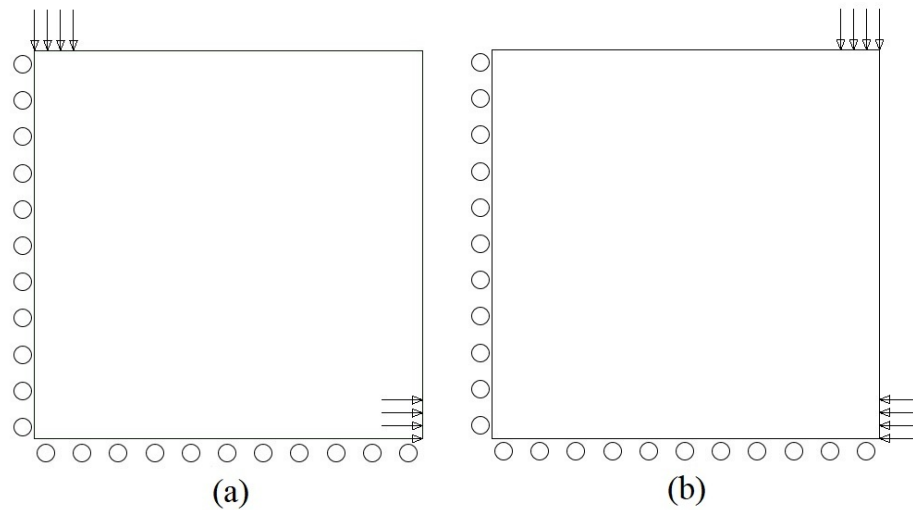


Figure 4.1. a) Positive mechanism and b) negative mechanism load and boundary conditions.

By performing topology optimization on these compliant mechanisms we obtain a 2D topology, which can be extruded for performance analysis using LS-Dyna. The volume fraction was set to 40%, and the penalization value for solid isotropic material with penalization (SIMP) was set to a value of 3. The volume fraction constraint was

added (despite the potential of topology masking) as excess mass prevents buckling of internal members and can conflict with the deformation path. A filter radius of 3 elements was also added to prevent the creation of spindly members. As previously mentioned, the k-ratio was set to 1, and the force-driven FE solver was used for optimization. Figure 4.2 , and the shows the results of the topology optimization code.

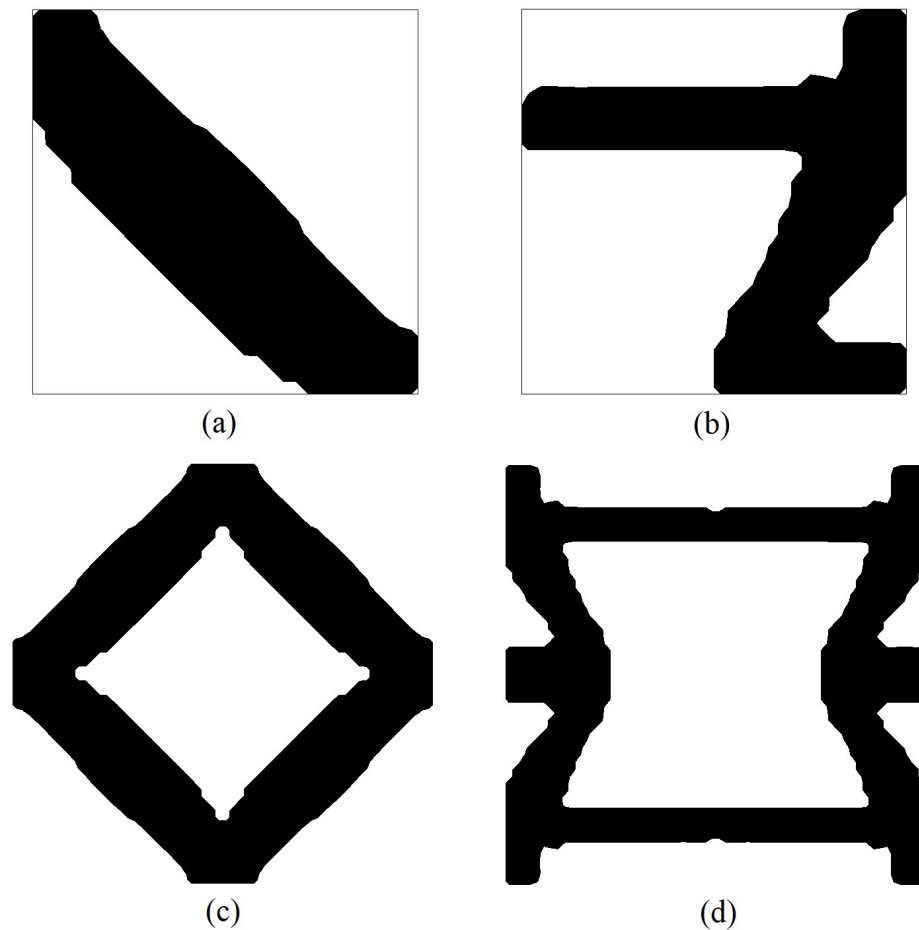


Figure 4.2. Optimized topology for a) positive mechanism and b) negative mechanism problems. c) Full positive mechanism topology. d) Full negative mechanism topology.

4.2 Mechanism Arrangements and Liner Architecture

Three different arrangements of mechanisms are assembled into liners for testing. These arrangements use (1) all positive mechanisms, (2) all negative mechanisms, and (3) alternating positive and negative mechanisms, shown in Figure 4.3. The liner with the alternating positive and negative mechanism arrangement is referred to as the net zero liner.

Each mechanism is 19.05 mm x 19.05 mm (0.75" x 0.75") with an extruded depth of 5 mm. The mechanism array sits between two polycarbonate shells that are 2 mm thick. To determine a suitable number of mechanisms, or liner section length, a helmet drop test simulation on a EPP foam helmet was studied to determine the impact area, shown in Figure 4.4 [28]. From this simulation, it was found that the projected impact area of a 4.9 kg headform moving at 5.46 m/s is approximately an oval with a major axis of 85 mm and a minor axis of 80 mm. Nine mechanisms were selected for each liner, as this covers over twice the length of the impact area's major axis.

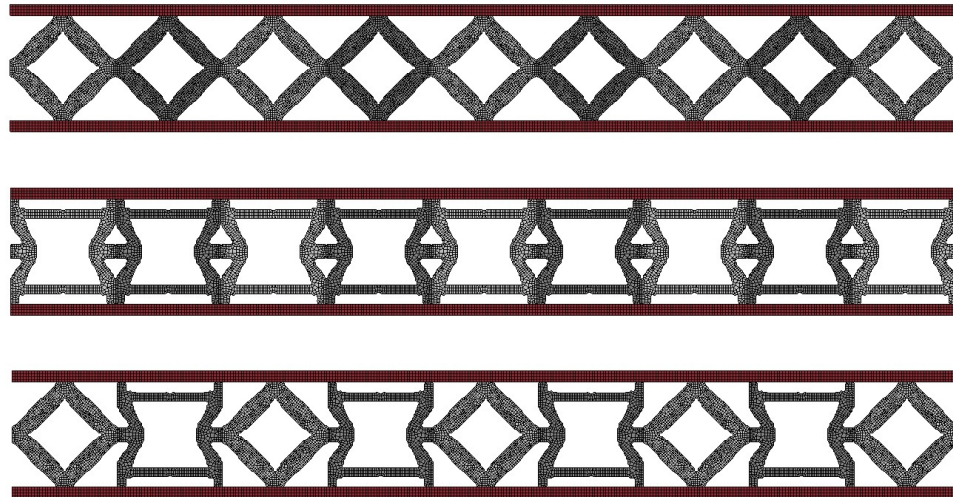


Figure 4.3. Liner designs. Top) positive arrangement liner. Middle) negative arrangement liner. Bottom) net zero liner.

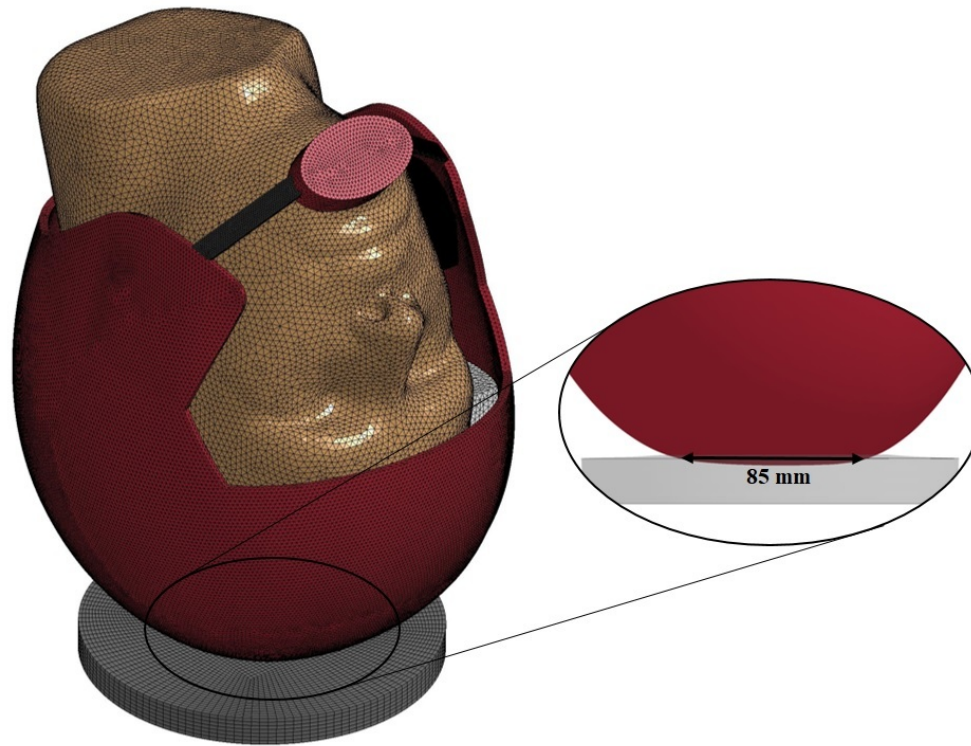


Figure 4.4. Simulation of NOCSAE Drop Test Method ND 001 on a basic foam-based helmet to determine width of (top side) impact area.

4.3 Numerical Results

Ballistic tests were carried out on a thickened 2D (extruded) cross-section of each liner. Performance results are compared and further developed with a weighted table. The final proposed liner is compared with current EPP foam liners to appraise its protective capabilities. A comparison of all designs is made through an acceleration vs. displacement plot.

4.3.1 Testing Setup

Each liner design is put through a simple ballistic test, using LS-Dyna (with base units of g, mm, ms). A rigid impactor impacts the top surface of the liner at 5 mm/ms. The bottom and side surface nodes are given roller supports, modeling the support of the head and other mechanisms of the liner. An additional roller support was added to the front and back sides of the 2D extrusion to prevent buckling and mimic support from 3D versions of the designs (see Appendix E for full 3D models). Figure 4.5 shows the initial test simulation setup for the positive arrangement liner test.

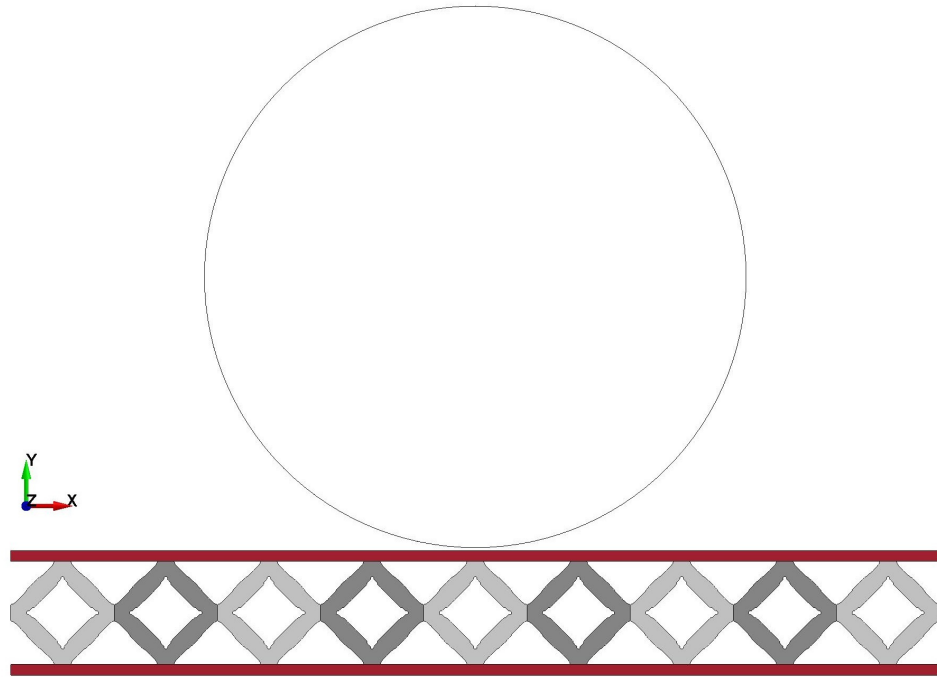


Figure 4.5. Initial ballistic test setup for the positive arrangement liner

4.3.2 Model Details

The polycarbonate shells are modeled as a simple elastic solid, as the given use case is well under yield stress for polycarbonate. Each shell has dimensions of 171.45 mm x 5 mm x 2 mm. A polycarbonate impactor is modeled as a rigid shell cylinder, and has diameter of 100 mm, depth of 5 mm, and mass of 70 g. Helmet Liners have their array mechanisms glued to each other, at their respective output ports, and to the top and bottom polycarbonate shells, at their respective input ports.

The compliant mechanisms were both modeled with the simple 2-parameter Mooney-Rivlin rubber model, as vulcanized natural rubber [29]. The parameters, C10 and C01, are obtained from a correlation study between Shore A hardness and the Mooney-Rivlin parameters [30]. Shore A hardness is a measure of the hardness of the material, using the type A scale (typically for softer materials). A Shore A hardness of 60A was used for mechanism arrangement comparison ballistic test. The mechanism hardness evaluation test compares the effects of using Shore A hardnesses of 50A, 60A, 70A, and 80A on the liners protection capabilities.

For the baseline simulation, LS-Dyna's low density foam model was used to model EPP [31]. All simulation variables not dependent on liner topology and hardnesses were kept consistent across ballistic tests. This includes timestep options, hourglass control, contact definitions, initial velocities, boundary conditions, element formulations, and other model options. Complete LS-Dyna material model keycards details can be found in Appendix D.

4.3.3 Performance Evaluation Criterion

It is widely known that TBIs and mTBIs are linked with the linear and rotational accelerations experienced by the brain [32]. Consideration of injurious accelerations are evaluated through the peak resultant acceleration and the head injury criterion (HIC) of the impactor.

The peak resultant acceleration of the impactor is the maximum value of the resultant acceleration experienced by the impactor. However, peak resultant acceleration does not capture all of the severity behind an impact. A steep spike to a low peak acceleration, can be just as detrimental, as a gradual rise to a high peak acceleration. As a result, the HIC, a measure of the likelihood of a head injury resulting from an impact, is used to capture the significance of the slope of a resultant acceleration curve [33]. The HIC is given by

$$\text{HIC} = \left\{ (t_2 - t_1) \left[\frac{1}{(t_2 - t_1)} \int_{t_1}^{t_2} a(t) dt \right]^{2.5} \right\}_{max} \quad (4.1)$$

where, $a(t)$ is the headform's acceleration measured in g (standard gravity), and t_1 and t_2 (seconds) are the initial and final times that correspond to a maximum HIC value. The ballistic test in this study does not accurately model impact on a human head, consequently, HIC values in this paper are used exclusively for comparison purposes, and should not be compared with external HIC measures. For this analysis we use HIC_{15} , which sets the integral's interval to 15 ms. When $t_2 = t_1 + 0.015$ ms, HIC_{15} is defined as

$$\text{HIC}_{15} = \left\{ (0.015)^{1.5} \left[\int_{t_1}^{t_1+0.015} a(t) dt \right]^{2.5} \right\}_{max}. \quad (4.2)$$

For the average adult, for an $\text{HIC}_{15} = 1000$, there is 18% probability of severe head injury, 55% probability of serious head injury, and 90% probability of a moderate head injury [34].

While it may seem suitable to analyze liners through their energy absorption capabilities, these evaluations are irrelevant within a simple impact test, as all liners absorb all of the initial kinetic energy when the impactor is brought to a stop (before rebounding). A more relevant evaluation would be to observe how the impact energy is managed within the array. To this end, the internal energy distribution among the compliant mechanism array is evaluated through the mean and standard deviation

(SD) of the distribution of each mechanisms peak internal energy (PIE). The last performance metric considered is the peak displacement seen by the impactors center of mass. Large displacements could prove to be uncomfortable for the wearer.

To aid with design evaluation, performance values, $f_j(x)$, are normalized. A subsequent weighted sum is calculated and used for justification of selecting designs for further analysis. The resulting weighted score is then defined as

$$F(\mathbf{x}) = \sum_j \omega_j \bar{f}_j(\mathbf{x}) \quad (4.3)$$

where \bar{f}_j is the normalized performance value and ω_j the corresponding weight. Table 4.1 shows the weights corresponding to performance criteria. Performance values in this analysis include peak resultant acceleration a_{max} , HIC₁₅, mean PIE, SD of PIE, and peak displacement d_{max} . Normalization is done as a fractional percent of the maximum value. Since most of the performance values are desired to be minimized, the best design is the one with the lowest weighted score. Note that, mean PIE criteria is desired to be maximized, so the sign was flipped for this row.

Table 4.1. Weights for Performance Evaluation Criteria

Criterion	Weight
Peak Resultant Accel.	0.30
HIC ₁₅	0.35
Mean PIE	0.05
SD of PIE	0.15
Peak Displacement	0.15

4.3.4 Liner Arrangement Analysis

Observing the acceleration curves of the liners, shown in Figure 4.6, one can see that the net zero liner has very steep and high spike in the acceleration curve lending this liner to have very poor acceleration-based criteria values. The positive and negative liners had much better acceleration curves with the negative having the best overall.

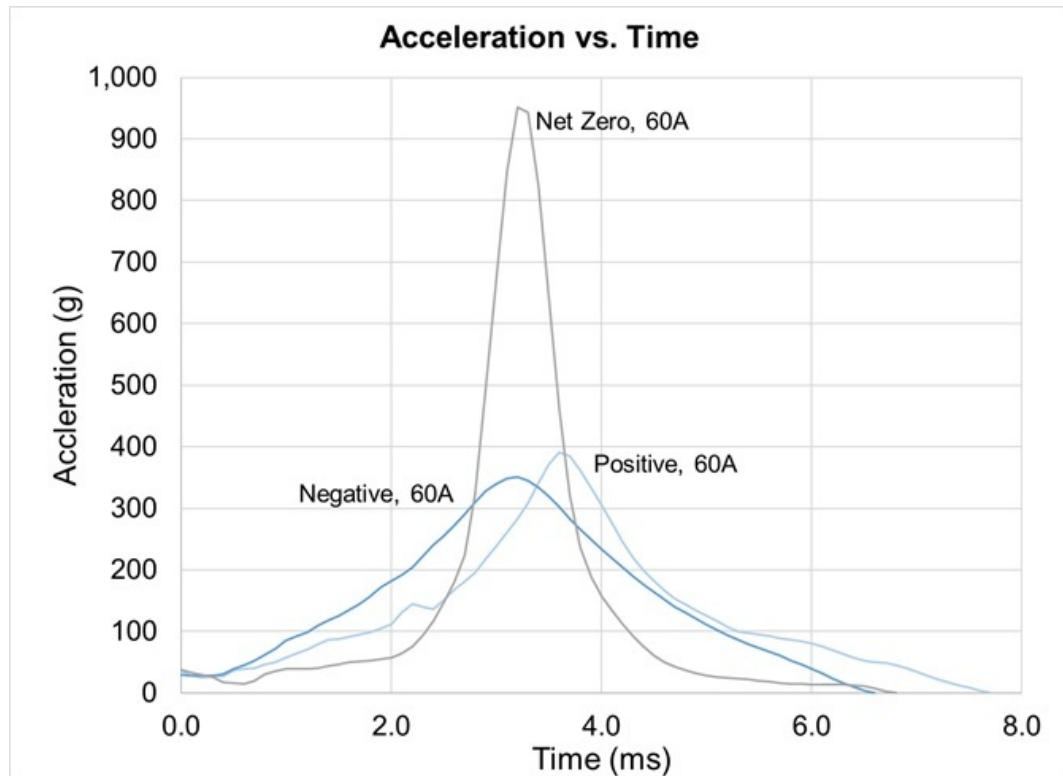


Figure 4.6. Acceleration vs. Time curves for positive, negative, and net zero liner arrangements.

The peak internal energy distribution among the compliant mechanisms (where compliant mechanisms are numbered from left to right) is shown in Figure 4.7. From here, it can be seen that the negative and net zero liners have very poor energy

management. The mechanisms that are near the impact location absorb significantly more energy than mechanisms that are farther away (on the left and right ends). The best overall PIE distribution among the compliant mechanisms is with the positive liner arrangement. The negative arrangement has double the SD of PIE compared to the positive arrangement. This is undesirable as failure to efficiently distribute energy into the compliant mechanism array leads to unnecessarily high stresses and strains, increasing the risk of mechanism failure.

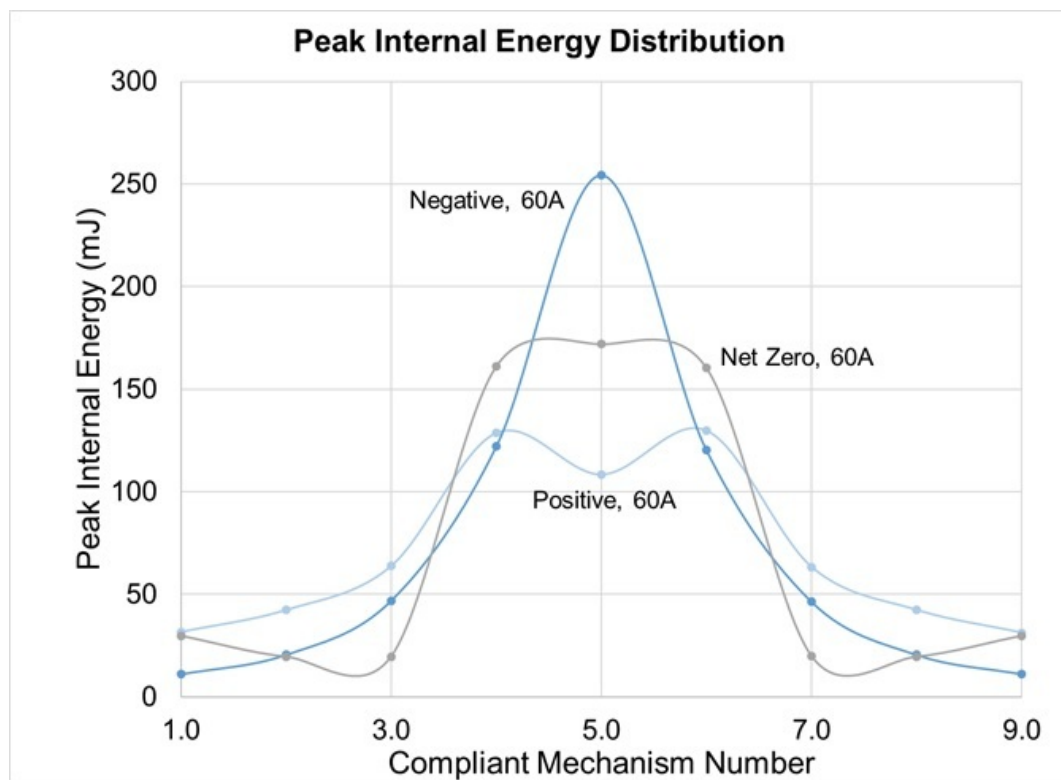


Figure 4.7. Peak internal energy distributions for compliant mechanisms in positive, negative, and net zero liner arrangements.

The impactor displacement for the different liners is similar at around 13 mm, however the negative liner did have the best overall impactor displacement of 11.1

mm. Compiling these results we obtain Table 4.2 and Table 4.3 which show the results of these test simulations and their normalized values.

Table 4.2. Mechanism Arrangement Performance Values

Mechanism Arrangement	Positive	Negative	Net Zero
Peak Resultant Acceleration of Impactor (g)	391.6	350.9	951.7
HIC ₁₅ of Impactor	273.3	255.3	768.3
Mean PIE of Compliant Mechanisms (N·mm)	71.4	72.7	70.2
SD of PIE of Compliant Mechanisms (N·mm)	40.3	80.6	70.9
Peak Displacement (mm)	13.0	11.1	13.4

Table 4.3. Mechanism Arrangement Performance Normalized Values

Mechanism Arrangement	Positive	Negative	Net Zero	Weight
Peak Resultant Acceleration of Impactor (g)	0.411	0.369	1.000	0.30
HIC ₁₅ of Impactor	0.356	0.332	1.000	0.35
Mean PIE of Compliant Mechanisms (N·mm)	-0.982	-1.000	-0.966	0.05
SD of PIE of Compliant Mechanisms (N·mm)	0.500	1.000	0.880	0.15
Peak Displacement (mm)	0.970	0.828	1.000	0.15
Weighted Score	0.419	0.451	0.884	–

The net zero arrangement exhibited very poor impact protection capabilities, obtaining nearly all normalized values of 1. This poor performance is due to lack of straining between adjacent positive and negative mechanisms. In the positive and negative arrangements, the output ports apply a compressive and tensile load (re-

spectively) on their members, converting kinetic energy to strain energy. In the net zero liner, input and output ports move together, collapsing the structure with little strain and thus, poor energy management (Figure 4.8).

The peak displacement across all arrangements were close to each other, with the negative arrangement having the smallest at 11.1 mm, due to the large void in the center of the topology. This void prevented the top shell from completely collapsing the structure, as was the case with the positive and net zero arrangements. Figure 4.8 shows the deformed topologies at their respective peak displacement. The positive mechanism arrangement had the best weighted score, and will be used for the following series of ballistic tests.

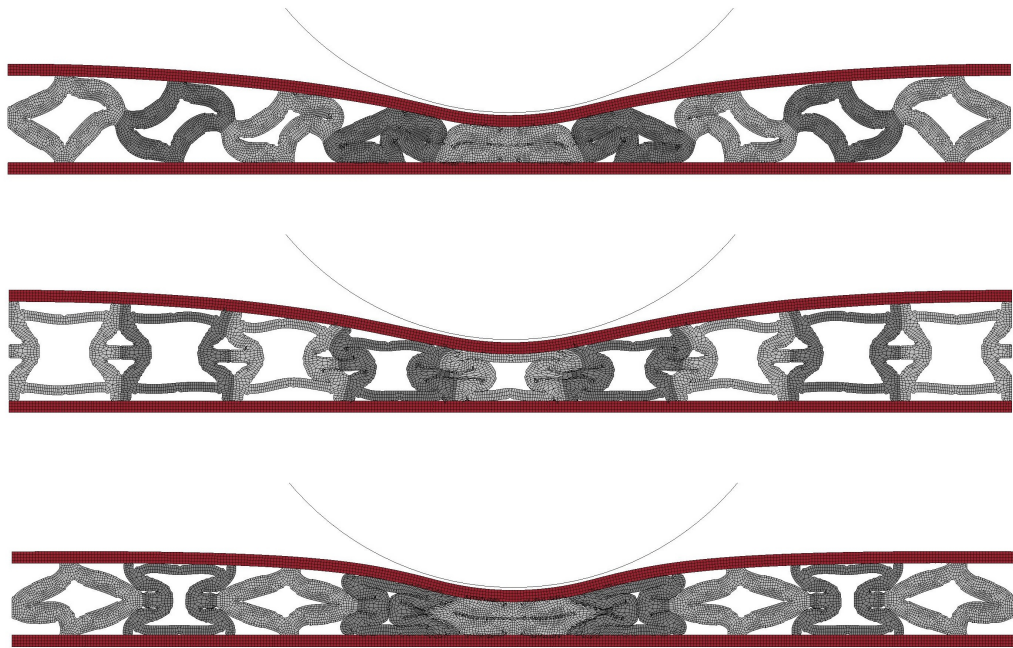


Figure 4.8. Deformed liners at peak impactor displacement. Top) positive arrangement liner. Middle) negative arrangement liner. Bottom) net zero liner.

4.3.5 Rubber Hardness Analysis

The second series of ballistic test simulations are designed to determine an ideal rubber hardness for use in the compliant mechanism array. Figure 4.9 shows the acceleration of the impactor with respect to time for the 50A, 60A, 70A, and 80A Shore A hardness liners. From this figure it can be seen that the soft 50A liner has a poor acceleration, with the 60A liner also seeing mediocre results. The best acceleration curves (and performance evaluation numbers) are seen in the 70A and 80A hardness liners.

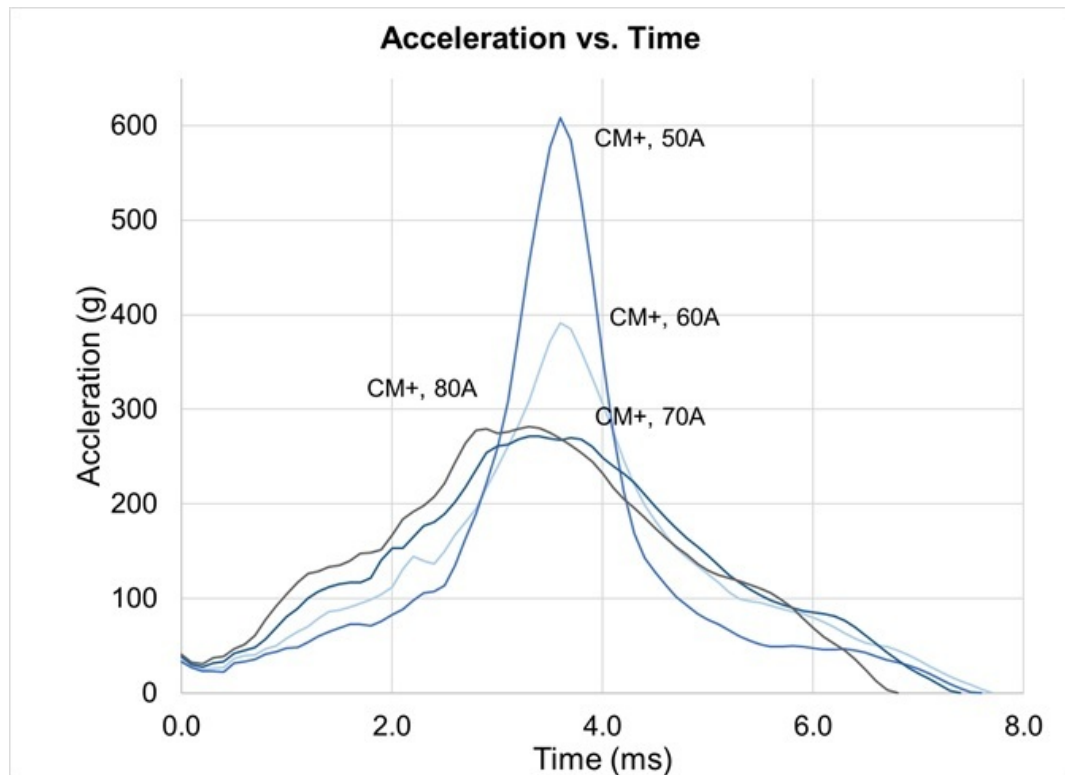


Figure 4.9. Acceleration vs. Time curves for 50A, 60A, 70A, and 80A rubber hardness liners.

The peak internal energy distribution among the compliant mechanisms (with the same numbering from the prior ballistic test) is shown in Figure 4.10. All of the liners showed similar peak internal energy distributions. Overall the best distribution is with the 60A liner, although the 70A liner is close in performance. PIE distribution is also significantly worse when comparing 70A with 80A, increasing from 46.4 N·mm to 56.6 N·mm.

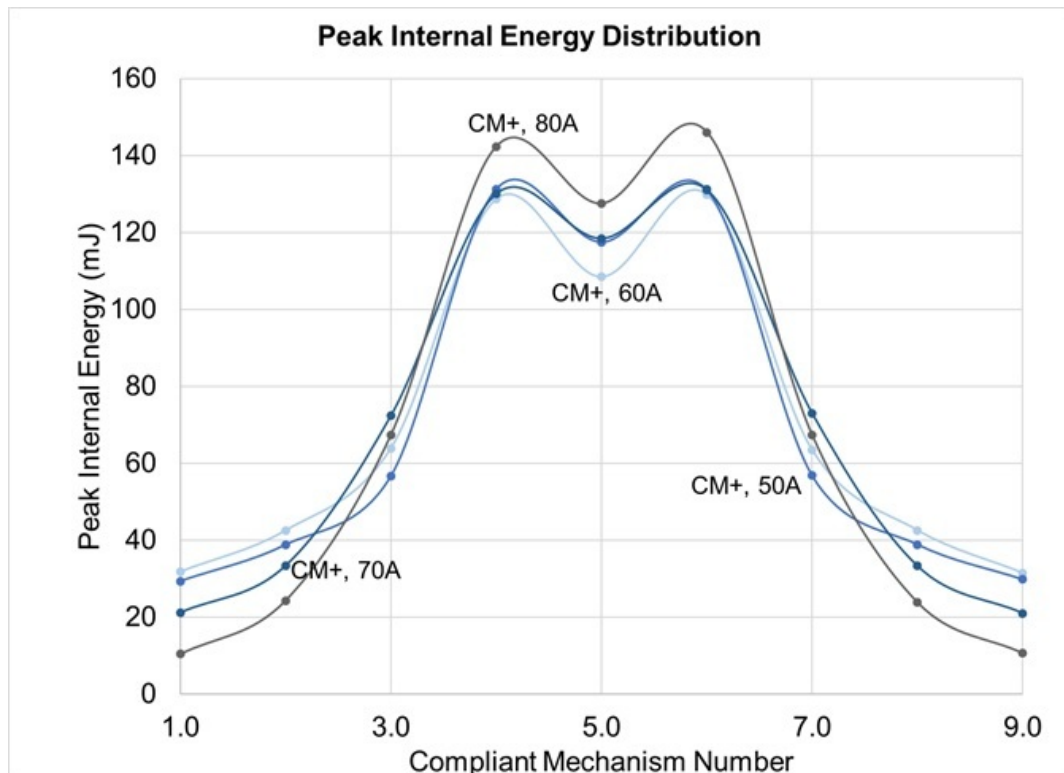


Figure 4.10. Peak internal energy distributions for compliant mechanisms in 50A, 60A, 70A, and 80A rubber hardness liners.

The impactor displacement seen in each liner is proportional to the rubber hardness. As expected, as soft rubber sees large displacement into the liner, while a harder rubber sees less displacement. Table 4.4 is a comprehensive table of all the results

of these test simulations. As with the last series of tests, Table 4.5 contains the normalized results.

Table 4.4. Shore A Hardness Performance Values

Shore A Hardness	50A	60A	70A	80A
Peak Resultant Acceleration of Impactor (g)	608.1	391.6	271.8	281.7
HIC ₁₅ of Impactor	455.8	273.3	210.9	200.7
Mean PIE of Compliant Mechanisms (N·mm)	70.0	71.4	70.4	68.9
SD of PIE of Compliant Mechanisms (N·mm)	43.7	40.3	46.3	56.6
Peak Displacement (mm)	13.7	13.0	11.9	10.9

Table 4.5. Shore A Hardness Performance Normalized Values

Shore A Hardness	50A	60A	70A	80A	Weight
Peak Resultant Acceleration of Impactor (g)	1.000	0.644	0.447	0.463	0.30
HIC ₁₅ of Impactor	1.000	0.600	0.463	0.440	0.35
Mean PIE of Compliant Mechanisms (N·mm)	-0.980	-1.000	-0.986	-0.965	0.05
SD of PIE of Compliant Mechanisms (N·mm)	0.772	0.712	0.818	1.000	0.15
Peak Displacement (mm)	1.000	0.949	0.869	0.796	0.15
Weighted Score	0.867	0.602	0.499	0.514	–

From Table 4.5 a Shore A hardness of 70A is the best natural rubber hardness to use with the positive mechanism arrangement. It should be noted that the 80A hardness liner was a close second to the 70A, losing out due to the slightly worse internal energy distribution. In actuality, due to the closeness of the weighted scores

between 70A and 80A, an ideal Shore A hardness lies between these two values. The 70A hardness design is used in the following section in comparison with a traditional EPP foam liner.

4.3.6 Expanded Polypropylene Foam Liner Comparison

Figure 4.11 shows a comparison of the proposed liner design with a traditional EPP foam liner. The proposed liner design sees significant improvement in peak resultant acceleration. The proposed liner design also has a less steep acceleration curve compared to the EPP foam's curve.

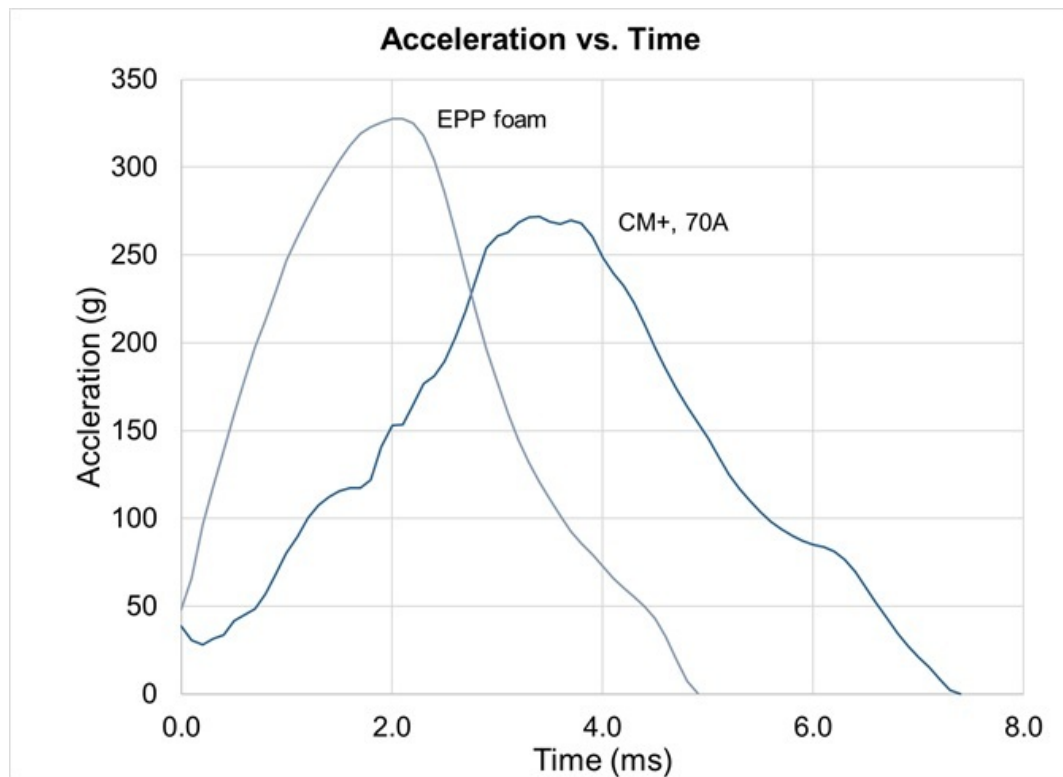


Figure 4.11. Acceleration vs. Time curves for the proposed positive 70A liner and an EPP foam liner.

The displacement is significantly less in the EPP foam liner due to their denser design. The cellular nature of the compliant mechanism lattice structure lends this design to much large liner displacements. Table 4.6 shows performance values for the two liners, however PIE distribution, of the compliant mechanism array, has been withdrawn as it is not applicable for the foam liner.

Table 4.6. EPP Foam vs. Proposed Design, Performance Comparison

Design	EPP Foam	Proposed Design	Percent Change from EPP Foam to Proposed Design
Peak Resultant Acceleration of Impactor (g)	327.4	271.8	16.98% \uparrow
HIC ₁₅ of Impactor	265.3	210.9	20.51% \uparrow
Peak Displacement (mm)	6.55	11.9	81.68% \downarrow

The proposed design shows significant benefits over a standard EPP liner with respect to acceleration-based criterion. Both the peak resultant acceleration and HIC of the impactor had improved values over the EPP foam liner. On the other hand, the compliant mechanism array deforms more than the EPP foam liner which could be uncomfortable for the user.

To further illustrate this comparison, Figure 4.12 shows a comprehensive acceleration vs displacement plot of all tested liner configurations. It can easily be observed from this plot that the compliant mechanism array liner offers better protection when considering the resulting acceleration experienced by the head, while the foam design experiences significantly less deformation, relative to the proposed design. Given these trade-offs, the proposed liner still demonstrates net improvements, in protection capabilities, over traditional EPP foam liners, as resultant acceleration is a far more frequent cause of head injuries than liner displacement.

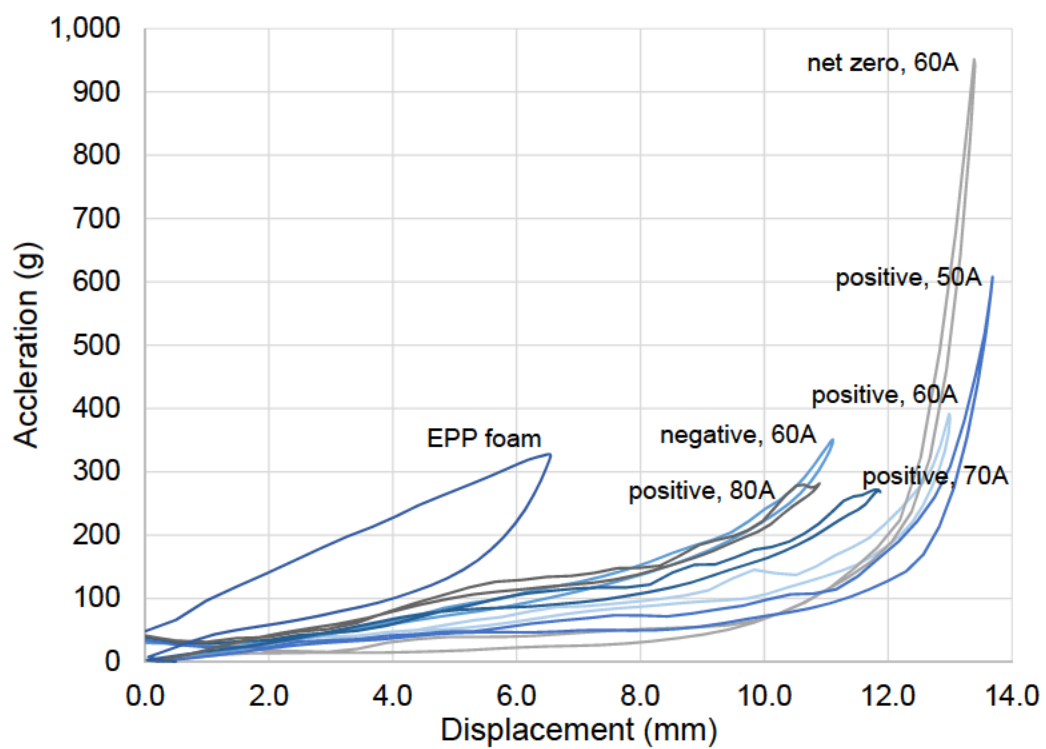


Figure 4.12. Acceleration vs. displacement simulation responses for EPP foam and compliant mechanisms array liner configurations.

5. SUMMARY AND FINAL REMARKS

5.1 Summary

This study presented a compliant mechanism lattice structure helmet liner for impact energy absorption. Compliant mechanisms were synthesized through a proposed design optimization tool and the mechanical and geometric advantage dependencies and trends of compliant mechanisms were also investigated. The research contributions made in this study include a 2D and 3D topology optimization code (featuring a LS-PrePost interfacier and displacement-driven finite element solver), the dependencies and trends of compliant mechanism advantages, and a developed helmet liner for impact energy absorption (featuring a compliant mechanism lattice structure).

In chapter 2, the development of the proposed design optimization tool was described. The details behind the LS-PrePost interfacier and it's uses was discussed. Select algorithms were outlined, along with the inter-workings of the displacement-driven finite element solver. In appendices A, B, and C, the extent of the optimizer tool's capabilities is given and shown through program documentation and example problems that will encourage future academic development.

In chapter 3, an investigation on the dependencies and trends of mechanical and geometric advantages in compliant mechanisms was performed. Human intuition and topology optimized design solutions were presented for the proposed seesaw problem. The merits of using topology optimization for compliant mechanism design was observed as mechanical and geometric advantage performance was significantly higher when compared with human intuition designs.

The dependencies and trends of the mechanical and geometric advantages were summarized. The mechanical advantage of a compliant mechanism was greater when using the force-driven finite element solver rather than the displacement-driven fi-

nite element solver. Additionally, the mechanical advantage is higher when a larger external stiffness is added to the input port relative to the output port's external stiffness. It was also shown that symmetrical topologies have geometric advantages of 1. The best geometric advantage was obtained when using the displacement-driven finite element solver and with a low k-ratio ($k_{in} > k_{out}$). Conversely, when designing with a high k-ratio ($k_{in} < k_{out}$), the best geometric advantage was obtained using the force-driven finite element solver. Lastly, it was noticed, for the seesaw problem, that change in the k-ratio had significant effects on the topology when using the displacement-driven finite element solver, however the topology was found to be more resilient when using the force-driven finite element solver.

In chapter 4, a helmet liner was designed using an array of positive and negative Poisson ratio compliant mechanisms. The mechanisms are arranged to form three different lattice structures. Liner designs were modeled as 2D extruded cross sections and two different ballistic tests were performed to determine the best liner design and mechanism rubber hardness. Performance values were tabulated and normalized for easy comparison. An acceleration vs. displacement graph is also provided for a visual comparison of all liner designs.

The best lattice structure configuration was the one comprised of all positive mechanisms. The best Shore A rubber hardness value was 70A. When the proposed final helmet liner design was compared with a traditional expanded polypropylene foam helmet liner, the proposed liner saw improvements in acceleration-based performance criterion (peak resultant acceleration and HIC_{15}) with a relatively large increase in peak liner displacement. While the peak liner displacement rose significantly, the helmet liner is still an improvement in protection capabilities over expanded polypropylene foam helmets, as the acceleration-based criterion is far more pertinent to head-injuries.

5.2 Limitations

For the sake of transparency, the limitations of this study are also detailed. The two-parameter Mooney-Rivlin rubber material model used in this study was based off of generalized values from a tensile uniaxial stress-strain test. While the comparative study to find the ideal Shore A rubber hardness helped alleviate some of the effects of using generalized rubber parameters, a final study is suggested for completeness.

Additionally, liners were modeled using 2D extruded cross section models. This was done for the sake of time, as prior 3D simulations of a single compliant mechanism could take up to 6 hours. The amount of computational time to solve a panel impact test would have been significantly longer. This run-time would have been compounded with increasing mechanisms. Thus, the liner configuration performance was tested in 2D, with the anticipation of a full 3D counterpart simulation.

5.3 Future Work

The future work of this study largely aims at addressing the limitations previously discussed. The first of these includes the collection and consolidation of a more accurate rubber model. Ideally, this model would be created off of 3D printed specimen tests, as the final helmet is intended to be 3D printed. Sections of helmet liners will be 3D printed for physical verification.

Now that the best lattice structure design has been identified, a panel of the liner needs to be modeled and simulated to better appraise the proposed liner's protective capabilities. The current lattice structure spans a 2D plane. Future work also pertains to the layering of the lattice structures to form a 3D structure. Additionally, the ideal size of the compliant mechanism could also be studied. As further developments continually get more advanced, eventually homogenization should be used to significantly reduce the computational cost of running each model.

After the compliant mechanism lattice structure is further developed, a high-fidelity full helmet model should be created and simulated with several NOCSAE

helmet test standards. Figure 5.1 shows a full helmet NOCSAE drop test simulation for a foam pad helmet. A similar compliant mechanism lattice structure helmet liner is planned to be similarly simulated. Due to the computational costs associated with such a complex liner, homogenization of the unit cell may be used to reduce computational costs to a more reasonable level. A full 3D printed helmet counterpart will be manufactured for physical verification. If proven to be sufficiently reliable at reducing head-injuries, the commercialization of this design will be brought to market.

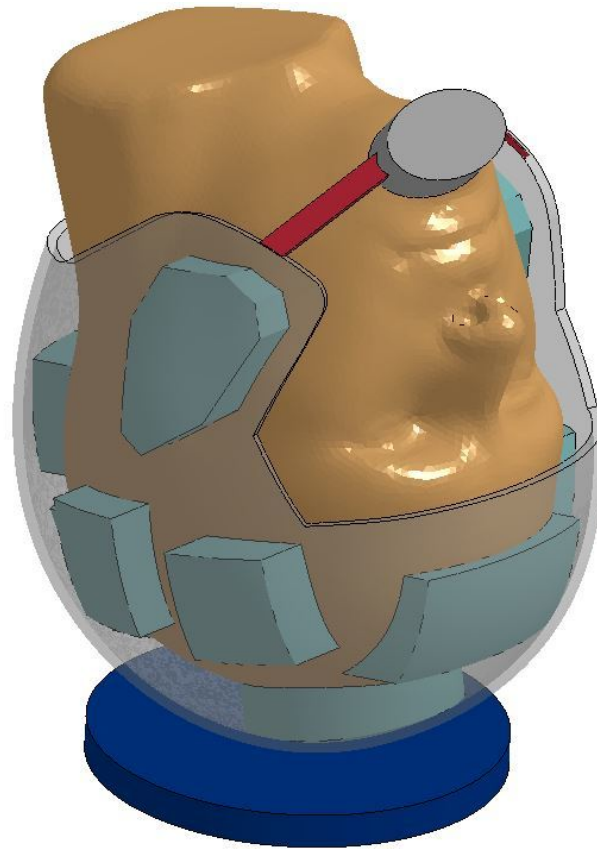


Figure 5.1. Full EPP foam helmet drop test simulation.

REFERENCES

REFERENCES

- [1] Recep Gorguluarslan, Umesh Gandhi, Raghuram Mandapati, and Seung-Kyum Choi. Design and fabrication of periodic lattice-based cellular structures. *Computer-Aided Design and Applications*, (A):50–62, Aug 2015.
- [2] Rebecca Firestone. Qii hers credit now allows open cell spray foam. *Green Compliance Plus - Mark English Architects*, pages 1–2, Dec 2011.
- [3] J. Brennan-Craddock, D. Brackett, R. Wildman, and R. Hague. The design of impact absorbing structures for additive manufacture. *Journal of Physics: Conference Series*, 382(1):1–8, 2012.
- [4] Fei Shen, Shangqin Yuan, Yanchunni Guo, Bo Zhao, Jiaming Bai, Mahan Qwamizadeh, Chee Kai Chua, Jun Wei, and Kun Zhou. Energy absorption of thermoplastic polyurethane lattice structures via 3d printing: Modeling and prediction. *International Journal of Applied Mechanics*, 8(7):1–13, 2016.
- [5] C.I. Hammett, R.G. Rinalid, and F.W. Zok. Pyramidal lattice structures for high strength and energy absorption. *Journal of Applied Mechanics*, 80:1–11, 2013.
- [6] Radek Vrana, Daniel Koutny, David Palousek, and Tomas Zikmund. Impact resistance of lattice structure made by selective laser melting from alsi12 alloy. *MM Science Journal*, pages 852–855, Dec 2015.
- [7] Radek Vrana, Daniel Koutny, and David Palousek. Impact resistance of different types of lattice structures manufactured by slm. *MM Science Journal*, pages 1579–1585, Dec 2016.
- [8] Shangqin Yuan, Fei Shen, Bai Jiaming, Chee Kai Chua, Jun Wei, and Kun Zhou. 3d soft auxetic lattice structures fabricated by selective laser sintering: Tpu powder evaluation and process optimization. *Materials & Design*, 120:317–327, April 2017.
- [9] A.G. Evans, V.S. Deshpande, J.W. Hutchinson, A.J. Jacobsen, and W. Barvosa-Carter. Concepts for enhanced energy absorption using hollow micro-lattices. *International Journal of Impact Engineering*, 37(9):947–959, April 2017.
- [10] Prasad S. Mehta, Jennifer Solis Ocampo, Andres Tovar, and Prathamesh Chaudhari. Bio-inspired design of lightweight and protective structures. *SAE International*, pages 1–8, April 2016.
- [11] Sean S Tolman, Issac L. Delimont, Larry L. Howell, and David T. Fullwood. Material selection for elastic energy absorption in origami-inspired compliant corrugations. *Smart Materials and Structures*, 23(9):1–5, Aug 2014.

- [12] Alexander Hasse and Lucio Flavio Campanile. Design of compliant mechanisms with selective compliance. *Smart Materials and Structures*, 18:1–5, Sept 2009.
- [13] Regina M. Gouker, Satyandra K. Gupta, Hugh A. Bruck, and Tobias Holzschuh. Manufacturing of multi-material compliant mechanisms using multi-material molding. *International Journal of Advanced Manufacturing Technologies*, 30:1049–1075, Feb 2006.
- [14] Vaibhav Gokhale. Design of a helmet with an advanced layered composite for energy dissipation using a multi-material compliant mechanism synthesis. *IUPUI ScholarWorks*, pages 1–53, Aug 2016.
- [15] Hong Zhou and Avinash R. Mandala. Topology optimization of compliant mechanisms using the improved quadrilateral discretization model. *Materials & Design*, 4:1–9, May 2012.
- [16] Qi Xia and Tielin Shi. Topology optimization of compliant mechanism and its support through a level set method. *Materials & Design*, 305:359–375, June 2016.
- [17] Liying Liu, Jian Xing, Qingwei Yang, and Yangjun Luo. Design of large-displacement compliant mechanisms by topology optimization incorporating modified additive hyperelasticity technique. *Materials & Design*, pages 1–9, Feb 2017.
- [18] Kai Liu and Andres Tovar. An efficient 3d topology optimization code written in matlab. *Structural and Multidisciplinary Optimization*, 50(6):1175–1196, Dec 2014.
- [19] Sangamesh R. Deepak, M. Dinesh, Deepak K. Sahu, and G.K. Ananthasuresh. A comparative study of the formulations and benchmark problems for the topology optimization of compliant mechanisms. *Journal of Mechanisms and Robotics*, 1:1–8, Feb 2009.
- [20] Gautam R. Kumar and G.K. Ananthasuresh. A study of mechanical advantage in compliant mechanisms. *National Conference on Machines and Mechanisms*, pages 1–8, Dec 2013.
- [21] Michael Yu Wang. Mechanical and geometric advantages in compliant mechanism optimization. *Frontiers of Mechanical Engineering in China*, 4(3):229–241, Sept 2009.
- [22] Bicycle Helmet Safety Institute. Bicycle helmet liners: Foam and other materials. *BHSI - Helmets, Bicycle Helmet Safety Institute website*, pages 1–2, Oct 2017.
- [23] Jesse Mez, Daniel H. Daneshvar, Patrick T. Kiernan, Bobak Abdolmohammadi, and et al. Clinicopathological evaluation of chronic traumatic encephalopathy in players of american football. *The Journal of the American Medical Association*, 318(4):360–370, July 2017.
- [24] Michael W. Collins, Mark R. Lovell, Grant L. Iverson, Robert Cantu, Maroon Joseph, and Field Melvin. Cumulative effects of concussion in high school athletes. *PubMed Neurosurgery*, 51(5):1175–1180, 2002.

- [25] Krister Svanberg. The method of moving asymptotes - a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24:359–373, 1987.
- [26] Erik Andreassen, Anders Clausen, Mattias Schevenels, Boyan S. Lazarov, and Ole Sigmund. Efficient topology optimization in matlab using 88 lines of code. *Structural and Multidisciplinary Optimization*, 43(1):1–16, Jan 2011.
- [27] The MathWorks Inc. Global optimization toolbox: User’s guide. *MathWorks*, pages 1–634, 2017.
- [28] National Operating Committee of Standards for Athletic Equipment. *Standard Test Method and Equipment used in Evaluating the Performance Characteristics of Headgear/Equipment*. NOCSAE ND 001 - 15m17, Rev. Jan, 2017.
- [29] P. H. Mott and C. M. Roland. Limits to poisson’s ratio in isotropic materials - general result for arbitrary deformation. *Physical Review B*, 80(13):1–3, 2009.
- [30] P. Altidis and B. Warner. Analyzing hyperelastic materials - with some practical considerations. *Midwest ANSYS Users Group*, pages 1–39, May 2009.
- [31] Remy Bouix, Viot Philippe, and Lataillade Jean-Luc. Polypropylene foam behavior under dynamic loadings: Strain rate, density and microstructure effects. *International Journal of Impact Engineering*, 36(2):329–342, 2009.
- [32] Thomas Gennarelli, Lawrence Thibault, G. Tomei, R. Wiser, D. Graham, and J. Adams. Directional dependence of axonal brain injury due to centroidal and non-centroidal acceleration. *SAE Technical Papers*, pages 564–574, Nov 1987.
- [33] Brian McHenry. Head injury criterion and the atb. *ATB Users Group*, pages 1–8, 2004.
- [34] Murray Mackay. The increasing importance of the biomechanics of impact trauma. *Sadhana*, 32(4):397–408, 2007.

APPENDICES

A. OPTIMIZER PROGRAM DOCUMENTATION

This is program documentation for the Optimizer 2D and 3D program (versions 1.6a & 1.6b).

A.1 Program Inputs

A.1.1 Input File

The optimizer program receives the optimization problem's geometry, loads, and boundary conditions from a .k file that is created in LS-PrePost. The .k file needs to be in the same directory as the rest of the program files. The name of the .k file is entered in line 20 of the Optimizer main file (see Figure A.1). The file extension '.k' needs to be included in the file name. For Poisson's ratio tailoring problems, this line is ignored and automatically replaced with the name PoissonsRatio2d.k (or PoissonsRatio3d.k for Optimizer 3D).

```

19      % INPUT FILE
20 -    fname = 'beam.k';
21      %For PR problem, fname is ignored. Problem parameters are defined below

```

Figure A.1. Lines 19-21: Input File Code Segment

A.1.2 Material Properties

Material properties are entered at lines 55-57 (lines 56-58 for Optimizer 3D) (see Figure A.2). $E0$ is the Young's modulus of a fully solid element, $x_i = 1$. $Emin$ is the Young's modulus of a void element, $x_i \approx 0$. nu is the Poisson's ratio of the material.

```

54      % MATERIAL PROPERTIES
55 -    E0 = 1;
56 -    Emin = 1e-9;
57 -    nu = 0.3;

```

Figure A.2. Lines 54-57: Material Properties Code Segment

A.1.3 Optimization Parameters

Optimization parameters are entered at lines 60-65 (lines 61-66 for Optimizer 3D) (see Figure A.3). *VF* is the volume fraction that the optimizer will constrain the problem to, if volume fraction constraints are enabled. *penal* is the penalization exponent for use with SIMP Young's modulus calculations. 3.0 is a recommended value. *rmin* is the minimum radius that members (or beams) of the topology can have. *tolx* is the tolerance for the change in step size of the objective function when fmincon algorithms are used. It is the tolerance for the maximum change in a single element of the design domain when using OC and MMA algorithms. Recommended values for *tolx* are 1e-2 or 1e-3. *maxloop* is the maximum number of iterations for the program. If the maximum number of iterations are reached before a solution is found, the program stops the optimization loop and proceeds to save the last iteration's results. *displayflag* is a boolean operator that dictates whether or not the program displays the topology at every step (1 displays topology, 2 hides it).

```

59      % OPTIMIZATION PARAMETERS
60 -    VF = 0.50;
61 -    penal = 3;
62 -    rmin = 2;
63 -    tolx = 1e-3;
64 -    maxloop = 1000;
65 -    displayflag = 1;

```

Figure A.3. Lines 60-65: Optimization Parameters Code Segment

A.1.4 Poisson's Ratio Problem Parameters

If the optimizer is used to perform Poisson's ratio tailoring, the program loads the predefined problem data and ignores the .k file name given. Lines 68-71 (lines 69-73 for Optimizer 3D) is where Poisson's ratio problem parameters are entered (see Figure A.4). *x* is the number of elements in the x-direction. *y* is the number of elements in the y-direction. *z* is the number of elements in the z-direction (Optimizer 3D only). *pr* is the objective Poisson's ratio that the program will try to synthesize a topology for. *s* is the number of elements for the input and output loads (in 3D, *s* is side length of the input and output squares). See Figure A.5 for a graphical representation of these variables in 3D. Using this objective function corresponds to the positive Poisson ratio problem used in the proposed helmet liner design (Figures 4.1 & 4.2).


```

67 % POISSON'S RATIO PROBLEM PARAMETERS
68 - x = 30; %Number of elements in the x-direction
69 - y = 30; %Number of elements in the y-direction
70 - pr = -1.0; %Poisson ratio objective
71 - s = 15; %Number of elements on top and right side that are loaded.

```

Figure A.4. Lines 68-71: Poisson's Ratio Problem Parameters Code Segment

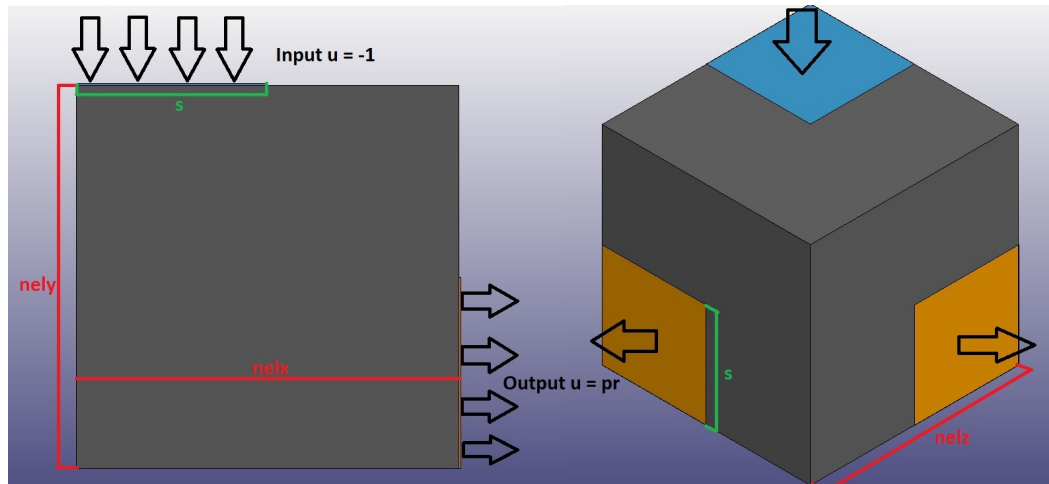


Figure A.5. Graphical Representation of Poisson's Ratio Problem Parameters

A.2 Program Options

A.2.1 Select Objective Function

In line 28, the objective function to use with the optimizer is specified (see Figure A.6). The variable, *obj*, can be a 1, 2, 3, or 4. Other entries besides this will be ignored and the default objective function (minimum compliance, 1 load) will be used. The key is as follows:

If $obj = 1$, use the minimum compliance, 1 load, objective function.

If $obj = 2$, use the minimum compliance, 2 loads, objective function.

If $obj = 3$, use the maximum mutual potential energy objective function.

If $obj = 4$, use maximum mutual potential energy objective function with the Poisson's ratio tailoring problem.

```

23 % SELECT OBJECTIVE FUNCTION TO USE
24 % 1 = Minimum Compliance (1 Load) (DEFAULT)
25 % 2 = Minimum Compliance (2 Load)
26 % 3 = Maximum Mutual Potential Energy (Compliant Mechanism)
27 % 4 = Poisson's Ratio Tailoring Problem
28 - obj = 2;

```

Figure A.6. Lines 23-28: Select Objective Function Code Segment

A.2.2 Select Algorithm

In line 38, the algorithm to use in the optimization loop is specified (see Figure A.7). The variable, *alg*, can be a 1, 2, 3, 4, 5, 6, or 7. Other entries besides this will be ignored and the default algorithm (Method of Moving Asymptotes) will be used. The key is as follows:

If *alg* = 1, use the Optimality Criterion algorithm.

If *alg* = 2, use the Interior-Point algorithm via MATLAB's *fmincon* function.

If *alg* = 3, use the Sequential Quadratic Programming algorithm via MATLAB's *fmincon* function.

If *alg* = 4, use the Sequential Quadratic Programming - Legacy algorithm via MATLAB's *fmincon* function.

If *alg* = 5, use the Trust-Region-Reflective algorithm via MATLAB's *fmincon* function.

If *alg* = 6, use the Active-Set algorithm via MATLAB's *fmincon* function.

If *alg* = 7, use the Method of Moving Asymptotes algorithm.

For most problems, the Method of Moving Asymptotes algorithm is recommended.

```

30 % SELECT ALGORITHM TO USE
31 % 1 = Optimality Criterion
32 % 2 = Interior-Point (fmincon)
33 % 3 = Sequential Quadratic Programming (fmincon)
34 % 4 = Sequential Quadratic Programming - Legacy (fmincon)
35 % 5 = Trust-Region-Reflective (fmincon)
36 % 6 = Active-Set (fmincon)
37 % 7 = Method of Moving Asymptotes (DEFAULT)
38 - alg = 7;

```

Figure A.7. Lines 30-38: Select Algorithm Code Segment

A.2.3 Select Finite Element Method

In line 43, the finite element method is specified (see Figure A.8). The variable, *fea*, can be a 1 or a 2. Other entries besides this will be ignored and the default method (Force-Driven FEA) will be used. The key is as follows:

If *fea* = 1, use force-driven finite element analysis.

If *fea* = 2, use displacement-driven finite element analysis.

```

40 % SELECT FINITE ELEMENT METHOD
41 % 1 = Force-Driven FEA (DEFAULT)
42 % 2 = Displacement-Driven FEA
43 - fea = 1;

```

Figure A.8. Lines 40-43: Select Finite Element Method Code Segment

A.2.4 Select Design Variable Initialization

In line 48, the way that the design variables are initialized is specified (see Figure A.9). The variable, *int*, can be a 1 or a 2. Other entries besides this will be ignored and the default initialization (initialize densities with volume fraction) will be used.

The key is as follows:

If *int* = 1, initialize densities with the volume fraction

If *int* = 2, initialize densities with random numbers.

```

40 % SELECT FINITE ELEMENT METHOD
41 % 1 = Force-Driven FEA (DEFAULT)
42 % 2 = Displacement-Driven FEA
43 - fea = 1;

```

Figure A.9. Lines 40-43: Select Finite Element Method Code Segment

At the start of the optimization loop, if the problem's topology is slow to change or the convergence criterion is met very quickly; consider running the problem with *int* = 2, as this may fix these problems.

A.2.5 Build Full Structure if the Problem is Symmetric?

If the problem has symmetrical boundary conditions, the program can build the full problem after the subproblem is optimized. Lines 51-52 (lines 51-53 for Optimizer

3D) allow the user to specify if this construction should occur and over which axes or planes to mirror the problem (see figure A.10). Setting a symmetry indicator to 'true' causes the mirroring of the problem.

```

50      % BUILD FULL STRUCTURE IF THE PROBLEM IS SYMMETRIC?
51 -    ysym = false; %(horizontal) Symmetry about y axis?
52 -    xsym = false; %(vertical) Symmetry about x axis?

```

Figure A.10. Lines 50-52: Enabling Symmetrical Problem Build Code Segment

A.2.6 Additional Options

Optimizer 2D and Optimizer 3D have additional options that cannot be changed from the main file.

- If random density initialization is selected ($int = 2$), then the interval that random numbers are assigned from can be changed. Line 8 of any algorithm file (alg1_OC.m, alg2_IP.m, etc...) defines variables a and b , which specify that random numbers assigned should be within the interval: $[a, b]$. The default range for all algorithms is in-between 0.0 and 1.0. Note: in order for the densities to be realizable, $a \geq 0$ and $b \leq 1$.
- For all fmincon algorithms, except Trust-Region-Reflective, the volume fraction constraint can easily be removed by uncommenting lines 20 and 21, and commenting lines 22 and 23 within the algorithm file (alg1_OC.m, alg2_IP.m, etc...). This change can be see in Figure A.11.

<pre> 19 %Optional Volume Constraints 20 %Aeq = []; 21 %Beq = []; 22 - Aeq = ones(1,numel(x)); 23 - Beq = numel(x)*VF; </pre>	<pre> 19 %Optional Volume Constraints 20 - Aeq = []; 21 - Beq = []; 22 %Aeq = ones(1,numel(x)); 23 %Beq = numel(x)*VF; </pre>
--	--

Figure A.11. Lines 19-23: Disabling Volume Fraction Constraint for fmincon Algorithms (left: volume fraction constraint enabled, right: volume fraction constraint disabled)

- When using the minimum compliance, 2 loads, ($obj = 2$) or maximum mutual potential energy ($obj = 3$) objective functions, a user-defined contact spring stiffness

may be added to the input and output degrees of freedom. This value can be changed on line 15 of the `objfn_comp2.m` and `objfn_mpe.m` files. For minimum compliance, 2 loads, the default value is 0. For maximum mutual potential energy, the default value is 0.1.

- The options behind all `fmincon` algorithms can be modified in lines 25-35 within the algorithm files (`alg1_OC.m`, `alg2_IP.m`, etc...).
- When using the minimum compliance, 2 loads, objective function, the weight for each load can be defined in line 38, `objfn_comp2.m`. w is the weight of load 1 and $(1 - w)$ is the weight of load 2.
- The output STL file creation options can be modified in the `Top3dSTL_v3.m` file.
- For Optimizer 3D, the options for the 3D topology figures can be changed in the `display_3D.m` file.

A.3 Program Output

A.3.1 Runtime Figures and Command-Line Printouts

During execution, the Optimizer will post

- the name of the input file
- the objective function used
- the algorithm used
- the finite element method used
- the initialization method used
- Material Properties
- Optimization Parameters

and a running list of the current optimization iteration, objective function value, volume fraction, and change in function. For compliant mechanisms (`obj = 3`), the mechanism's mechanical and geometric advantage is also calculated and displayed. Figure A.12 shows an output sample.

The current iteration's topology will also be displayed in a figure. If the user wishes to save the current iteration before convergence criterion are met, they may press the 'Esc' key while the figure window is active to prematurely terminate the optimization loop and save the output (for OC and MMA algorithms only; `fmincon` produces a plot that has a stop button on it for the same effect).

```

Running 2D Optimizer on file: seesaw.k

Program Options:
obj = 2, Minimum Compliance (2 Loads) Objective Function
alg = 7, Method of Moving Asymptotes Algorithm
fea = 1, Force-Driven FEA
int = 1, Volume Fraction Density Initialization

Material Properties:
E0 = 1, Emin = 1.0e-09, nu = 0.30

Optimization Parameters:
VF = 0.25, penal = 3, rmin = 2.0, tolx = 1.0e-03, maxloop = 1000

Output:
It.:  1 Obj.:  210.7050 Vol.:  0.181 ch.:  0.397
It.:  2 Obj.:   43.0493 Vol.:  0.216 ch.:  0.325
It.:  3 Obj.:   13.6626 Vol.:  0.240 ch.:  0.242
It.:  4 Obj.:    8.3750 Vol.:  0.247 ch.:  0.151
It.:  5 Obj.:    7.8207 Vol.:  0.247 ch.:  0.201
It.:  6 Obj.:    7.5128 Vol.:  0.247 ch.:  0.194
It.:  7 Obj.:    7.2714 Vol.:  0.247 ch.:  0.173
It.:  8 Obj.:    7.0636 Vol.:  0.247 ch.:  0.180
It.:  9 Obj.:    6.8591 Vol.:  0.246 ch.:  0.193
It.: 10 Obj.:    6.6430 Vol.:  0.245 ch.:  0.266
It.: 11 Obj.:    6.4056 Vol.:  0.244 ch.:  0.354
It.: 12 Obj.:    6.1915 Vol.:  0.247 ch.:  0.298
It.: 13 Obj.:    6.0875 Vol.:  0.249 ch.:  0.247
It.: 14 Obj.:    6.0368 Vol.:  0.249 ch.:  0.229
It.: 15 Obj.:    6.0075 Vol.:  0.249 ch.:  0.272
It.: 16 Obj.:    5.9807 Vol.:  0.249 ch.:  0.349

Premature stop triggered by user (Esc key), ending Optimization loop.
Wrote 1040 facets to C:\Users\jnajmon\Documents\Thesis Research - Helmet Design\
Elapsed time is 2.773803 seconds.

```

Figure A.12. Command-Line Sample Output

A.3.2 Saved Files

An output folder is created with the name: ‘OPTIMIZER_2D_OUTPUT_yourfilename’ to save the three output files of the program. The program saves a .mat file of the design domain’s densities, a .STL file of the resulting topology, and a .txt file of the command-line printouts.

A.4 Creating a Problem Input File in LS-PrePost

A.4.1 Geometry Creation

In LS-PrePost, use the ‘shape mesher’ tool to create 2D rectangles and 3D cuboids. Within the tool, the ‘4N Shell’ entity is used to create 2D problems (define point counter-clockwise starting with the lower left point) and the ‘Box Solid’ entity is used to create 3D problems. Figure A.13 shows the shape mesher interface.

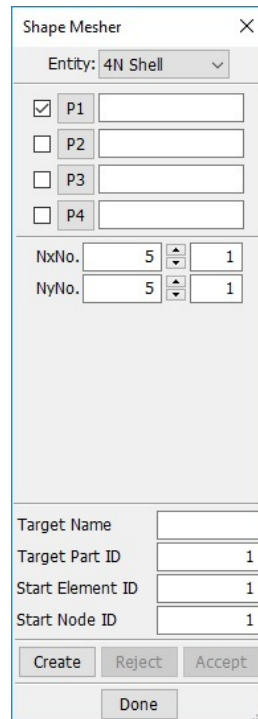


Figure A.13. LS-PrePost: Shape Mesher Tool Interface

A.4.2 Boundary Condition Creation

In LS-PrePost, use the ‘Entity Creation’ interface to apply boundary conditions to the problem. Within the ‘Boundary’ tab, select ‘Spc’ (specific point constraint) to access the menu used to add boundary conditions (Figure A.14). Optimizer 2D only recognizes X and Y degrees of freedom constraints. Optimizer 3D recognizes X, Y, and Z degrees of freedom constraints.

For every boundary condition that is created, a corresponding node set is also created. By default, LS-PrePost will name the node sets “NODESET(SPC) #”, where # starts at 1 for the first node set and counts upward. The LS-PrePost interpreting MATLAB code looks for these default names, so any attempt to rename the node sets will result in an error. It is actually the best practice to leave sub-features of the model with their default names. Another noteworthy practice is to always ensure your model’s node sets are number sequentially. Out of order numbering (which usually results from modifying/deleting prior boundary conditions) can cause errors.

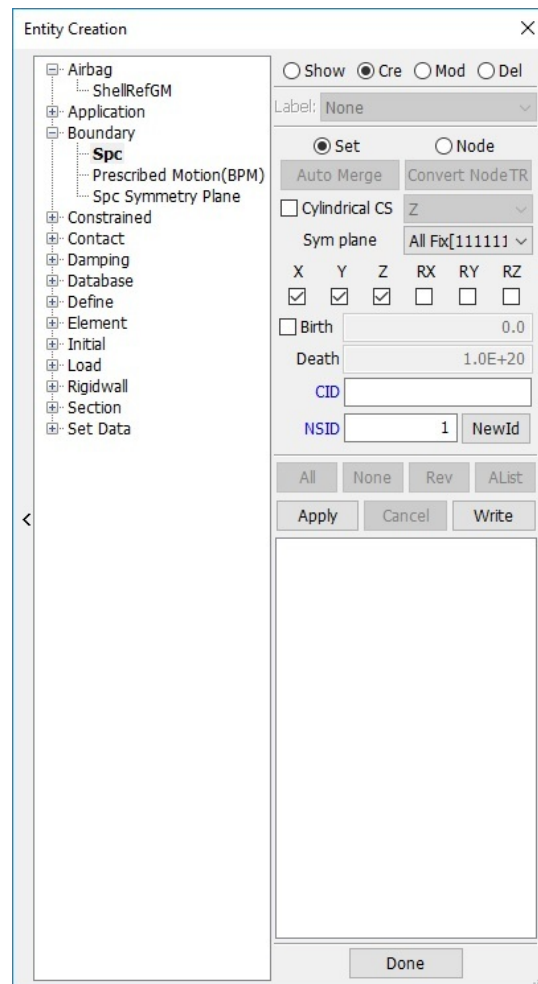


Figure A.14. LS-PrePost: Entity Creation Tool Interface - Boundary Creation

A.4.3 Load Case Creation

Objective functions minimum compliance, 2 loads, and maximum mutual potential energy require two load cases to be applied to the problem. Minimum compliance, 1 load, ignores a second load case. To create load cases, use the ‘Entity Creation’ interface in LS-PrePost (Figure A.14). Within the ‘Load’ tab, select ‘Node’ to access the menu to create nodal loads. Similar to boundary condition creation, Optimizer 2D and Optimizer 3D only read loads in Cartesian coordinate directions (X_dir, Y_dir, Z_dir).

LCID is read by the program to specify which load case the current load will be added to (LS-PrePost uses LCID for something else). Unlike boundary condition creation, LS-PrePost does not create a node set for each load applied. This makes adding additional loads to load cases easy, despite having varying directions, magnitudes, or application points. Additional loads can simply be added separately and will be appended onto the corresponding load case list that the LCID specifies.

The SF value is the magnitude of the force. The CID value is not read by MATLAB. LS-PrePost limits the number of loads that can be applied to an individual node to one. Therefore, in order to add loads that contain multiple coordinates, the .k file will need to be opened in a text editor and manually manipulated. Within the .k file, find the ‘*LOAD_NODE_POINT’ keyword title. Here is where the nodal loads are listed. Locate the node(s) that you wish to add a second (or third) load to, and copy and append these lines onto the list. From here you can manually change the dof, lcid, and sf column values to create the loads.

B. 2D EXAMPLES

B.1 Snail Problem - Minimum Compliance, 1 Load Case

- Open LS-PrePost.
- In the upper-left menu go to → FEM → Element and Mesh → Shape Mesher
- Choose '4N Shell' in the Entity Menu.
- Enter the following data (Figure B.1) into the tool, to create a 60 x 20 2D mesh.

Shape Mesher

Entity: 4N Shell

P1 0,0,0

P2 60,0,0

P3 60,20,0

P4 0,20,0

NxNo. 60 1

NyNo. 20 1

Target Name

Target Part ID 1

Start Element ID 1

Start Node ID 1

Create Reject Accept

Create mesh for entity

Figure B.1. Create 60 x 20 2D Mesh

- In the upper-left menu go to → FEM → Model and Part → Create Entity
- Open the Boundary drop-down menu. Click on Spc (specific point constraint).

- Under the Create Tab (Cre), deselect the Z, RX, RY, and RZ constraints, and apply a fixed x and y degree of freedom constraint on the nodes (16 nodes on left upper edge, and 2 on the bottom edge) shown in Figure B.2.

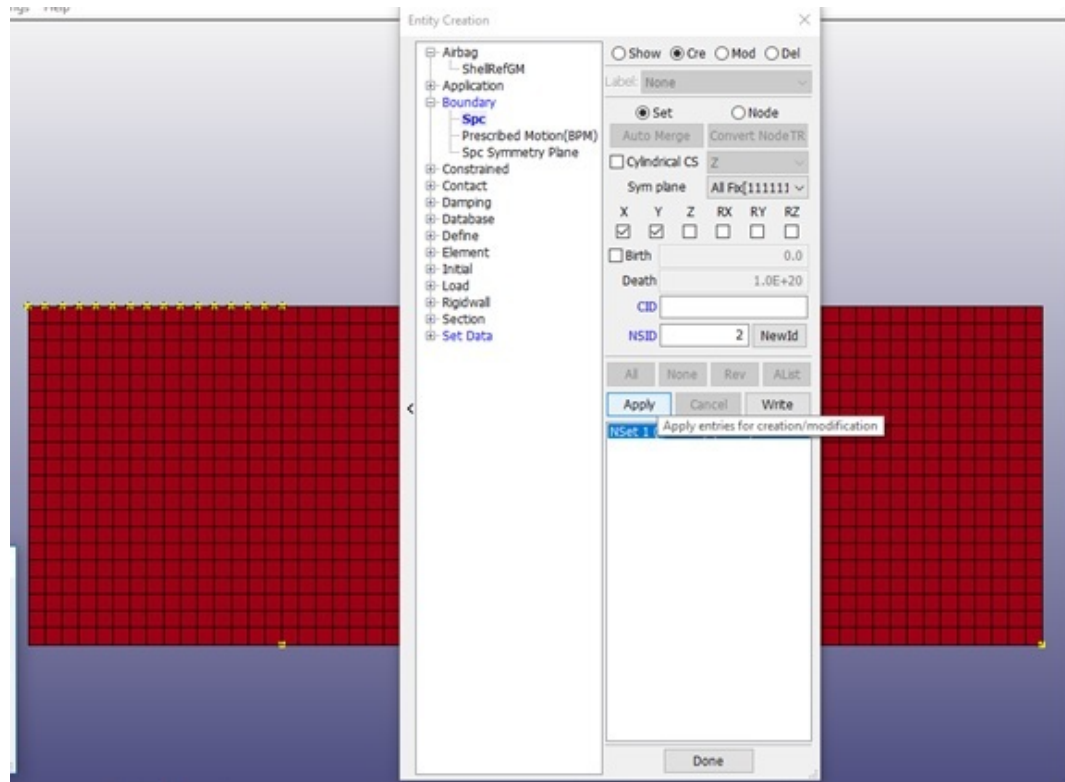


Figure B.2. Constrain X and Y Degrees of Freedom on Selected Nodes

- Open the Load drop-down menu. click on Node.
- Under the Create Tab (Cre), apply a load in the x-direction on the points shown in Figure B.3. The load on the left point has a value of -5.0. The load on the right point has a value of 1.0. Make sure that the LCID is 1 for both loads.
 - In the upper-left menu go to → File → Save As
 - Save the file in the same location as the Optimizer 2D program. Save the file as snail.k (see Figure B.4)
 - Open the Optimizer 2D program in Matlab.
 - Enter the file name as 'snail.k'
 - Set obj = 1; alg = 7; fea = 1; int = 1;
 - Set E0 = 1; Emin = 1e-9; nu = 0.3;

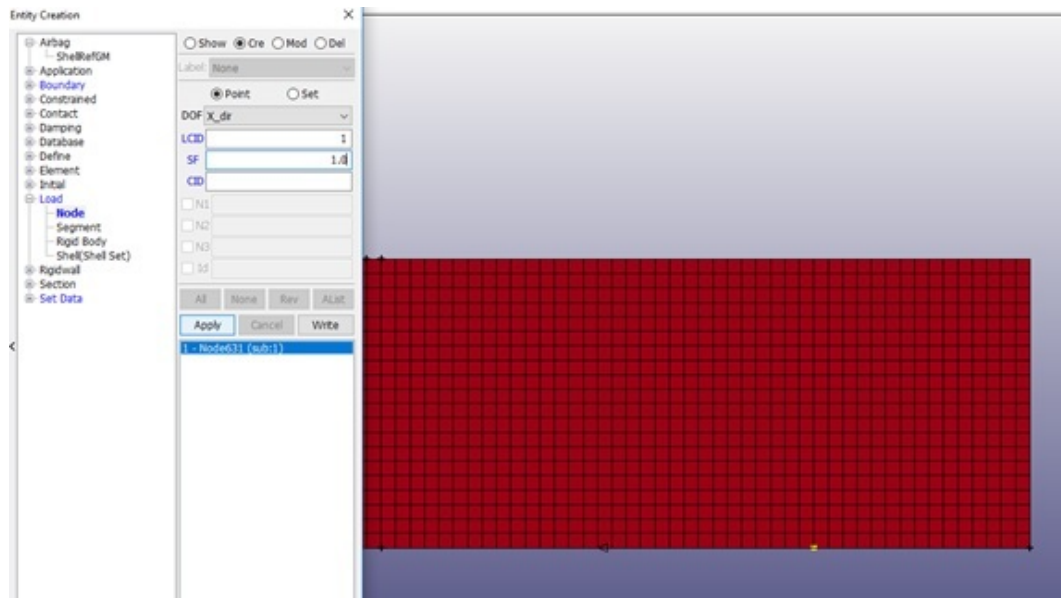


Figure B.3. Apply Loads in the X direction

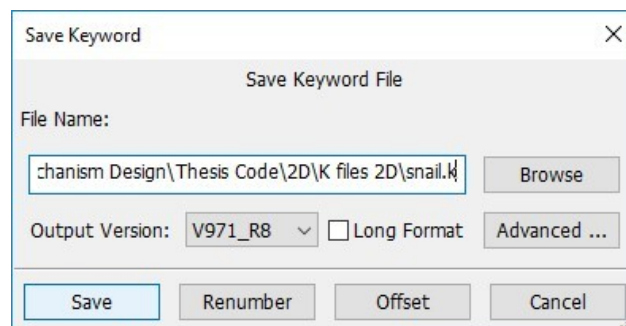


Figure B.4. Save As snail.k

- Set $VF = 0.5$; $penal = 3$; $rmin = 2$; $tolx = 1e-3$; $maxloop = 1000$; $displayflag = 1$;
- Run the Optimizer 2D script. The initial command-line output should look like Figure B.5.
 - When the problem has converged the termination command-line output should be displayed (Figure B.6), and the final topology should be visible (see Figures B.7 & B.8).
 - Open the folder 'OPTIMIZER_2D_OUTPUT_snail' to find saved output files.

```
Running 2D Optimizer on file: snail.k

Program Options:
obj = 1, Minimum Compliance (1 Load) Objective Function
alg = 7, Method of Moving Asymptotes Algorithm
fea = 1, Force-Driven FEA
int = 1, Volume Fraction Density Initialization

Material Properties:
E0 = 1, Emin = 1.0e-09, nu = 0.30

Optimization Parameters:
VF = 0.50, penal = 3, rmin = 2.0, tolX = 1.0e-03, maxloop = 1000

Output:
It.:  1 Obj.:  743.1961 Vol.:  0.426 ch.:  0.450
It.:  2 Obj.:  248.3896 Vol.:  0.468 ch.:  0.292
It.:  3 Obj.:  142.8515 Vol.:  0.488 ch.:  0.242
It.:  4 Obj.:  131.5821 Vol.:  0.495 ch.:  0.236
It.:  5 Obj.:  126.9476 Vol.:  0.496 ch.:  0.274
It.:  6 Obj.:  124.2844 Vol.:  0.497 ch.:  0.284
It.:  7 Obj.:  122.3503 Vol.:  0.497 ch.:  0.247
It.:  8 Obj.:  120.7349 Vol.:  0.497 ch.:  0.209
It.:  9 Obj.:  119.1043 Vol.:  0.498 ch.:  0.242
It.: 10 Obj.:  117.6772 Vol.:  0.498 ch.:  0.229
It.: 11 Obj.:  116.6794 Vol.:  0.499 ch.:  0.180
It.: 12 Obj.:  116.0023 Vol.:  0.499 ch.:  0.205
It.: 13 Obj.:  115.5109 Vol.:  0.499 ch.:  0.204
It.: 14 Obj.:  115.1415 Vol.:  0.499 ch.:  0.190
It.: 15 Obj.:  114.8190 Vol.:  0.499 ch.:  0.227
It.: 16 Obj.:  114.5368 Vol.:  0.499 ch.:  0.282
It.: 17 Obj.:  114.3148 Vol.:  0.500 ch.:  0.292
It.: 18 Obj.:  114.1476 Vol.:  0.499 ch.:  0.480
```

Figure B.5. Initial Command-Line Output for Snail Problem

```

It.: 247 Obj.: 111.5572 Vol.: 0.500 ch.: 0.002
It.: 248 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 249 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 250 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 251 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 252 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 253 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 254 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 255 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 256 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 257 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
It.: 258 Obj.: 111.5572 Vol.: 0.500 ch.: 0.001
Wrote 2952 facets to C:\Users\jnajmon\Documents\Thesis Research
Elapsed time is 22.702872 seconds.

```

Figure B.6. Termination Command-Line Output for Snail Problem

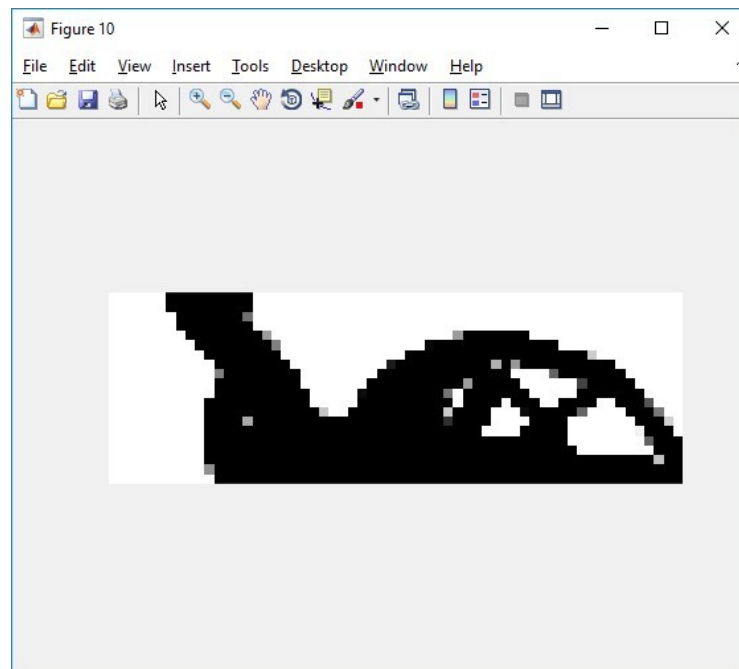


Figure B.7. Topology for Snail Problem

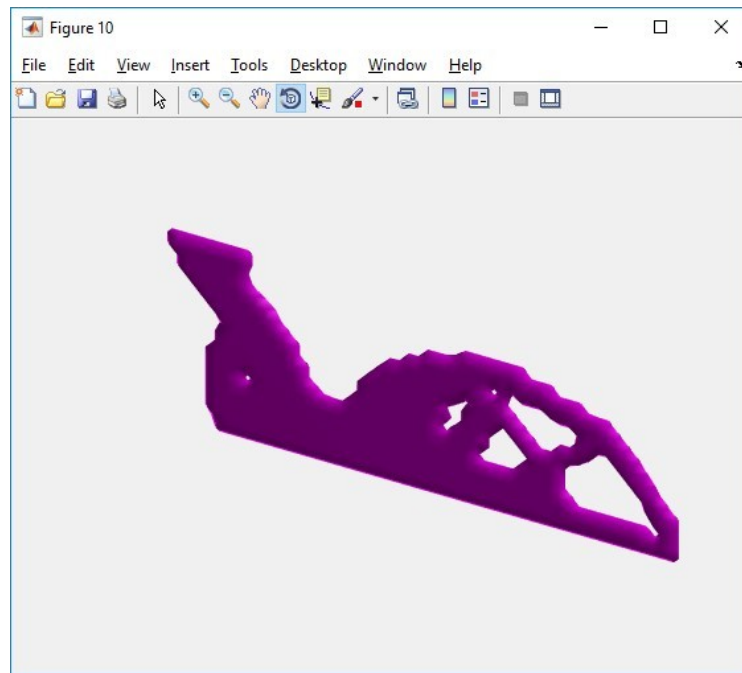


Figure B.8. 3D STL Iso-View of Snail Problem

B.2 Force Inverter Problem - Maximum Mutual Potential Energy

- Open LS-PrePost.
- In the upper-left menu go to → FEM → Element and Mesh → Shape Mesher
- Choose '4N Shell' in the Entity Menu.
- Enter the following data (Figure B.9) into the tool, to create a 40 x 20 2D mesh.
- In the upper-left menu go to → FEM → Model and Part → Create Entity
- Open the Boundary drop-down menu. Click on Spc (specific point constraint).
- Under the Create Tab (Cre), deselect the X, Z, RX, RY, and RZ constraints, and apply a fixed y degree of freedom constraint on the nodes along the top edge of the shell (roller support) shown in Figure B.10.
 - Fix the x and y degrees of freedom for the bottom left two nodes as shown in Figure B.10.
- Open the Load drop-down menu. click on Node.
 - Under the Create Tab (Cre), apply a load in the x-direction on the upper left node. Make sure LCID is 1. SF, the magnitude of the load, should also be 1. See Figure B.11.
 - Apply another load in the x-direction on the upper right node. Set LCID to 2. SF should be -1.
- In the upper-left menu go to → File → Save As

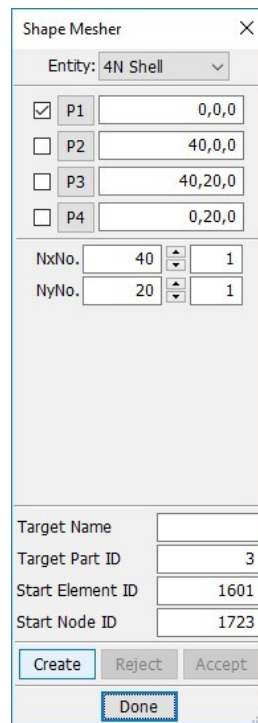


Figure B.9. Create 40 x 20 2D Mesh

- Save the file in the same location as the Optimizer 2D program. Save the file as forceinverter.k (see Figure B.12)
 - Open the Optimizer 2D program in Matlab.
 - Enter the file name as 'forceinverter.k'
 - Set obj = 3; alg = 7; fea = 1; int = 1;
 - Set E0 = 1; Emin = 1e-9; nu = 0.3;
 - Set VF = 0.3; penal = 3; rmin = 1.5; tolx = 1e-3; maxloop = 1000; displayflag = 1;
- Run the Optimizer 2D script. The initial command-line output should look like Figure B.13.
 - When the problem has converged the termination command-line output should be displayed (Figure B.14), and the final topology should be visible (see Figures B.15 & B.16).
 - Open the folder 'OPTIMIZER_2D_OUTPUT_forceinverter' to find saved output files.

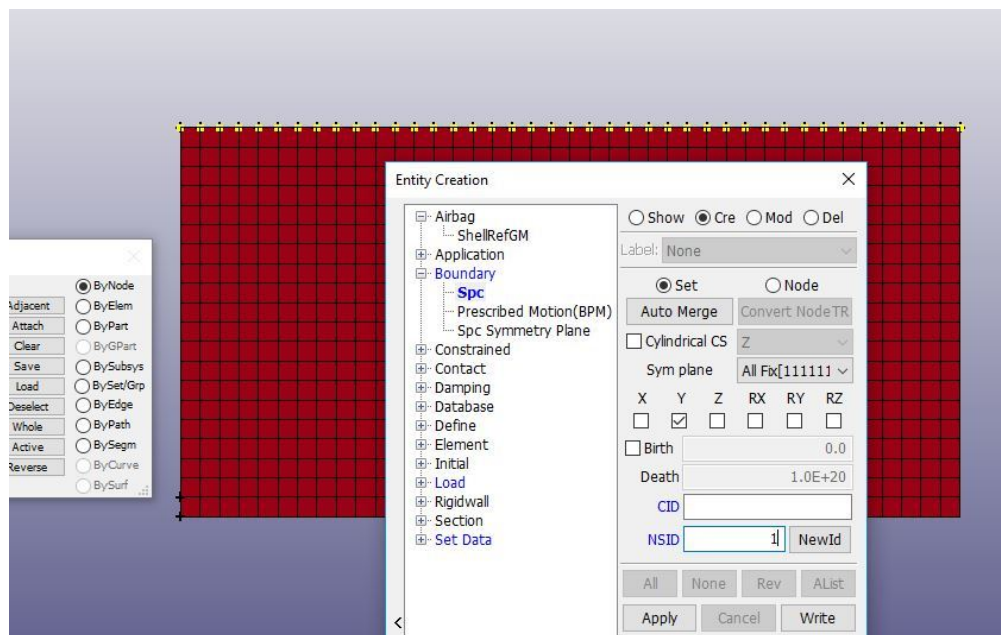


Figure B.10. Constrain Y Degrees of Freedom on Nodes Along the Top edge

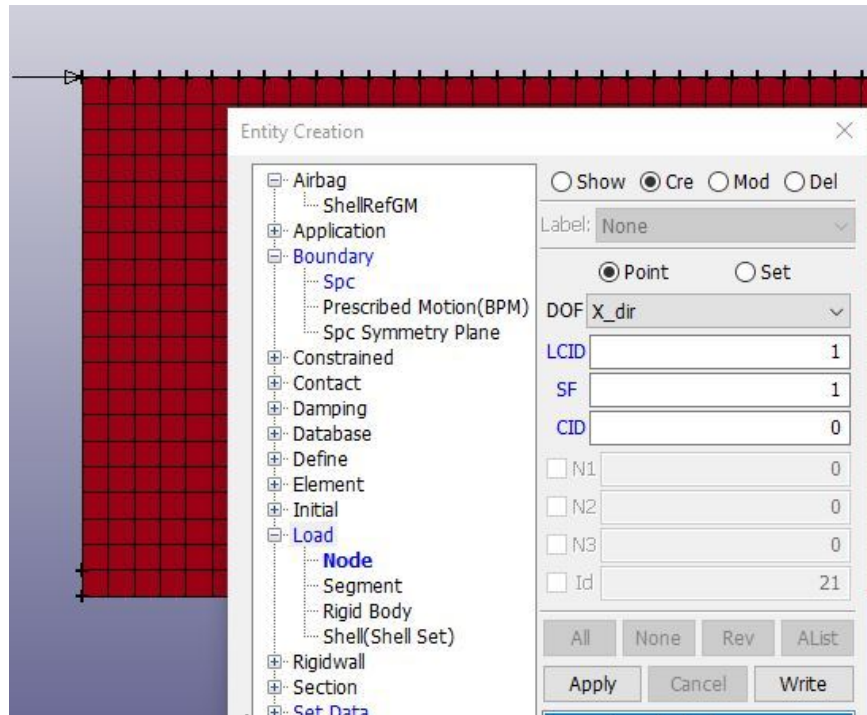


Figure B.11. Apply Inverter Loads in the X direction

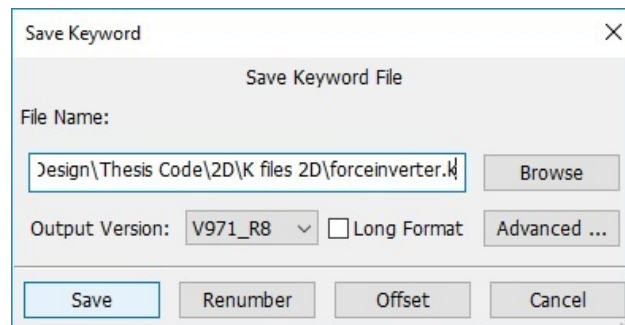


Figure B.12. Save As forceinverter.k

```
Running 2D Optimizer on file: forceinverter.k

Program Options:
obj = 3, Maximum Mutual Potential Energy Objective Function
alg = 7, Method of Moving Asymptotes Algorithm
fea = 1, Force-Driven FEA
int = 1, Volume Fraction Density Initialization

Material Properties:
E0 = 1, Emin = 1.0e-09, nu = 0.30

Optimization Parameters:
VF = 0.30, penal = 3, rmin = 1.5, tolx = 1.0e-03, maxloop = 1000

Output:
It.:  1 Obj.:  -9.6879 Vol.:  0.204 ch.:  0.429
It.:  2 Obj.: -10.0000 Vol.:  0.275 ch.:  0.153
It.:  3 Obj.:  -9.9328 Vol.:  0.290 ch.:  0.144
It.:  4 Obj.:  -9.9998 Vol.:  0.297 ch.:  0.071
It.:  5 Obj.:  -9.9813 Vol.:  0.298 ch.:  0.094
It.:  6 Obj.:  -9.9152 Vol.:  0.291 ch.:  0.266
It.:  7 Obj.:  -9.7801 Vol.:  0.259 ch.:  0.385
It.:  8 Obj.:  -9.7245 Vol.:  0.262 ch.:  0.539
It.:  9 Obj.:  -9.4981 Vol.:  0.273 ch.:  0.392
It.: 10 Obj.:  -9.1850 Vol.:  0.288 ch.:  0.257
```

Figure B.13. Initial Command-Line Output for Force Inverter Problem

```

It.: 49 Obj.: -8.6656 Vol.: 0.300 ch.: 0.018
It.: 50 Obj.: -8.6657 Vol.: 0.300 ch.: 0.014
It.: 51 Obj.: -8.6657 Vol.: 0.300 ch.: 0.010
It.: 52 Obj.: -8.6658 Vol.: 0.300 ch.: 0.006
It.: 53 Obj.: -8.6658 Vol.: 0.300 ch.: 0.004
It.: 54 Obj.: -8.6658 Vol.: 0.300 ch.: 0.003
It.: 55 Obj.: -8.6658 Vol.: 0.300 ch.: 0.002
It.: 56 Obj.: -8.6659 Vol.: 0.300 ch.: 0.002
It.: 57 Obj.: -8.6659 Vol.: 0.300 ch.: 0.001
It.: 58 Obj.: -8.6659 Vol.: 0.300 ch.: 0.002
It.: 59 Obj.: -8.6659 Vol.: 0.300 ch.: 0.001
Wrote 1360 facets to C:\Users\jnajmon\Documents\Thesis Research
Elapsed time is 10.773527 seconds.

```

Figure B.14. Termination Command-Line Output for Force Inverter Problem

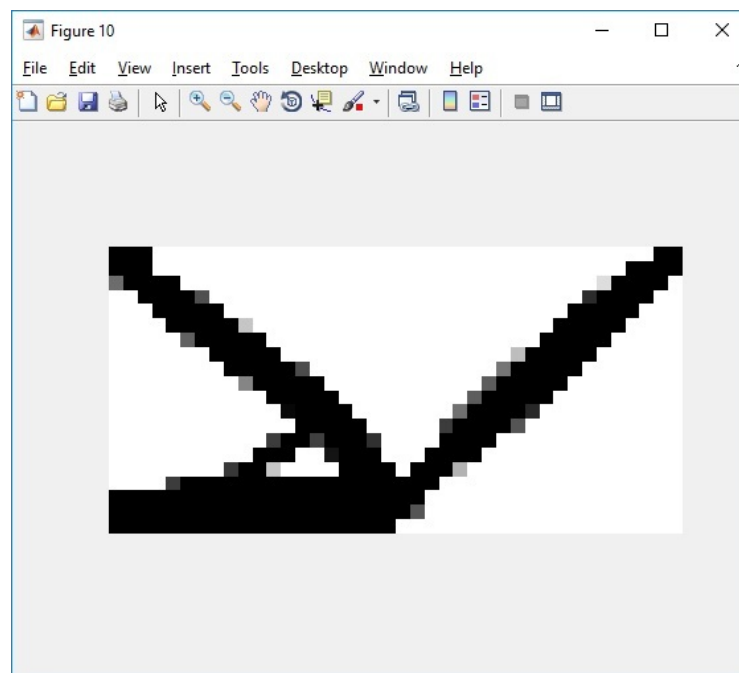


Figure B.15. Topology for Force Inverter Problem

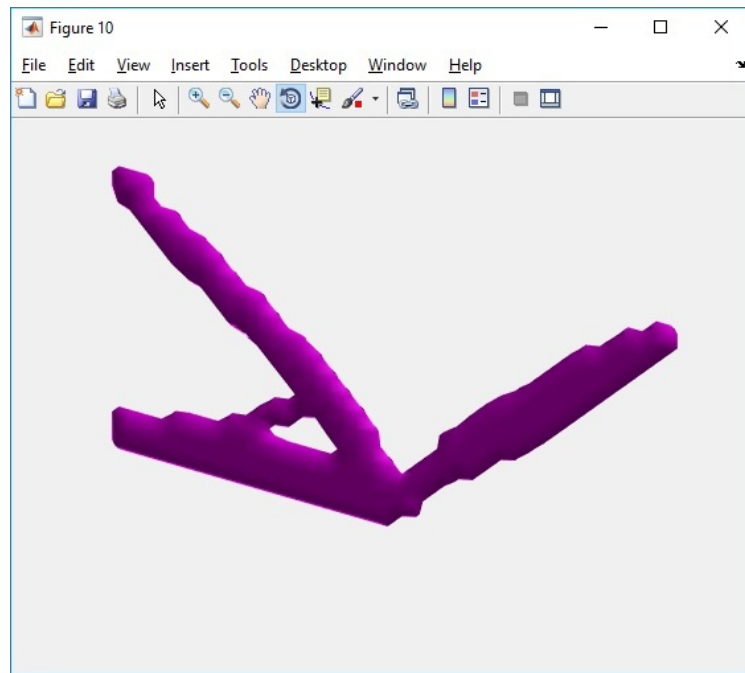


Figure B.16. 3D STL Iso-View of Force Inverter Problem

C. 3D EXAMPLES

C.1 Two Load Case Problem - Minimum Compliance, 2 Load Cases

- Open LS-PrePost.
- In the upper-left menu go to → FEM → Element and Mesh → Shape Mesher
- Choose 'Box Solid' in the Entity Menu.
- Enter the following data (Figure C.1) into the tool, to create a 60 x 60 x 4 3D mesh.

Entity: Box Solid	
<input checked="" type="radio"/> Region <input type="radio"/> Define Box	
PMin	PMax
X: 0	60
Y: 0	60
Z: 0	4
<input checked="" type="radio"/> Number <input type="radio"/> Size	
Vx: 60	1
Vy: 60	1
Vz: 4	1
Gap:	
Target Name	
Target Part ID	1
Start Element ID	14401
Start Node ID	18606
<input type="button" value="Create"/> <input type="button" value="Reject"/> <input type="button" value="Accept"/>	
<input type="button" value="Create mesh for entity"/>	

Figure C.1. Create 60 x 60 x 4 3D Mesh

- In the upper-left menu go to → FEM → Model and Part → Create Entity
- Open the Boundary drop-down menu. Click on Spc (specific point constraint).

- Under the Create Tab (Cre), deselect the RX, RY, and RZ constraints, and apply a fixed x, y, and z degree of freedom constraint on the left face nodes shown in Figure C.2.

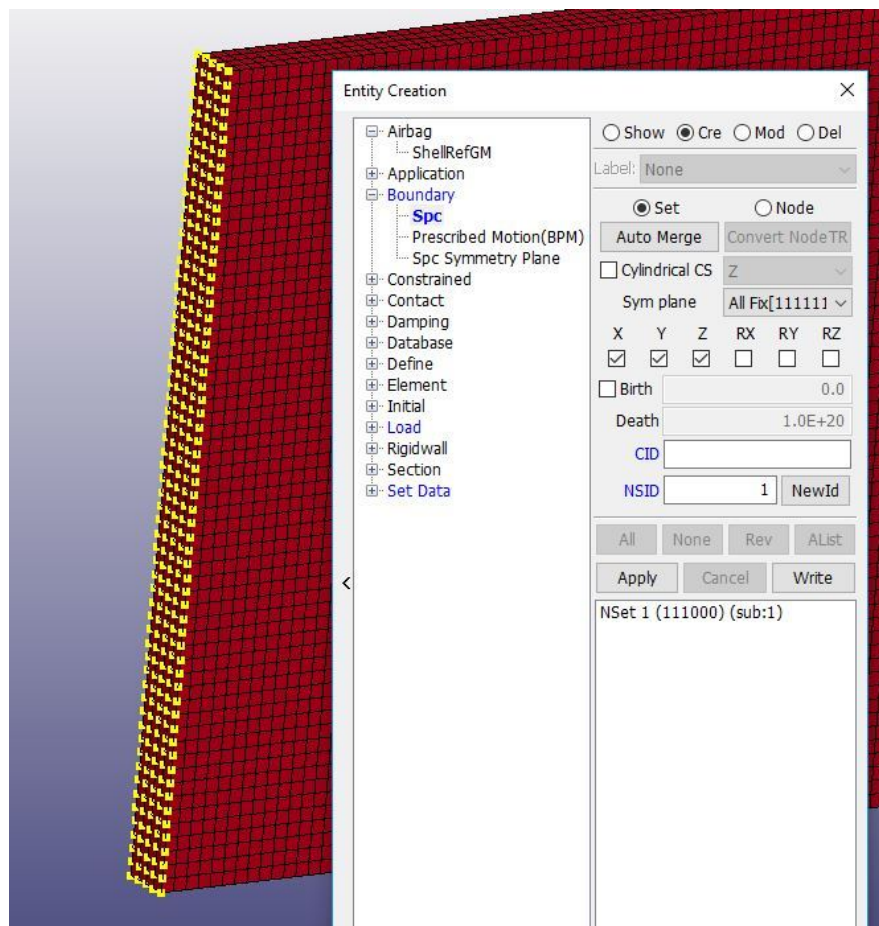


Figure C.2. Constrain X, Y, and Z Degrees of Freedom on Left Face Nodes

- Open the Load drop-down menu. click on Node.
- Under the Create Tab (Cre), apply the first load case (LCID = 1) load on the bottom right corner nodes in the negative y-direction with a magnitude of 1. Apply an opposite, second load case (LCID = 2), on the top right corner nodes in the positive y-direction with a magnitude of 2. Figure C.3 shows the applied loads.
 - In the upper-left menu go to → File → Save As
 - Save the file in the same location as the Optimizer 3D program. Save the file as twoload.k (see Figure C.4)
 - Open the Optimizer 3D program in Matlab.

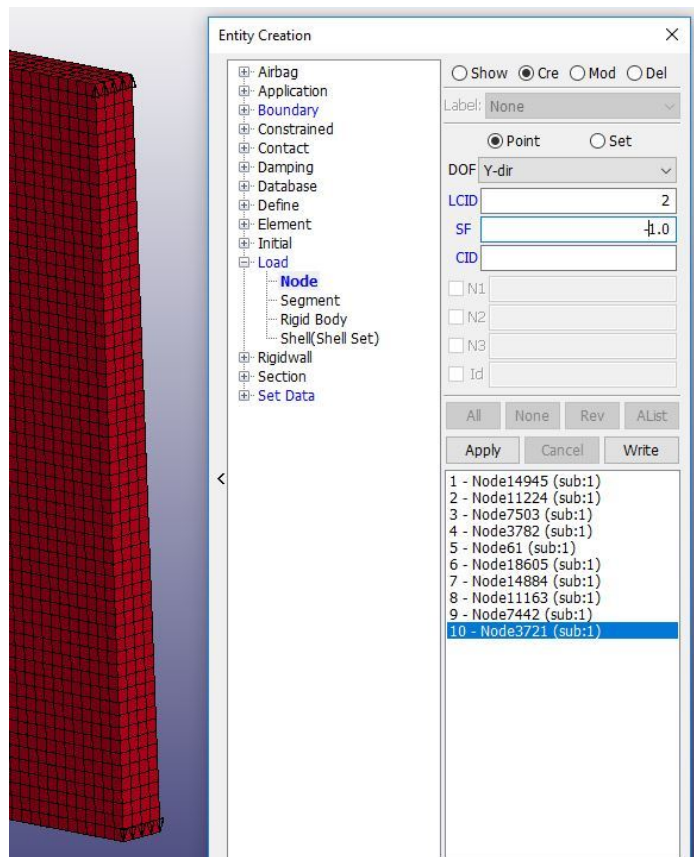


Figure C.3. Apply Two Opposing Load Cases in Y-Direction

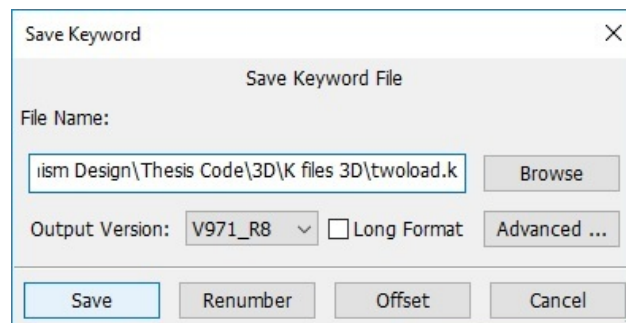


Figure C.4. Save As twoload.k

- Enter the file name as ‘twoload.k’
- Set obj = 2; alg = 7; fea = 1; int = 1;
- Set E0 = 1; Emin = 1e-9; nu = 0.3;
- Set VF = 0.4; penal = 3; rmin = 1.5; tolx = 1e-2; maxloop = 1000; displayflag = 1;
- Run the Optimizer 3D script. The initial command-line output should look like Figure C.5.

```

Running 3D Optimizer on file: twoload.k

Program Options:
obj = 2, Minimum Compliance (2 Loads) Objective Function
alg = 7, Method of Moving Asymptotes Algorithm
fea = 1, Force-Driven FEA
int = 1, Volume Fraction Density Initialization

Material Properties:
E0 = 1, Emin = 1.0e-09, nu = 0.30

Optimization Parameters:
VF = 0.40, penal = 3, rmin = 1.5, tolx = 1.0e-02, maxloop = 1000

Output:
It.:  1 Obj.: 1561.4122 Vol.: 0.375 ch.: 0.444
It.:  2 Obj.:  952.6234 Vol.: 0.390 ch.: 0.217
It.:  3 Obj.:  693.7981 Vol.: 0.393 ch.: 0.238
It.:  4 Obj.:  577.1914 Vol.: 0.395 ch.: 0.223
It.:  5 Obj.:  501.4438 Vol.: 0.396 ch.: 0.227
It.:  6 Obj.:  446.7369 Vol.: 0.396 ch.: 0.235
It.:  7 Obj.:  400.7806 Vol.: 0.396 ch.: 0.226
It.:  8 Obj.:  359.0607 Vol.: 0.394 ch.: 0.238
It.:  9 Obj.:  317.9033 Vol.: 0.393 ch.: 0.294
It.: 10 Obj.:  278.9051 Vol.: 0.394 ch.: 0.347

```

Figure C.5. Initial Command-Line Output for Two Load Case Problem

- When the problem has converged the termination command-line output should be displayed (Figure C.6), and the final topology should be visible (see Figures C.7 & C.8).
- Open the folder ‘OPTIMIZER_3D_OUTPUT_twoload’ to find saved output files.

```
It.: 246 Obj.: 199.9530 Vol.: 0.400 ch.: 0.034
It.: 247 Obj.: 199.9529 Vol.: 0.400 ch.: 0.034
It.: 248 Obj.: 199.9529 Vol.: 0.400 ch.: 0.034
It.: 249 Obj.: 199.9529 Vol.: 0.400 ch.: 0.035
It.: 250 Obj.: 199.9528 Vol.: 0.400 ch.: 0.037
It.: 251 Obj.: 199.9528 Vol.: 0.400 ch.: 0.040
It.: 252 Obj.: 199.9527 Vol.: 0.400 ch.: 0.044
It.: 253 Obj.: 199.9526 Vol.: 0.400 ch.: 0.050
It.: 254 Obj.: 199.9526 Vol.: 0.400 ch.: 0.057
It.: 255 Obj.: 199.9525 Vol.: 0.400 ch.: 0.022
It.: 256 Obj.: 199.9524 Vol.: 0.400 ch.: 0.005
Wrote 11124 facets to C:\Users\jnajmon\Documents\Thesis Research
Elapsed time is 3063.832371 seconds.
```

Figure C.6. Termination Command-Line Output for Two Load Case Problem

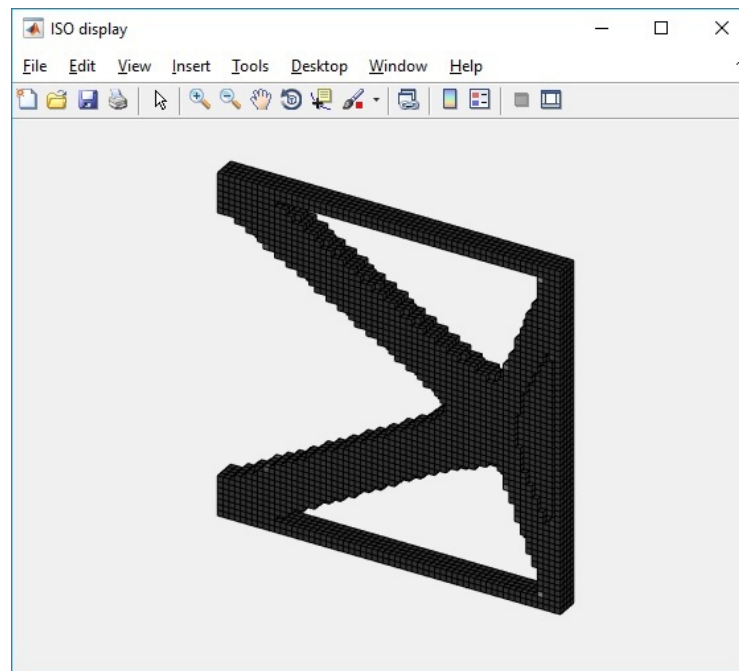


Figure C.7. Topology for Two Load Case Problem

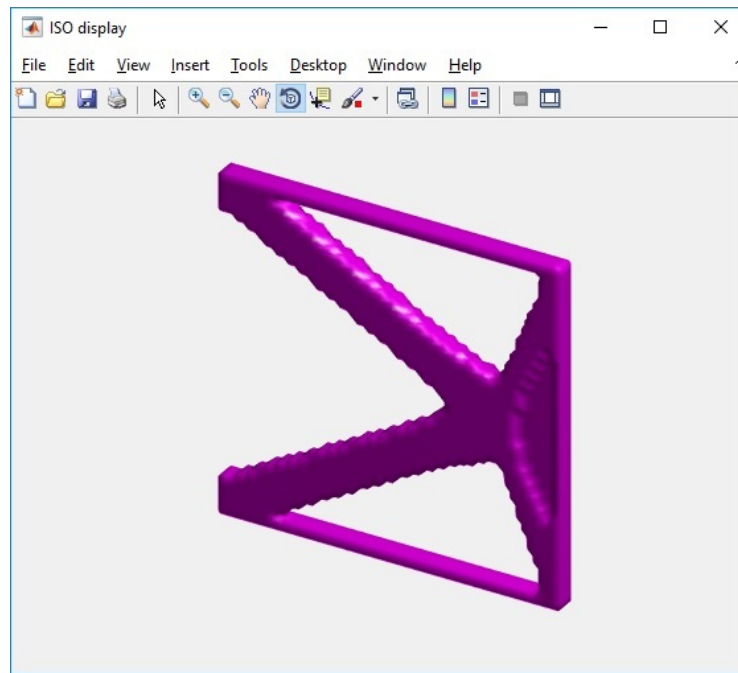


Figure C.8. 3D STL Iso-View of Two Load Case Problem

C.2 Positive Poisson's Ratio Mechanism Problem - Maximum Mutual Potential Energy

- Open the Optimizer 3D program in Matlab.
- Set $\text{obj} = 4$; $\text{alg} = 7$; $\text{fea} = 1$; $\text{int} = 1$;
- Set yzsym , xysym , and xzsym to true;
- Set $E0 = 1$; $E_{\text{min}} = 1e-9$; $\text{nu} = 0.3$;
- Set $\text{VF} = 0.2$; $\text{penal} = 3$; $\text{rmin} = 2$; $\text{tolx} = 1e-2$; $\text{maxloop} = 1000$; $\text{displayflag} = 1$;
- Set $x = 20$; $y = 20$; $z = 20$; $\text{pr} = 0.5$; $s = 4$;
- Run the Optimizer 3D script. The initial command-line output should look like Figure C.9.
 - When the problem has converged the termination command-line output should be displayed (Figure C.10), and the final topology should be visible (see Figures C.11 & C.12).
 - Open the folder 'OPTIMIZER_3D_OUTPUT_PoissonsRatio3d' to find saved output files.

Running 3D Optimizer on file: PoissonsRatio3d.k

Program Options:

obj = 4, Poissons Ratio Tailoring Mode
 alg = 7, Method of Moving Asymptotes Algorithm
 fea = 1, Force-Driven FEA
 int = 1, Volume Fraction Density Initialization

Material Properties:

E0 = 1, Emin = 1.0e-09, nu = 0.30

Optimization Parameters:

VF = 0.20, penal = 3, rmin = 2.0, tolx = 1.0e-02, maxloop = 1000

Poissons Ratio Parameters:

nelx = 20, nely = 20, nelz = 20, pr = 0.50, s = 4

Output:

It.:	1	Obj.:	-4.9990	Vol.:	0.200	ch.:	0.103
It.:	2	Obj.:	-4.9988	Vol.:	0.200	ch.:	0.040
It.:	3	Obj.:	-4.9987	Vol.:	0.200	ch.:	0.025
It.:	4	Obj.:	-4.9986	Vol.:	0.200	ch.:	0.022
It.:	5	Obj.:	-4.9986	Vol.:	0.200	ch.:	0.027
It.:	6	Obj.:	-4.9984	Vol.:	0.200	ch.:	0.047
It.:	7	Obj.:	-4.9981	Vol.:	0.200	ch.:	0.032
It.:	8	Obj.:	-4.9981	Vol.:	0.200	ch.:	0.022
It.:	9	Obj.:	-4.9982	Vol.:	0.200	ch.:	0.051
It.:	10	Obj.:	-4.9981	Vol.:	0.199	ch.:	0.086

Figure C.9. Initial Command-Line Output for Positive Poisson's Ratio Mechanism Problem

```

It.: 356 Obj.: -4.9714 Vol.: 0.200 ch.: 0.012
It.: 357 Obj.: -4.9713 Vol.: 0.200 ch.: 0.015
It.: 358 Obj.: -4.9712 Vol.: 0.200 ch.: 0.012
It.: 359 Obj.: -4.9712 Vol.: 0.200 ch.: 0.028
It.: 360 Obj.: -4.9712 Vol.: 0.200 ch.: 0.012
It.: 361 Obj.: -4.9711 Vol.: 0.200 ch.: 0.016
It.: 362 Obj.: -4.9710 Vol.: 0.200 ch.: 0.011
It.: 363 Obj.: -4.9710 Vol.: 0.200 ch.: 0.017
It.: 364 Obj.: -4.9710 Vol.: 0.200 ch.: 0.014
It.: 365 Obj.: -4.9709 Vol.: 0.200 ch.: 0.017
It.: 366 Obj.: -4.9708 Vol.: 0.200 ch.: 0.009
Wrote 16008 facets to C:\Users\jnajmon\Documents\Thesis Research
Elapsed time is 2671.814612 seconds.

```

Figure C.10. Termination Command-Line Output for Positive Poisson's Ratio Mechanism Problem

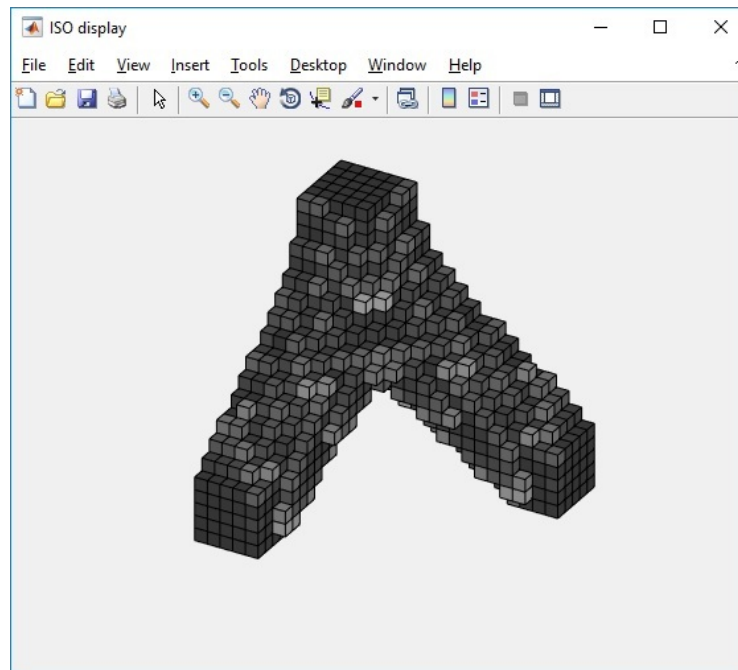


Figure C.11. Topology for Positive Poisson's Ratio Mechanism Problem

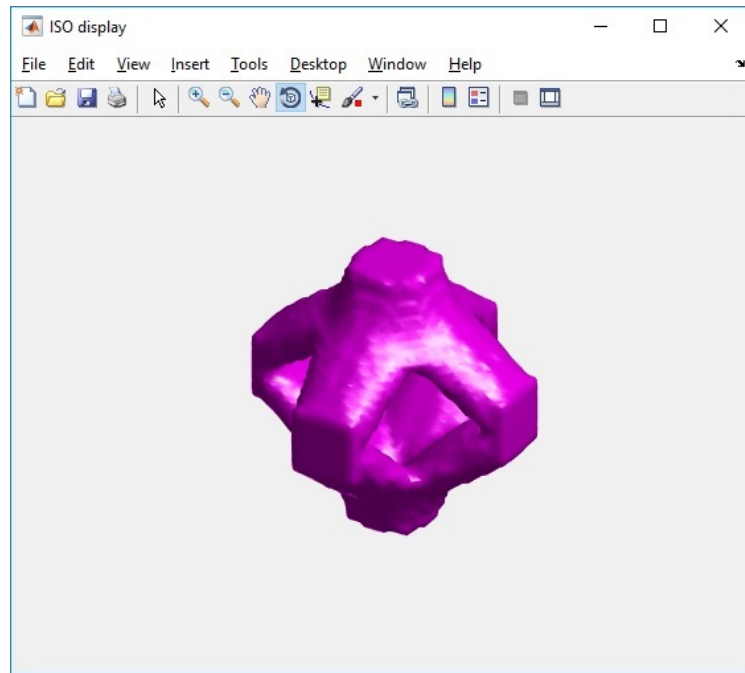


Figure C.12. 3D STL Iso-View of Positive Poisson's Ratio Mechanism Problem

D. LS-DYNA MATERIAL MODEL KEYCARD DETAILS

D.1 Material Model Parameters for the Helmet Liner

LS-Dyna material model keycard details for helmet liner ballistic tests can be found in Tables D.1, D.2, D.3, and D.4. Keycard parameters that are not in the table were left at default values. LS-Dyna simulations were ran with base units of g, mm, ms.

Table D.1. Polycarbonate Shell Material Keycard

MAT 001 - ELASTIC	RO	E	PR
	0.012	2390	0.37

Table D.2. Impactor Material Keycard

MAT 020 - RIGID	RO	E	PR	CON1
	0.045	2390	0.37	3

Table D.3. Natural Rubber Material Keycard

MAT 027 - MOONEY-RIVLIN RUBBER	Shore A Hardness	RO	PR	A	B
	50A	0.012	0.4999	0.302	0.076
	60A	0.012	0.4999	0.474	0.118
	70A	0.012	0.4999	0.736	0.184
	80A	0.012	0.4999	1.038	0.260

Table D.4. Expanded Polypropylene Foam Material Keycard

MAT 057 - LOW DENSITY FOAM	RO	E	TC	HU	DAMP	SHAPE
	8.6e-5	1	1e+10	0.2	0.1	5

E. 3D VERSIONS OF THE POSITIVE AND NEGATIVE MECHANISMS

Positive Mechanism

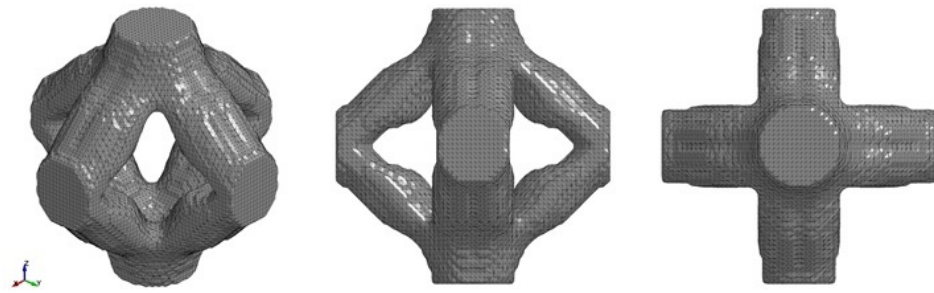


Figure E.1. 3D version of positive mechanism.

Negative Mechanism

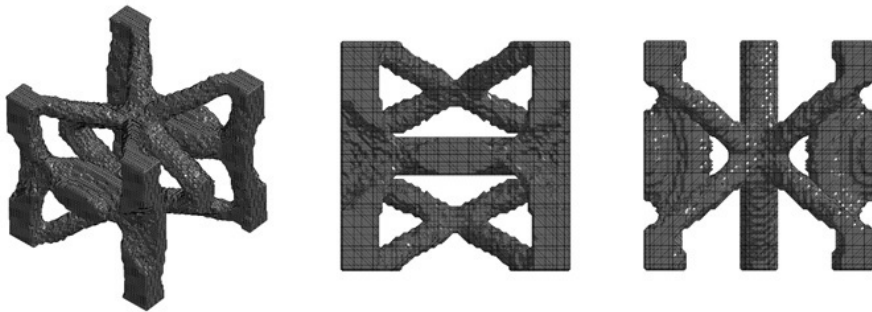


Figure E.2. 3D version of negative mechanism.

F. OPTIMIZER 2D MAIN MATLAB CODE

Optimizer 2D Main Function MATLAB Code

This is the main function MATLAB code for Optimizer 2D, version 1.6a. It is provided in this paper for reference. To obtain a full copy of Optimizer 2D and/or Optimizer 3D please email: jnajmon@iupui.edu or tovara@iupui.edu.

```

1  % Optimizer 2D (Version 1.6a) (Nov 2017)
2  % The main function file of Optimizer 2D, written by Joel Najmon
3  % This is a topology optimization program that can optimize structures for
4  % minimum compliance (one and two loads) and compliant mechanisms.
5  % Problem setup is done with LS-PrePost. This program reads the .k file and
6  % will save all outputs after a solution is reached. Early termination can
7  % be triggered by pressing the 'Esc' key while the output figure window is
8  % active.
9  % The user can select from 7 different algorithms to use. The type of
10 % finite element method can be also selected (force-driven or displacement
11 % driven). The way that densities are initialized is also user-defined.
12 % Other program options are detailed in the Appendix of the corresponding
13 % Master's Thesis for this code, written by Joel Najmon.
14 close all; clear; clc; tic;
15
16 % ===== PROGRAM INPUTS/OPTIONS =====
17
18 % INPUT FILE
19 fname = 'cantileverbeam2d.k';
20 %For PR problem, fname is ignored. Problem parameters are defined below
21
22 % SELECT OBJECTIVE FUNCTION TO USE
23 % 1 = Minimum Compliance (1 Load) (DEFAULT)
24 % 2 = Minimum Compliance (2 Load)
25 % 3 = Maximum Mutual Potential Energy (Compliant Mechanism)
26 % 4 = Poisson's Ratio Tailoring Problem
27 obj = 1;
28
29 % SELECT ALGORITHM TO USE
30 % 1 = Optimality Criterion
31 % 2 = Interior-Point (fmincon)
32 % 3 = Sequential Quadratic Programming (fmincon)
33 % 4 = Sequential Quadratic Programming - Legacy (fmincon)
34 % 5 = Trust-Region-Reflective (fmincon)
35 % 6 = Active-Set (fmincon)
36 % 7 = Method of Moving Asymptotes (DEFAULT)
37 alg = 7;
38
39 % SELECT FINITE ELEMENT METHOD
40 % 1 = Force-Driven FEA (DEFAULT)
41 % 2 = Displacement-Driven FEA
42 fea = 1;
43
44 % SELECT DESIGN VARIABLE INITIALIZATION
45 % 1 = Initialize densities with volume fraction (DEFAULT)

```

```

46 % 2 = Initialize densities with random numbers
47 int = 1;
48
49 % BUILD FULL STRUCTURE IF THE PROBLEM IS SYMMETRIC?
50 ysym = false; %(horizontal) Symmetry about y axis?
51 xsym = false; %(vertical) Symmetry about x axis?
52
53 % MATERIAL PROPERTIES
54 E0 = 1;
55 Emin = 1e-9;
56 nu = 0.3;
57
58 % OPTIMIZATION PARAMETERS
59 VF = 0.5;
60 penal = 3;
61 rmin = 3;
62 tolx = 1e-3;
63 maxloop = 1000;
64 displayflag = 1;
65
66 % POISSON'S RATIO PROBLEM PARAMETERS
67 x = 30; %Number of elements in the x-direction
68 y = 30; %Number of elements in the y-direction
69 pr = 0.5; %Poisson ratio objective
70 s = 15; %Number of elements on top and right side that are loaded.
71
72 % VERIFY INPUT OPTIONS ARE VALID
73 if obj~=1 && obj~=2 && obj~=3 && obj~=4
74     obj = 1; %Invalid obj entry, defaulting to minimum compliance 1 load
75 end
76 if alg~=1 && alg~=2 && alg~=3 && alg~=4 && alg~=5 && alg~=6 && alg~=7
77     alg = 7; %Invalid alg entry, defaulting to MMA
78 end
79 if fea~=1 && fea~=2
80     fea = 1; %Invalid fea entry, defaulting to force-driven FEA
81 end
82 if int~=1 && int~=2
83     int = 1; %Invalid int entry, defaulting to volume frac. intialization
84 end
85 if obj == 4
86     fname = 'PoissonsRatio2d.k'; %Name of output folder for PR problem
87     ysym = true;
88     xsym = true;
89 end
90
91 % ===== PROGRAM START =====
92
93 % CREATE OUTPUT FOLDER | SAVE COMMAND WINDOW
94 foutput = ['OPTIMIZER_2D_OUTPUT_' fname(1:(end-2))];
95 sname = [fname(1:(end-2)) '_obj' num2str(obj) '_alg' num2str(alg) '_fea'...
96 num2str(fea) '_int' num2str(int) '_vf' num2str(VF) '_rmin' num2str(rmin)];
97 if ~exist([cd '\ ' foutput], 'dir') %create output folder
98     mkdir([cd '\ ' foutput]);
99 end
100 diary off
101 if exist([cd '\ ' foutput '\ ' sname '_log.txt'])==2
102     delete([cd '\ ' foutput '\ ' sname '_log.txt']);
103 end
104 diary([cd '\ ' foutput '\ ' sname '_log.txt'])
105 if exist('temp.mat')==2 %#ok<*EXIST>
106     delete('temp.mat');
107 end
108
109 % PRINT RUN INFORMATION
110 fprintf(' Running 2D Optimizer on file: %s\n\n',fname);
111 fprintf(' Program Options:\n');

```

```

112 prog_options(obj,alg,fea,int)
113 fprintf(' Material Properties:\n E0 = %1d, Emin = %1.1e, nu = %1.2f\n\n'...
114         ,E0,Emin,nu);
115 fprintf([' Optimization Parameters:\n VF = %1.2f, penal = %1.1d, '...
116         'rmin = %1.1f, tolx = %1.1e, maxloop = %4d\n\n']...
117         ,VF,penal,rmin,tolx,maxloop);
118
119 % IMPORT ELEMENT, LOAD, AND SUPPORT DATA
120 if obj == 4
121     fprintf([' Poissons Ratio Parameters:\n nelx = %3d, nely = %3d, '...
122             'pr = %1.2f, s = %3d\n\n'],x,y,pr,s);
123     save('temp.mat','x','y','pr','s','E0','Emin','nu','VF','penal',...
124         'rmin','tolx','maxloop','displayflag','obj','int','fea')
125     pr_2d_import()
126 else
127     save('temp.mat','fname','E0','Emin','nu','VF','penal','rmin',...
128         'tolx','maxloop','displayflag','obj','int','fea')
129     lsp_2d_import()
130 end
131
132 % PREPARE FEA
133 if fea == 1
134     prepare_2d_fdfea()
135 elseif fea == 2
136     prepare_2d_ddfea()
137 end
138
139 % PREPARE FILTER
140 prepare_2d_filter()
141
142 % RUN OPTIMIZER
143 fprintf(' Output:\n');
144 if alg == 1
145     xopt = alg1_OC();
146 elseif alg == 2
147     xopt = alg2_IP();
148 elseif alg == 3
149     xopt = alg3_SQP();
150 elseif alg == 4
151     xopt = alg4_SQPL();
152 elseif alg == 5
153     xopt = alg5_TRR();
154 elseif alg == 6
155     xopt = alg6_AS();
156 elseif alg == 7
157     xopt = alg7_MMA();
158 end
159
160 if obj == 3
161     MA = mechadv(xopt);
162     GA = geoadv(xopt);
163     fprintf(' Compliant Mechanism Advantage Analysis:\n');
164     fprintf([' Mechanical Advantage: %2.3f'...
165             ' Geometric Advantage: %2.3f\n\n'],MA,GA);
166 end
167
168 % SAVE RESULTS
169 clf; display_2D(xopt);
170 if ysym == true
171     xopt = [xopt flip(xopt,2)];
172 end
173 if xsym == true
174     xopt = [flip(xopt); xopt];
175 end
176 save([cd '\ ' foutput '\ ' sname '.mat'], 'xopt')
177 Top3dSTL_v3([cd '\ ' foutput '\ ' sname '.stl'], xopt, 'Format', 'bin',...

```

```

178         'Mode', 'iso', 'FaceColor', 'm', 'Plot', true)
179     toc
180     diary off
181     delete temp.mat
182     % =====
183     % == This program was written by Joel Najmon and Andres Tovar, ==
184     % == Dept. of Mechanical Engineering, Indiana University - ==
185     % == Purdue University, Indianapolis Indiana, United States of America ==
186     % == -----
187     % == Please send your suggestions and comments to: jnajmon@iupui.edu ==
188     % == -----
189     % == The program was off of the existing MATLAB topology optimization ==
190     % == codes top88 and top3d. These codes can be found at ==
191     % == (http://www.topopt.dtu.dk/?q=node/751) and (https://top3dapp.com/) ==
192     % == -----
193     % == Please email jnajmon@iupui.edu or tovara@iupui.edu for the full ==
194     % == Optimizer 2D (or 3D) code. The corresponding Master's thesis, ==
195     % == Design of Compliant Mechanism Lattice Structures for Impact Energy ==
196     % == Absorption (includes program documentation and examples), can be ==
197     % == downloaded from (https://scholarworks.iupui.edu/handle/1805/199) ==
198     % == by searching for the title or author. ==
199     % == -----
200     % == Disclaimer: ==
201     % == The authors reserves all rights for the program. ==
202     % == The authors do not guarantee that the code is free from errors and ==
203     % == are not responsible for any event caused by the use of this code. ==

```