Modeling and Finite Element Analysis Methods for the Dynamic Crushing of Honeycomb Cellular Meso-Structures

Jesse Schultz
Clemson University, jesses@clemson.edu

Follow this and additional works at: https://tigerprints.clemson.edu/all_theses

Part of the Mechanical Engineering Commons

Recommended Citation
https://tigerprints.clemson.edu/all_theses/1106

This Thesis is brought to you for free and open access by the Theses at TigerPrints. It has been accepted for inclusion in All Theses by an authorized administrator of TigerPrints. For more information, please contact kokeefe@clemson.edu.
ABSTRACT

The effective static mechanical properties, such as the moduli of elasticity and rigidity and Poisson’s ratio, of honeycomb cellular meso-structures are capable of control due to variations of their cellular geometry. While the dynamic properties of these structures are a popular topic of research, there is a lack of both consistent modeling methods and generalizations in terms of honeycomb cellular geometry. In order to fill these gaps, this study presents a standard set of methods for the finite element analysis (FEA) of honeycomb cellular materials subject to dynamic loading conditions, as well as illustrates the effects of the cellular geometry parameters on a honeycomb structure’s response to non-static loads.

The first study performed compares the response of four different hexagonal honeycomb geometries to in-plane impact of varying velocities, which show different failure modes while maintaining a constant effective modulus in the loading direction.

The second section describes a newly developed design of experiments method for the simulation of honeycomb cellular materials that can efficiently gather sufficient data for the generalization of the relationships between cellular geometry and energy absorbed by the structure due to plastic deformation of the cells. This allows for the targeting of specific responses through the modification of cellular geometric parameters.

The final study discussed in this thesis discusses the simulation of models of reduced size in order to decrease the computational expense of the finite element analyses, while measuring error when compared to structures of larger numbers of cells.
This allows an analyst to determine the desired trade-off between time required to perform an analysis and accuracy of the results.
DEDICATION

I dedicate this thesis to the members of the Clemson Engineering Design Application and Research (CEDAR) lab. Your support, friendship, guidance, and knowledge have made my graduate experience valuable to my personal, professional, and academic aspects of my life. I could not imagine working with a better group of engineers. I hope this work provides value to your further research.
ACKNOWLEDGMENTS

First, I thank Michelin North America for providing my stipend funding while working on the NIST/ATP project. This allowed me to conduct this research in parallel to the other funded work.

I would like to thank the members of his advisory committee – Dr. Joshua Summers, for providing the funded research projects that spawned the topic for this thesis, and advising on proper research methods and intellectual growth. Dr. Lonny Thompson, for sharing his knowledge in solid mechanics and finite element methods, and asking hard-to-answer questions on fine details with every aspect of this research. Dr. Paul Joseph, who provided valuable input and answering questions regarding fundamental theory regarding the research of solid mechanics. Finally, Dr. Jaehyung Ju, for teaching me the fundamental mechanics of cellular materials and providing both extensive background literature and help in several technical areas.

I would also like to thank the student members of the Clemson Engineering Design Application and Research (CEDAR) lab. Each individual has aided me by providing valuable outside input and comments, as well as through friendship and moral support in stressful times.

Finally, thank you to my girlfriend Crista and to my parents, for standing by me and encouraging me during my two year pursuit of my degree.
TABLE OF CONTENTS

Title Page ......................................................................................................................... i
Abstract ............................................................................................................................... ii
Dedication ............................................................................................................................. iv
Acknowledgments ............................................................................................................... v
List of Tables ...................................................................................................................... viii
List of Figures .................................................................................................................... ix
Chapter 1: Static and Dynamic Analysis of Honeycomb Meso-Structures ...................... 1
Chapter 2: Motivation, Research Questions, and Thesis Overview ................................. 9
  2.1. Thesis Overview ....................................................................................................... 10
Chapter 3: Finite Element Modeling of Honeycomb Cellular Meso-Structures .............. 13
  3.1. Response Collection ................................................................................................. 18
  3.2. Validation of Simulation Method .............................................................................. 21
Chapter 4: Effects of Honeycomb Cellular Geometry on Dynamic Impact Response ... 26
  4.1. Honeycomb Cellular Geometries ............................................................................ 26
  4.2. Results ...................................................................................................................... 29
  4.3. Honeycomb Effectiveness ....................................................................................... 43
Chapter 5: Design of Experiments Method for the Targeting of Honeycomb Response to Dynamic Loading Conditions ................................................................. 48
  5.1. Objective Function .................................................................................................. 48
  5.2. Design of Experiment Results ............................................................................... 52
  5.3. Sensitivity Analysis of Design Variables ................................................................. 53
  5.4. Response Surface Model ......................................................................................... 56
  5.5. Optimization of Honeycomb Cellular Geometry .................................................. 61
  5.6. Improvements in Computational Efficiency ......................................................... 63
Chapter 6: Sizing of Honeycomb Models for the Computational Efficiency of Dynamic Simulations of Cellular materials

6.1. Methodology

6.2. Results

Chapter 7: Conclusions and Future Work

7.1. Experimental Validation

7.2. Future Applications

7.3. Application of Honeycomb Analysis Methods to Future Work

Works Cited

Appendices

Appendix A. ABAQUS Modeling Descriptions for Chapter 4 Simulations

A.1. Visual Descriptions of Results

Appendix B. Sequential Simulation Source Code

B.1. MATLAB Source Code
LIST OF TABLES

Table 1.1: Prefix definitions ................................................................................................. 1
Table 1.2: Previous research in the design and analysis of honeycomb structures .......... 4
Table 3.1: Total and incremental time steps for impact simulations .................................. 19
Table 4.1: In-Plane Effective Properties of the Designed Honeycombs ....................... 28
Table 4.2: Cellular collapse at low and high velocities ................................................. 37
Table 5.1: DOE results ........................................................................................................ 52
Table 5.2: Optimal honeycomb geometry parameters and error calculations for maximum energy absorption .................................................................................................................. 62
Table A-1: ABAQUS modeling parameters for all impact simulations .................. 87
Table A-2: Type A honeycomb ABAQUS modeling parameters ............................. 89
Table A-3: Type B honeycomb ABAQUS modeling parameters ............................... 89
Table A-4: Type C honeycomb ABAQUS modeling parameters ............................... 90
Table A-5: Type D honeycomb ABAQUS modeling parameters ............................... 90
LIST OF FIGURES

Figure 1.1: Honeycomb structures used for simulations: Nakamoto [12] (top) vs. Zou [15] (bottom) ................................................................. 5
Figure 1.2: Impact direction relative to honeycomb unit cell ........................................ 6
Figure 1.3: Disparities in crushing mechanisms due to variation in modeling parameters: Nakamoto et al [11] (left) vs. Zheng et al [10] (right) ....................................................... 7
Figure 2.1: Thesis organization flowchart .................................................................. 10
Figure 3.1: Physical Configuration of Impact Simulation .............................................. 14
Figure 3.2: Crushing of honeycomb structure without symmetry boundary condition .... 17
Figure 3.3: Schematic of honeycomb loading and support conditions ....................... 19
Figure 3.4: Published proximal end stress vs. strain for Type A honeycomb [15] ........ 23
Figure 3.5: Current proximal end stress vs. strain for Type A honeycomb .................. 23
Figure 3.6: Published distal end stress vs. strain for Type A honeycomb [15] .......... 24
Figure 3.7: Current distal end stress vs. strain for Type A honeycomb ...................... 24
Figure 4.1: Unit-Cell of Hexagonal honeycombs (θ=30°, 15° -15°, and -30°) ............ 27
Figure 4.2: Type A (left) and Type B (right) response to 1 m/s impact ....................... 31
Figure 4.3: Type C (left) and Type D (right) response to 1 m/s impact ....................... 32
Figure 4.4: Type A (left) and Type B (right) response to 10 m/s impact ...................... 33
Figure 4.5: Type C (left) and Type D (right) response to 10 m/s impact ..................... 34
Figure 4.6: Type A (left) and Type B (right) response to 100 m/s impact .................... 35
Figure 4.7: Type C (left) and Type D (right) response to 100 m/s impact .................... 36
Figure 4.8: Proximal (top) and distal (bottom) reaction forces for Type A honeycomb structure ........................................................................... 39
Figure 4.9: Honeycomb reaction forces to 1 m/s impact - proximal (top) and distal (bottom) ................................................................................. 41
Figure 4.10: Honeycomb reaction forces to 100 m/s impact - proximal (top) and distal (bottom) .............................................................................. 43
Figure 4.11: Specific energy absorption of four honeycomb cellular geometries ........ 46
Figure 5.1: Honeycomb unit cell and nomenclature ..................................................... 48
Figure 5.2: Algorithm for response surface creation .................................................. 51
Figure 5.3: Normal probability plot of $t_2$ from Kolomogorov-Smirnov test .............. 54
Figure 5.4: Normal probability plot of $\theta$ from Kolomogorov-Smirnoff test ................................ 55
Figure 5.5: Pareto effect chart ........................................................................................................ 56
Figure 5.6: Isight RBF model .......................................................................................................... 58
Figure 5.7: Cellular geometry providing maximum (left – Type X) and minimum (right – Type Y) plastic energy absorption .............................................................................. 58
Figure 5.8: Type X (left) and Type Y (right) honeycomb response to 100 m/s impact . 59
Figure 5.9: Type X and Type Y response to 100 m/s impact: Proximal (top) and Distal (bottom) reaction forces .......................................................... 61
Figure 6.1: Type A honeycomb structures used for sizing analysis ................................................. 66
Figure 6.2: Type D honeycomb structures used for sizing analysis ................................................. 67
Figure 6.3: Proximal (top) and distal (bottom) reaction forces for Type A honeycomb structures of different numbers of rows subject to 100 m/s impact ..................... 68
Figure 6.4: Normalized RMS error of reduced-size honeycomb structures compared to 41 row x 74 column honeycomb structure reaction force data ........................................... 70
Figure 6.5: Type A (left) and Type D (right) computational time required for simulations ............... 71
Figure 6.6: Scaled proximal (top) and distal (bottom) reaction forces for Type A honeycomb subject to 100 m/s impact ........................................................ 72
Figure 7.1: ME 401 experimental apparatus prototype ................................................................. 77
Figure 7.2: Hopkinson bar apparatus [46] ....................................................................................... 79
Figure 7.3: Deformation waves in Type D structure subject to 2 m/s impact ....................... 80
Figure A- 1: Type A response to 100 m/s impact ............................................................................ 93
Figure A- 2: Type B response to 100 m/s impact ............................................................................ 94
Figure A- 3: Type C response to 100 m/s impact ............................................................................ 95
Figure A- 4: Type D response to 100 m/s impact ............................................................................ 96
Figure A- 5: Type A response to 50 m/s impact ............................................................................. 97
Figure A- 6: Type B response to 50 m/s impact ............................................................................. 98
Figure A- 7: Type C response to 50 m/s impact .............................................................................. 99
Figure A- 8: Type D response to 50 m/s impact .......................................................................... 100
Figure A- 9: Type A response to 10 m/s impact ............................................................................. 101
Figure A- 10: Type B response to 10 m/s impact ......................................................................... 102
Figure A- 11: Type C response to 10 m/s impact ......................................................................... 103
Figure A- 12: Type D response to 10 m/s impact ......................................................................... 104
Figure A-13: Type A response to 2 m/s impact ......................................................... 105
Figure A-14: Type B response to 2 m/s impact ......................................................... 106
Figure A-15: Type C response to 2 m/s impact ......................................................... 107
Figure A-16: Type D response to 2 m/s impact ......................................................... 108
Figure A-17: Type A response to 1 m/s impact ......................................................... 109
Figure A-18: Type B response to 1 m/s impact ......................................................... 110
Figure A-19: Type C response to 1 m/s impact ......................................................... 111
Figure A-20: Type D response to 1 m/s impact ......................................................... 112
CHAPTER 1: STATIC AND DYNAMIC ANALYSIS OF HONEYCOMB MESO-STRUCTURES

Honeycomb cellular meso-structures have shown the potential for forward engineering design, that is, the development of new products and solutions to design problems, through the control of their cellular geometry. The modification of the meso-geometry allows for the control of the effective, or meta, properties of the entire structure. Table 1.1 defines these prefixes and provides examples of the variables within their domain in this research.

Table 1.1: Prefix definitions

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meta-</td>
<td>The effective properties on a large scale that are controlled by lesser-order parameters</td>
<td>Effective moduli, effective Poisson’s ratio, crushing mechanisms</td>
</tr>
<tr>
<td>Meso-</td>
<td>The intermediate level of parameters that are defined to control higher-order properties</td>
<td>Cell height, cell length, cell wall thicknesses, cell angle</td>
</tr>
<tr>
<td>Micro-</td>
<td>Constitutive properties that are defined by the host material of the honeycomb</td>
<td>Host material density, moduli, Poisson’s ratio</td>
</tr>
</tbody>
</table>

The effective static properties of a honeycomb structure, for example, the moduli in various directions and Poisson’s ratios or shear strains, can be controlled through changing of the cell angle, cell wall heights and lengths, or cell wall thicknesses. Ju et al [1] has shown that multiple, previously conflicting properties can be targeted through the manipulation of specific parameters. Similar studies confirm this ability via the
agreement between analytical models [2], finite element (FE) simulations [3], and physical experimentation [4].

Berglind, et al derived a new method for the design of honeycombs for targeting a high shear flexure [5], using newly defined cellular geometry parameters and showing their direct effects on the shear compliance of the structure. These studies showed that highly auxetic structures are capable of attaining high shear flexure while maintaining a constant volume filled by the honeycomb.

The cyclic energy loss of similar honeycomb meso-structures subjected to strains within their elastic domain has been quantified [6] for their use in the shear band of the Michelin Tweel® tire, showing that the geometry of low energy loss constituent materials, such as aluminum can be controlled in such a way to mimic the flexibility of higher energy loss polymers or elastomers. These results, however, are still limited to quasi-static loading conditions that do not plastically deform the material.

The dynamic responses of honeycombs to impact have been studied in both the out-of-plane [7,8] and in-plane [9,10,11,12,13,14] regimes. Definitions of specific crushing mechanisms dependent on impact velocity have also been made, identifying three main crushing behaviors based upon both the wave trapping velocity and the critical wave speed of the honeycomb structure [15,16,17,18,19,20,21,22]. These identified crushing mechanisms include the formation of “X”-, “V”-, and “I”-shaped crushing bands, each forming in specific ranges of the velocity at which the structure is impacted. The influences of effective density of honeycomb imperfections and inclusions on crushing mechanisms have also been researched, showing that the response of a structure
is greatly influenced by the density of defects, especially at low and moderate impact velocities [18].

The direct effects of honeycomb cellular geometry, specifically auxetic geometry, that is, configurations leading to a negative Poisson’s ratio, is limited to the response to static loading conditions. Topics in this field include targeting of shear compliance and stress parameters through the modification to the cellular angle and thicknesses [23,3,1,6]. The effect of cell angle on the specific energy absorption, that is, energy absorbed per unit mass, has been quantified, but is limited to positive cell angles and quasi-static loading [16]. One goal of this research intends to fill the gap in literature that encompasses both positive and non-positive cell angles subject to dynamic loading conditions.

Table 1.2 summarizes several studies found in the field of the design and analysis of honeycomb cellular structures for both static and dynamic loading situations. There are several studies that relate simulations of varying impact velocities, both in-plane and out-of-plane, to experimental results, but do not consider variation in the honeycomb cellular geometry. As previously stated, several studies describe the behavior of honeycombs of varying cellular geometry, but are limited to static or quasi-static loading. Stress-strain relationships within the honeycomb structures are also heavily reported in several articles, a measure that is questioned in Chapter 4. Instead, force-displacement curves are used for analysis in this research. The six columns on the right side of the table are addressed in this paper because limited literature was found that simultaneously studies these topics and their relationships.
Table 1.2: Previous research in the design and analysis of honeycomb structures

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ajdari [9]</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Atli [16]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deqiang [8]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hu [17]</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ju [3,1,6]</td>
<td></td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Li [14]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lira [24]</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Liu [13]</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nakamoto [11,12]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattofatto [25,21]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sibeaud [22]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wang [7]</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wu [20]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yamashita [26]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yasui [27]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zarei [19]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zhang [18]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zheng [10]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zou [15]</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

There are several inconsistencies apparent when comparing literature in the research of the dynamic behaviors of honeycomb meso-structures. Different finite element (FE) simulation techniques are used to model similar scenarios, making the
direct comparison between multiple studies both difficult and unreliable. For example, while some studies analyze the response of honeycombs with two-dimensional FE simulations using beam elements [11,12,16], others use three-dimensional shell elements [9,10,15] to evaluate very similar physical problems.

A second irregularity noticed while investigating literature in this field is the size of the honeycomb structure in terms of number of cells appropriate to accurately gather their effective material properties. Several studies [9,11,12,14,18] simulate the response of square sections of honeycomb, while others [8,13,15,17] investigate the behavior of more slender rectangular structures, as seen in Figure 1.1.

![Figure 1.1: Honeycomb structures used for simulations: Nakamoto [12] (top) vs. Zou [15] (bottom)](image)

While each study justifies the models simulated with valid physical problems, the modeling of a standard physical problem of a honeycomb meso-structure will be beneficial for comparison across various cellular specifications. While almost all research
found analyzes the honeycombs in the same direction relative to the unit cell, that is, perpendicular to the flat faces of the hexagon, as seen in Figure 1.2, the disparity in the outer geometry and other modeling parameters of the structures analyzed shows variance in the visual representation of their crushing response.

![Figure 1.2: Impact direction relative to honeycomb unit cell](image)

Figure 1.3 illustrates this problem, showing different responses two structures of similar cellular geometry to the same velocity impact. As seen in the figure, the two crushing responses show deformation at opposite ends of the structure, as well as reversal in the shape of the crushing bands. The differences between the cellular parameters used in the two different studies make direct comparison between similar studies difficult without the presence of a generalizable response as a function of various cellular meso-parameters. The differences between the two studies shown in the figure include the yield strength of the host material, the number of cells in the non-loading direction, and cell wall thicknesses. These disparities between the two are yet more reasons for the need of a set of common modeling parameters when simulating the dynamic behavior of cellular materials, thus allowing for more direct comparison.
Figure 1.3: Disparities in crushing mechanisms due to variation in modeling parameters: Nakamoto et al [11] (left) vs. Zheng et al [10] (right)

Honeycombs have also shown a strong potential for impact absorption in both out-of-plane and in-plane scenarios, but there is a lack of both sufficient understanding and generalization for their design in such cases. The objective of this research is to analyze and study the trends in honeycomb crushing simulation experiments so that an effective method for designing honeycombs for in-plane impact absorption can be developed. With this method, honeycomb structures can be designed to not only achieve the maximum amount of energy absorption, but can be tailored to achieve a specific amount of energy absorption in addition to having other features that are needed for a given application. Honeycombs then can be designed to possess the characteristics needed for specific applications and requirements. Some of these applications may include but are not limited to automotive, aerospace and military applications in which honeycombs may serve as lightweight but still effective material for absorbing impact and improving crashworthiness design. In all these applications, lightweight is a desirable property due to its improvements on fuel efficiency and performance [28,29].
Extensive research has been done to increase the understanding and describe the response of honeycomb cellular meso-structures to impact, including qualitative and quantitative descriptions of the crushing mechanisms observed and reaction forces within the structures. The quantification of the effects of honeycomb cellular geometry on the dynamic response to impact is desired to allow for their forward design for such applications.

While the published studies found relevant to this topic are successful in describing the physical problems analyzed within the scope of their research, they either vaguely describe or do not describe at all the modeling parameters defined within the finite element simulations. This lack of information significantly increases the difficulty of both re-creating and advancing upon the published studies, generating a need for the publication of a fully defined standard set of FE modeling parameters, which is discussed in Chapter 3.
While the crushing behavior of honeycombs made with irregular cell shapes and variable cell wall thickness has been studied \[9,14\], limited literature is found that considers the response to dynamic loading of honeycombs made with cells of controlled, varying geometry, other than the standard positive angle hexagonal cells. The generalization of the responses of honeycomb structures of varying cellular geometries to impact is desired to allow for their use in the development of new applications and solutions through forward engineering design.

The benefits of filling these identified gaps in literature can be expanded to the application of honeycombs to engineering environments that require specific responses to impact while maintaining low weight of the structure. This can include, but is not limited to, the design and analysis of crushing regions in automotives, armor for military applications, and the design of a shear band for the Michelin Tweel \textregistered\ tire as a replacement for the current polyurethane layer that experiences high hysteretic energy loss and increased rolling resistance \[3\].

This research attempts to provide answers for the following questions in order to increase information on the dynamic properties of honeycomb cellular meso-structures for their use in forward engineering design.

RQ1. Does cellular geometry have a significant effect on the response of a honeycomb meso-structure to dynamic impact loading?
RQ2. How can specific responses of honeycomb cellular meso-structures be targeted using the modification of cellular geometry?

RQ3. How small of a model of honeycomb meso-structures can be linearly translated to large-scale models?

2.1. Thesis Overview

This report is organized as seen in the flow chart in Figure 2.1.

![Figure 2.1: Thesis organization flowchart](image)

Chapter 1 discusses several published studies in the design and analysis of honeycomb cellular meso-structures, and identifies the gap in the literature that this
research intends to fill. This chapter identifies the motivating factors for this research, including a summary of identified gaps in literature and specific applications that can benefit from filling them. Chapter 3 describes the finite element modeling parameters and methods that apply to the remaining chapters in this report. These methods are verified with the comparison between current and previously published results [15].

Chapter 4 discusses the methods, results, and analysis performed to provide a detailed solution to RQ1. Four different honeycomb meso-structures of different cellular geometries are simulated to experience impacts of varying velocities from an effectively rigid beam. The responses of the four structures to impacts of varying velocities are compared both qualitatively, in terms of the crushing mechanisms observed, and quantitatively, in terms of the reaction forces at the leading and trailing ends of the structure. The trends in these responses are described in detail, which illustrate the effects that the cellular geometric parameters have on a structure’s response to impact.

Chapter 5 describes in detail a newly developed method for the rapid simulation of several finite element models using cellular geometric parameters specified by a design of experiments to satisfy the requirements of RQ2. This method allows for the sequential analysis of any number of impact simulations, from which the results can be used to generate generalized models of the relationships between input parameters and response variables. Such relationships can then be used for the targeting of desired responses for specific engineering applications.

Chapter 6 discusses the efforts to describe the behavior of honeycomb structures of lower numbers of cells to decrease the computational expense of the finite element
simulations of their response to impact. In order to address RQ3, in this chapter, honeycomb structures of different numbers of rows of cells, capturing the reaction forces and comparing to larger models using a linear scaling method to determine the trade-off between simulation error and computer processing time.

The results are summarized, and conclusions of the advantages of these efforts are presented in Chapter 7. This chapter also discusses future work that can use the methods and results presented in this paper and the real-world engineering benefits of performing such studies.
CHAPTER 3 : FINITE ELEMENT MODELING OF HONEYCOMB CELLULAR MESO-STRUCTURES

A standard, verified set of finite element (FE) modeling parameters and methods is necessary for the ability to accurately record and draw conclusions from the responses of the honeycombs to dynamic loading. While several failed iterations and extensive troubleshooting were required to arrive at a final set of specifications and methods, this chapter outlines the final FE modeling methods and parameters used in the subsequent chapters of this report.

For all simulations described in this study, a honeycomb cellular structure is subject to an impact from an effectively rigid body with a constant velocity, as seen in Figure 3.1. Constant velocity can be assumed since the mass of the body, \( m \), is large, while the force applied, \( F \), in crushing the honeycomb is small, i.e, acceleration \( = \frac{F}{m} \approx 0 \). The initial simulation setup is a re-creation of a published study that analyzes the effects of impact velocity on a standard hexagonal honeycomb structure [15].
The back walls of the cells opposite from the impact are fixed in the loading (horizontal) direction, while allowing for sliding in the vertical direction to simulate the distal end being in contact with a rigid wall. All nodes in the model are fixed in the out-of-plane (out of the page) direction to prevent buckling. The terms “proximal” and “distal” will be used for the remainder of this document, referring to the impacted and fixed ends of the honeycomb structure, respectively.

ABAQUS v. 6.9-1 is used for generating, analyzing, and post-processing each model described in this report. The models are generated with a constant 21 rows of cells in the vertical direction and 74 columns of cells in the horizontal, with an out-of-plane depth of 4 mm, specifications also taken from a previously published study [15]. Three-dimensional, explicit S4R shell elements are used for the honeycomb structure, which are necessary because of its ability for self-contact in dynamic analyses, making them widely used in similar analyses to this research [8,9,10,13,14,15,17,18,24]. Two-dimensional beam elements, which are also commonly used in the quasi-static simulation of
honeycomb structures [11,12,16], are limited in their ability for self-contact in dynamic, explicit analyses in ABAQUS. The honeycomb cell walls are defined with a single cell with five integration points across its thickness. Three-dimensional, explicit C3D8R cubic elements are used for the impacting body, as it is a common element type used for rectangular-shaped parts [30].

3.1. Physical Model Assumptions and Boundary Conditions

Several assumptions are made for the simplification of the physical model to be simulated. These assumptions can then be translated to specific modeling parameters defined within the finite element simulations. The first assumption made is that the impacting object has a much higher stiffness than the effective stiffness of the honeycomb. As seen later in Table 4.1, the effective modulus of the honeycomb structures is on the order of 100 MPa, which is three orders of magnitude less than the modulus of most metals. This large difference validates modeling the impacting solid as a rigid member, as its deflections would be negligible when compared to the deflection of the honeycomb structures.

The second assumption made is that both the surfaces of the impacting solid and the honeycomb are smooth, meaning that there is negligible friction between all surfaces in the model. Because of this, there is no coefficient of friction defined in the finite element models described in this report.

A third assumption made in the simulations is that the mass of the impacting solid is much larger than the mass of the honeycomb meso-structure. This assumption, coupled with the first assumption that the effective stiffness of the honeycomb structure is much
less than the stiffness of the impacting solid, implies that the speed of the impacting solid will remain constant throughout the simulation. This assumption allows a simplification of the model, in which a mass-less impacting solid is given a constant velocity boundary condition throughout each simulation, which does not allow for any deformation within the part, making the solid effectively rigid. This boundary condition is applied to all nodes within the solid. ABAQUS can then calculate the force required to apply to these nodes to maintain the constant velocity of the impacting solid, regardless of its mass.

As also specified by [15], a 6061 Aluminum alloy is defined for the honeycomb constituent material, with Young’s Modulus $E = 68$ GPa, Poisson’s ration $\nu = 0.33$, and density $\rho = 2700$ kg/m$^3$. The material is modeled as elastic, perfectly plastic with yield stress $\sigma_y = 130$ MPa. The impacting body is given the mechanical properties of AISI 1020 steel, with $E = 210$ GPa, $\nu = 0.30$, and $\rho = 7800$ kg/m$^3$ [31]. The impacting body becomes effectively rigid, however, because of the constant velocity applied to all the nodes within the body, which does not allow for any deformation. These parameters are simply defined because of restrictions within ABAQUS. It was found that neither the mass nor the stiffness of the impacting bar showed effects on the results of the simulations.

The outer wall of the distal column of cells is fixed in the x-direction with freedom to slide vertically in the y-direction. The nodes along the center-line of the structure are fixed vertically to ensure symmetry in the response. This boundary condition was found necessary through unsuccessful simulations, which resulted in the entire honeycomb structure sliding upwards or downwards along the front face of the impacting solid, as seen in Figure 3.2. This behavior can be attributed to a rounding of
calculations done by ABAQUS, showing an initial small displacement of a single node along the centerline of the structure, which compounds into much more significant displacements as the structure is crushed further.

**Figure 3.2: Crushing of honeycomb structure without symmetry boundary condition**

The simulation of half of the honeycomb structure with the use of a periodic boundary condition, while investigated, is impractical because it allows the free tips of the half-honeycomb cells along the centerline of the structure to pierce one another, causing fatal errors within the FEA.
The nodes of the impacting body are given a constant velocity for each simulation, starting instantaneously, for the entire time step. The specific velocities simulated are 1 m/s, 2 m/s, 10 m/s, 50 m/s, and 100 m/s. A general contact interaction is defined for all surfaces within the model, with frictionless tangential behavior and “Hard” contact normal behavior, allowing for separation after contact.

All finite element analysis jobs for this report are run using 6 cores and 90% of the available 16.0 GB RAM on a Dell Precision 7400, with dual Intel® Xeon® E5405 quad-core processors. Detailed specifications for the generation of the ABAQUS models used in this report are presented in Appendix A.

3.2. Response Collection

The response of the honeycomb meso-structures to the impact loads is measured both qualitatively, via detailed descriptions of the crushing mechanisms observed, and quantitatively, by recording the reaction forces at the proximal and distal ends of the structure throughout the duration of the simulation. This section discusses the steps taken to gather the data required for accurate and detailed descriptions of the honeycomb meso-structures’ response to dynamic loading conditions.

Figure 3.3 shows a schematic of the loading and support conditions defined for the honeycomb crushing problem. As previously stated, the honeycomb structure is supported in the center of the distal end in the y-direction, with support of the entire distal column of cells in the x-direction.
The reaction force in the loading direction (history data output RF1 in ABAQUS) is recorded for both the distal and proximal ends of the honeycomb structure for 200 evenly spaced time steps over the duration of each simulation, shown in Table 3.1.

Table 3.1: Total and incremental time steps for impact simulations

<table>
<thead>
<tr>
<th>Impact Velocity [m/s]</th>
<th>Total Time of Simulation [s]</th>
<th>Incremental Time Step [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.4</td>
<td>0.002</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>0.001</td>
</tr>
<tr>
<td>10</td>
<td>0.04</td>
<td>0.0002</td>
</tr>
<tr>
<td>50</td>
<td>0.008</td>
<td>0.00004</td>
</tr>
<tr>
<td>100</td>
<td>0.004</td>
<td>0.00002</td>
</tr>
</tbody>
</table>

The reaction force at all nodes along the outer wall of the distal cells is summed to generate the overall distal reaction force for the honeycomb structure. The proximal force is obtained in a similar fashion, but recorded at each node along the inner face of the impacting body rather than the outer faces of the proximal column of cells of the honeycomb structure. The reported proximal reaction force is the sum of these nodal
reaction forces. This force can be described as the force necessary for the impacting solid to maintain a constant velocity, regardless of the mass of the body.

A high level of subjectivity is found when attempting to define the stresses in the honeycomb structure, which is a common metric reported in outside research, as can be seen in Table 1.2. This can be attributed to a few main factors, including (1) the non-uniformity of the change in the effective cross-sectional area of the honeycomb structure and (2) the lack of a true cross-sectional area, as material is not present at all points in the cross section.

The stress $\sigma$ is defined as the total reaction force $R_l$ across a specific node region divided by the cross-sectional area $A$ [15], which is not a sufficiently defined metric in previous work. The cross-sectional area is defined as if the honeycomb meso-structure were a solid, homogeneous material, in which case the area is the outer height multiplied by the out-of-plane depth of the structure.

$$\sigma = \frac{R_l}{A}$$  \hspace{1cm} (3.1)

While the reaction force is a straightforward to record as a history data output within ABAQUS, the cross-sectional area continually changes in a non-uniform manner as the structure is crushed, making engineering stress a controversial measure. This requires the user to define the cross-sectional area manually for each recorded time step, in order to accurately define the stress as a true stress, rather than an engineering stress.

Since the cross-sectional area is an effective area of the outer geometry of the structure rather than the true area of the present material, the stress reported can only be
reported as an effective stress, rather than a true stress. The true reaction force at either end of the honeycomb structure is reported in this study to counter these limitations.

Despite these limitations, the reported stress in several published studies may still be a reliable measure, as it acts simply as a scaling factor for easy comparison with other structures. However, the studies described in this report avoid the confusion of the aforementioned issues by reporting the true reaction forces at the proximal and distal ends of the honeycomb meso-structures.

3.3. Validation of Simulation Method

The method described in this section, which is used for all studies within this report, is validated via comparison with similar reported in literature [15], in order to ensure reliable comparison data when simulating other geometries and impact scenarios.

The previous studies show that the velocity of the impact plays a large effect on the behavior of the honeycomb during crushing, showing increases in average proximal end stress with increases in impact velocity, while the average distal stress remains relatively constant. Figure 3.4 through Figure 3.7 compare the results obtained by Zou [15] to the results obtained during this study for the proximal and distal ends, respectively.

The cell height and length consist of two elements, with two elements in the out-of-plane depth. Mesh sensitivity analysis and previous studies [15] have shown that four elements per cell wall length are sufficient when compared to finer meshing of the honeycombs in terms of data accuracy. Initial research during these efforts, however, show that two elements per cell wall are not only adequate, but provide for less noise in
the simulation response, as seen in Figure 3.5 through Figure 3.7. The stress-strain curves obtained during this study for validation against the previous research show a high degree of similarity in both average value and trends, but shows significantly less noise due to the reduced number of elements in the model. While the previous study crushes the structures to 90% global strain to show the densification of the structures, the current simulations stop at 80% global strain, as this research does not attempt to quantitatively or qualitatively describe such properties of the honeycombs.
Figure 3.4: Published proximal end stress vs. strain for Type A honeycomb [15]

Figure 3.5: Current proximal end stress vs. strain for Type A honeycomb
Figure 3.6: Published distal end stress vs. strain for Type A honeycomb [15]

Figure 3.7: Current distal end stress vs. strain for Type A honeycomb
It can be seen in the plots that while the current results have less noise than do the previous studies, both the averages and trends observed are highly similar. While analytical models will not be presented in this report, the plots shown in Figure 3.4 and Figure 3.6 have been verified with direct comparison to one-dimensional shock theory [15]. The correlation between the current results and those obtained by published studies [15], verify that the current approach gives meaningful results. Therefore the method used to develop these models will be used in the remainder of this thesis.
CHAPTER 4 : EFFECTS OF HONEYCOMB CELLULAR GEOMETRY ON DYNAMIC IMPACT RESPONSE

This chapter presents experimental simulation studies that explore how the honeycomb cellular geometry provides a significant effect on its structure’s response to low, moderate, and impact velocities, using the parameters and methods as defined in Chapter 3. The sections within this chapter include descriptions of four test geometries considered for simulation, detailed descriptions of their crushing mechanisms, and finally numerical force-displacement curves of the simulations. These results conclude that the manipulation of a honeycomb structure’s cellular geometry, even while maintaining a constant effective modulus in the loading direction, show enough variation to allow for their forward design for specific application requirements.

4.1. Honeycomb Cellular Geometries

Four honeycomb structures are evaluated in this chapter, with honeycomb cells at angles, $\theta$ of -30° (Type D), -15° (Type C), 15° (Type B), and 30° (Type A), ranging from highly auxetic (negative Poisson’s ratio) hexagonal cells to a conventional hexagon, as seen in Figure 4.1. For all geometries except for $\theta= -30^\circ$, the cell height $h$ and length $l$ are designed to be 4mm. For $\theta= -30^\circ$, $h$ is designed to be $2l$ (8mm) to prevent the overlap of the cell walls and vertices.
The effective modulus in the loading ($x_1$) direction is kept constant across the four cellular geometries by altering the thickness in equation (4.1) in response to change in cell angle [2],

$$E_i^* = E \left( \frac{t}{l} \right)^3 \frac{\cos \theta}{(\alpha + \sin \theta) \sin^2 \theta}$$  \hspace{1cm} (4.1)

Where $E$ is the constituent material elastic modulus, $\theta$ is the cell angle, $t$ is the cell wall thickness, $h$ is the cell height. The variable $\alpha$ is the cell aspect ratio ($\alpha=h/l$).

The wall thickness, $t$, of the conventional hexagonal cells (Type A in Figure 4.1) is first chosen to be 0.346 mm to maintain a relative density of 0.1 [15], which provided an effective modulus of 103MPa. The wall thicknesses of the Type D, Type C, and Type...
B configurations are calculated to be 0.346 mm, 0.17 mm, and 0.203 mm, respectively. It should be noted that the relative densities of the four models, as well as the other effective static material properties, cannot be kept constant in conjunction with a constant effective modulus.

The dimensions of each unit cell and the corresponding effective properties of the hexagonal honeycombs are shown in Table 1.

Table 4.1: In-Plane Effective Properties of the Designed Honeycombs

<table>
<thead>
<tr>
<th></th>
<th>Type A $\theta=30^\circ$</th>
<th>Type B $\theta=15^\circ$</th>
<th>Type C $\theta=-15^\circ$</th>
<th>Type D $\theta=-30^\circ$</th>
</tr>
</thead>
<tbody>
<tr>
<td>h (mm)</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>l (mm)</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>t (mm)</td>
<td>0.346</td>
<td>0.2029</td>
<td>0.17</td>
<td>0.346</td>
</tr>
<tr>
<td>$E_1^*$ (MPa)</td>
<td>103.0</td>
<td>103.0</td>
<td>102.9</td>
<td>103.0</td>
</tr>
<tr>
<td>$E_2^*$ (MPa)</td>
<td>103.0</td>
<td>12.6</td>
<td>4.3</td>
<td>103.0</td>
</tr>
<tr>
<td>$G_{12}^*$ (MPa)</td>
<td>25.7</td>
<td>3.9</td>
<td>1.4</td>
<td>3.9</td>
</tr>
<tr>
<td>$\rho^*$ (kg/m$^3$)</td>
<td>269.7</td>
<td>169.0</td>
<td>240.4</td>
<td>359.6</td>
</tr>
<tr>
<td>$\nu_{12}^*$</td>
<td>1</td>
<td>2.8637</td>
<td>-4.8637</td>
<td>-1</td>
</tr>
<tr>
<td>$\nu_{21}^*$</td>
<td>1</td>
<td>0.3492</td>
<td>-0.2056</td>
<td>-1</td>
</tr>
<tr>
<td>Rows</td>
<td>21</td>
<td>25</td>
<td>43</td>
<td>21</td>
</tr>
<tr>
<td>Columns</td>
<td>74</td>
<td>66</td>
<td>66</td>
<td>74</td>
</tr>
</tbody>
</table>

$E_1^*$, $E_2^*$, and $G_{12}^*$ are the effective in-plane moduli in the $x_1$ and $x_2$ directions and in-plane effective shear modulus, respectively. $\rho^*$ is the effective density of a honeycomb. $\nu_{12}^*$ and $\nu_{21}^*$ are the effective in-plane Poisson’s ratios of the honeycomb configuration. All effective properties are calculated for the honeycomb as if it were a homogeneous material with the given outer boundary conditions. The definitions for each of the effective static mechanical properties are [2]:

28
The outer dimensions obtained by the 74-cell horizontal x 21-cell vertical honeycomb structure of the conventional hexagonal cells are 512 mm x 128 mm. The necessary number of cells in the remaining three geometries varies to accommodate this constraint. While the number of cells for the Type D honeycomb does not change, a network of 66 x 43 of the Type C honeycomb are needed and 66 x 25 cells of the Type B cells is necessary for a consistent outer geometry.

4.2. Results

As previously stated, each of the four models is simulated with five different impact velocities for a total of 20 simulations. Each simulation captures a snapshot of the honeycomb response at overall strain in steps of 4% to a final 80% compression, at which the structures densify to a solid structure of the host material, which in this case is aluminum.
4.2.1. Crushing Mechanisms of Honeycomb Structures

Wave patterns within the honeycomb structures are observed and analyzed for each impact velocity. It is found that visual representation, along with descriptions of the crushing patterns of the meso-structures, are adequate in describing the crushing mechanisms of the four geometries. As defined by Zou et al [15], the low wave-trapping speed for honeycombs shows a wave forming at the distal end of the structure when first subjected to an impacting load. After a higher “critical wave” speed, the cells at the proximal end will fully crush without distributing the load to the distal cells. These behaviors are described in detail in the following three subsections. The 1 m/s impact simulations illustrate the honeycomb response below their wave trapping speed, while the 100 m/s simulations show the response after the critical wave speed. The 10 m/s simulations display the responses between these two speeds.

Each of the four cellular geometries is simulated at five crushing velocities to encompass the different crushing mechanisms discussed. This section displays the observed responses of the honeycomb structures at (1) 1 m/s impact velocity (2) 10 m/s and (3) 100 m/s impact velocity, illustrating the crushing mechanism differences between the four geometries. Figures of all crushing responses with descriptions of the crushing mechanisms are included in Appendix A.1.

4.2.1.1. Low Velocity Crushing Response – 1 m/s

The 1 m/s impact provides a quasi-static loading case that provides consistent proximal-end-only crushing response across all four honeycomb structures. The cells in the positive angle (Type A and B) structures collapse in such a way to form an “X”-
shaped crushing region at the proximal end of the structure, seen in Figure 4.2. These “X” regions collapse further into “V” bands as the structure is crushed further. The Type A structure forms a second “X” band, which grows in width by the sequential collapse of adjacent cells. The “V” band in the Type B structure, while not forming the secondary crushing region, follows a similar behavior in that the “V” band grows in width via the collapse of adjacent cells, with a less ordered fully collapsed region near the centerline of the structure.

![Figure 4.2: Type A (left) and Type B (right) response to 1 m/s impact](image)

The impact response of the auxetic Type C and D structures follow similar trends to that of Type A and B, but collapse inwards as the structure is crushed further as seen in
This behavior explained by the negative effective Poisson’s Ratio of the Type C and Type D structures. While the structures experience the bulk of the deformation at their proximal ends, it is noticed that these two structures do experience noticeable deformation at the distal end near the end of the simulation. This can be attributed to the lack of fully collapsed cells, unlike the responses of the Type A and C honeycomb structures. A secondary buckling region is noticed in the initial response of the Type C structure, which grows by collapsing the local columns of cells until the collapsed proximal columns meet it near 40% global compression.

Figure 4.3: Type C (left) and Type D (right) response to 1/ms impact

4.2.1.2. Moderate Velocity Crushing Response – 10 m/s

All four honeycomb structures show both proximal and distal end crushing when subjected to a moderate velocity impact of 10 m/s. The positive angle Type A and Type
B structures experience “V”-shaped crushing regions at both ends, which cascade into adjacent columns of cells until the two regions meet near 80% global compression, as seen in Figure 4.4. The Type B structure shows significant outwards collapse of the outermost columns of cells at both the proximal and distal ends, much greater than the collapse observed in the Type A structure.

![Figure 4.4: Type A (left) and Type B (right) response to 10 m/s impact](image)

The Type C structure, while at first exhibiting its negative Poisson’s ratio behavior by collapsing inwards at its proximal end, soon collapses back outwards, forming stacked “I”-shaped crushed columns of cells that grow towards the distal end. The distal end of the Type C structure initially responds with an inwards buckling region,
which soon forms more “I”-shaped collapsed columns of cells. The two regions meet near 60% global compression. The Type D structure follows a similar behavior, however it continues to collapse inwards throughout the simulation due to its more negative Poisson’s ratio. The deformed columns of cells only partially collapse until the structure is 50% crushed, afterwards the columns densify into highly compacted regions of fully crushed cells. These two responses can be seen in Figure 4.5.

Figure 4.5: Type C (left) and Type D (right) response to 10 m/s impact

4.2.1.3. High Velocity Crushing Response – 100 m/s

The responses of the four structures to high velocity (100 m/s) impact are highly disparate from their low and moderate velocity impact responses, which is in agreement with observations in published research [8,9,15]. The structures experience almost all deformation at their proximal end with the formation of “I”-shaped bands of fully collapsed cells. As seen in Figure 4.6, the standard Type A and B structures show a
significant outward collapse of the proximal columns of cells, with the only deformation of the distal cells occurring near 60% global compression.

Figure 4.6: Type A (left) and Type B (right) response to 100 m/s impact

The auxetic structures respond similar compared to the standard geometries, but again collapsing inwards due to their negative Poisson’s ratio. As seen in Figure 4.7, the Type C structure absorbs the bulk of the impact at its proximal end by fully collapsing columns of cells into a seemingly chaotic densified region. A buckling region forms near 40% global compression, which becomes more prominent as it collapses the full local column of cells as the structure is crushed further. The Type D structure, while still experiencing the majority of deformation at its proximal end, is more efficient in fully
collapsing its cells, which transfers a smaller ratio of the stress to the distal end of the structure.

Several generalizations can be made of the visual crushing responses of the four honeycomb structures across range of the simulated impact velocities. It is observed that the Type A and Type D structures are more efficient in the full collapse of their cells while preventing the collision of the vertices of cell walls. This is illustrated in Table 4.2, which compares a typical cellular collapse of all four cell types at both low and high velocities. The low-velocity cellular collapse of the Type A honeycomb shows a nodal shift to generate an “S” shape, with noticeable linear strain of the angled cell walls and negligible deformation of the vertical walls. The high velocity impact causes the Type A cell to fully collapse into a vertically slender configuration. The Type B and C cells

Figure 4.7: Type C (left) and Type D (right) response to 100 m/s impact
respond to the low velocity impact in a similar fashion to that of Type A by exhibiting a shift of the top and bottom nodes, but show a higher degree of buckling in the vertical direction when impacted at 100 m/s, due to their lower $E_2^*$ values. The Type D structure exhibits an almost identical reaction to both impacts as the Type A, but shows an inwards collapse of its cells due to its negative Poisson’s ratio.

<table>
<thead>
<tr>
<th>Type</th>
<th>Before</th>
<th>1 m/s</th>
<th>100 m/s</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>After</td>
<td>After</td>
</tr>
<tr>
<td>Type A</td>
<td><img src="image1" alt="Before" /></td>
<td><img src="image2" alt="1 m/s" /></td>
<td><img src="image3" alt="100 m/s" /></td>
</tr>
<tr>
<td>Type B</td>
<td><img src="image4" alt="Before" /></td>
<td><img src="image5" alt="1 m/s" /></td>
<td><img src="image6" alt="100 m/s" /></td>
</tr>
<tr>
<td>Type C</td>
<td><img src="image7" alt="Before" /></td>
<td><img src="image8" alt="1 m/s" /></td>
<td><img src="image9" alt="100 m/s" /></td>
</tr>
<tr>
<td>Type D</td>
<td><img src="image10" alt="Before" /></td>
<td><img src="image11" alt="1 m/s" /></td>
<td><img src="image12" alt="100 m/s" /></td>
</tr>
</tbody>
</table>

The low-speed impact cellular responses do not sufficiently illustrate the overall crushing mechanism of the structure, as all four cellular geometries exhibit a seemingly ordered crushing behavior. The differing factor between the visually “ordered” Type A
and D honeycombs and less ordered Type B and C honeycombs is how typical the displayed cellular crushing mechanism is within the entire structure. The displayed collapsed cells for the Type A and Type D structures are much more common than those shown for the Type B and Type C honeycombs.

It is clearly evident from the collapse of the structures on a cellular level that the Type A and D honeycombs collapse to the highest density when subjected to the high velocity impact, while the Type B and C honeycombs exhibit a high degree of randomness, causing a more chaotic crushing mechanism of their structures.

While the visual representations of the impact simulations discussed in this chapter are successful in demonstrating the disparities in the responses between the four honeycomb geometries, a quantifiable measure for direct comparison is necessary for means of direct comparison across the impact velocities and cellular geometries.

4.2.2. Quantified Effects of Impact Velocity on Honeycomb Response

One observation on the data recorded is that the reaction forces are dependent on the velocity of the impacting body, which is in agreement with published research [15]. It is found that while the distal reaction force remains similar across all the simulations for each geometry, the magnitude of the proximal reaction force is directly related to the impact velocity, as seen in Figure 4.8. These plots show the proximal and distal reaction forces for the Type A honeycomb structure at varying impact velocities. This shows that the honeycomb structures transfers a similar magnitude of the impacting force to their distal end, regardless of the impact velocity. This behavior can be explained by the fact that at that higher impact velocities, the structures experience a higher degree of plastic
deformation that while resisting the motion of the impact, does not allow the proximal forces to be transmitted through to the distal end of the structure.

Figure 4.8: Proximal (top) and distal (bottom) reaction forces for Type A honeycomb structure

4.2.3. Cellular Geometry Effects on Honeycomb Impact Response

A second observation made is that the cellular geometry has a large effect on the reaction forces of the structure, aside from solely visual crushing mechanism differences.
The differences are evident across all crushing velocities, as shown in Figure 4.9, which displays the low velocity (1 m/s) reaction forces at the proximal and distal ends, and Figure 4.10, displaying the high velocity (100 m/s) reaction forces. As can be seen in these plots, there is little disparity between the reaction forces of the different structures at both the proximal and distal ends of each structure, showing that the low impact velocity scenario is highly similar to quasi-static loading conditions. This effect shows a greater influence of the effective static properties of the four honeycomb structures on their response.
Figure 4.9: Honeycomb reaction forces to 1 m/s impact - proximal (top) and distal (bottom)

It can be seen when comparing the reaction forces in Figure 4.9 that there is little disparity between the proximal and distal reaction forces of the four meso-structures, meaning that there is negligible effective plastic strain energy, as defined in equation (4.7), being dissipated by the honeycomb structures at low impact velocities.

The Type A and Type D honeycomb structures exhibit similar reaction forces to one another because the two have equal effective moduli in the vertical direction, or $E_2^*$. 
The Type B and C honeycombs show lesser reaction forces at both ends because they have a significantly lower $E_2^*$ than the Type A and B structures, as seen in Table 4.1. This effect is lessened as the impact velocity increases, showing greater dependence of the honeycomb response on their nonlinear dynamic properties that have yet to be effectively quantified.

As seen in Figure 4.10, there is high disparity between the proximal and distal reaction forces recorded during the simulations of the high velocity impacts. This shows that all the honeycomb configurations gain effectiveness in the dissipation of the impact energy through greater plastic energy absorption as the impact velocity increases.
Figure 4.10: Honeycomb reaction forces to 100 m/s impact - proximal (top) and distal (bottom)

4.3 Honeycomb Effectiveness

The high disparity between the reaction forces at the two ends of the structure identifies the need of a new metric, which quantifies the effectiveness of the honeycomb structure in absorbing the impact rather than transmitting the impact energy through the structure. It is desirable in many applications for the cellular structure to have a large reaction force at the proximal end of the structure while maintaining a low reaction force at the distal end. For example, if placed in the front bumper of an automobile, a highly
effective apparatus would react by requiring a high force on the proximal end to maintain
the constant velocity of the impacting solid member, while leaving the distal end, or end
closer to the driver, free from large-scale reaction forces. This metric can be defined as
the effective plastic strain energy dissipated by the structure \( W \),

\[
W = \sum_{i=1}^{n} \left( R_p(t_i) - R_d(t_i) \right) \Delta u
\]  

(4.7)

\( R_p(t_i) \) and \( R_d(t_i) \) represent the total reaction forces at the proximal and distal ends
for each time increment, respectively, and \( \Delta u \) represents the incremental displacement.
This displacement remains a constant value for each simulation, defined as

\[
\Delta u = \frac{\varepsilon_{max} \ast L}{n}
\]  

(4.8)

with \( \varepsilon_{max} \) representing the maximum global compression, \( L \) pertaining to the length of the
honeycomb structure in the loading direction, and \( n \) representing the number of time steps
for the simulation. For this study, a constant 200 time steps are recorded over a span of 0-
80% global compression.

The honeycombs are compared in terms of specific plastic energy absorption, defined as

\[
w = \frac{W}{m_h}
\]  

(4.9)

with \( m_h \) representing the mass of the entire honeycomb structure. The specific plastic
energy absorption allows for non-biased comparison of the four simulated geometries, as
seen in Figure 4.11. While an increase is noticed in specific energy absorption across all four structures, there are slight differences in the trends of each geometry. The histogram shows that at low impact velocities, the Type A, Type C, and Type D honeycombs show a negative effective plastic energy absorbed by the structure, meaning that it transmits more force to the distal end of the structure than resists on the proximal end. The Type B honeycomb is the only structure that does not exhibit this behavior at any impact velocity. The trends show that the auxetic Type C and Type D structures perform slightly better in velocities of 50 m/s and greater, with the Type D structure performing the greatest. The results from additional simulations performed are presented for the Type D structure to better illustrate the trend between specific plastic energy absorption and impact velocity.
The results presented in this chapter present that effects of the cellular geometry on the response of a honeycomb structure to dynamic impact are clearly evident, supported by the qualitative descriptions of the crushing mechanisms of the structures in section 4.2.1 and the quantitative representations of the reaction forces and plastic energy absorbed by the structures in section 4.2.3. However, there is still a lack of sufficient data for generalization of the response in terms of specific geometric parameters. A new method is found to be necessary for the acquisition of such amounts of data with greater efficiency and less need for user interaction and input. The developed method in Chapter 5 presents an algorithm for the sequential running of multiple simulations with any number of specified parameters from a Design of Experiments (DOE) table. The method is used for creating, analyzing, and post-processing ABAQUS simulations and the creation of a Response Surface Model that displays a desired output as a function of the
pre-specified inputs from the user. A much larger design space of the parametric values for the honeycomb configurations can be explored in this manner.
CHAPTER 5: DESIGN OF EXPERIMENTS METHOD FOR THE TARGETING OF HONEYCOMB RESPONSE TO DYNAMIC LOADING CONDITIONS

The cellular geometry of the honeycomb meso-structure is varied in this study to determine energy absorbed by a honeycomb structure when subjected to a high velocity impact from an effectively rigid solid, as previously represented in Figure 3.1. Figure 5.1 shows the geometry notation used for this section.

![Honeycomb unit cell and nomenclature](image)

**Figure 5.1: Honeycomb unit cell and nomenclature**

5.1. Objective Function

Various design objectives are used in previous research of the dynamic properties of honeycomb structures. Extensive work has been done in designing honeycombs with the objective of generating specific shear strengths using cell length and thickness as the design variables [32,1]. Because of the lightweight properties of honeycomb meso-structures, another design objective has been to minimize weight while still retaining targeted stiffness values [33].
Several studies have also been done to optimize the impact energy absorption, but with either out of plane crushing [34], honeycomb filled structures [19], or honeycomb sandwich panels [35]. In these instances, parameters such as number of cells, cell wall thickness, yield stress, or shape of filled structure are used as the design variables.

The total energy absorption is the objective function for this study, which is defined in equation (4.7). Both the angled wall thickness \( t_2 \) and the cell angle \( \theta \) are free variables for the design of experiments. For manufacturability, the thickness is given a minimum of 0.1 mm and a maximum of 0.5 mm, while the cell angle is bounded by angles of -15° and 60°. The mass of the honeycomb structure is kept constant in order to allow equal comparison across all geometries tested. This is done by maintaining a constant cross-sectional area \( A \) of each cell, as constructed by ABAQUS. As previously mentioned in Chapter 3, S4R shell elements are defined across the model, which apply the cell wall thickness by performing a mid-plane extrusion on either side equal to the user-defined section thickness. In this particular case, the cell wall thicknesses \( t_1 \) and \( t_2 \) define the section thicknesses, as seen in Figure 5.1. These values are then determined to define the cross sectional area,

\[
A_{cell} = h \left( t_1 + 2t_2 \right) \quad (5.1)
\]

Which when multiplied by the out-of-plane depth \( d \) of the shell is equal to the cell volume \( V_{cell} \):

\[
V_{cell} = A_{cell} d = h d \left( t_1 + 2t_2 \right) \quad (5.2)
\]
The cellular volume is then multiplied by the number of columns $n_{cols}$ and rows $n_{rows}$ in the structure to obtain the volume of the entire structure, and finally multiplied by the density of the constituent material $\rho$ (in this case aluminum) to obtain its mass, $M_{structure}$.

$$M_{structure} = \rho V_{structure} = \rho n_{cols} n_{rows} V_{cell}$$

(5.3)

The vertical wall thickness is defined as a function of the angled wall thickness in order to maintain the constant mass of the structure.

$$t_1 = \frac{V_{cell}}{h d} - 2t_2$$

(5.4)

The algorithm used for this chapter is shown in Figure 5.2. The process begins with the use of MATLAB to define a DOE with the input parameters of interest, which are then used to modify the dependent parameters within the model. A python script template (seen in Appendix B) is edited to match the parameters for each run in the DOE and used to run an ABAQUS/Explicit simulation for each cellular configuration. The results from the ABAQUS simulation are interpreted to the response desired and stored. This process is repeated for each design in the sequence. The input parameters are then analyzed for their significance on the response. The trends in the design variables versus the response variable $W$ are displayed using a response surface model (RSM) and can then be used for the targeting of specific response values.
Figure 5.2: Algorithm for response surface creation

MATLAB R2009b serves as a black-box function which processes the inputs and generated the outputs from which Isight v. 4.5 can generate the response surface for the function of the response $W$ versus the free variables $t_2$ and $\theta$. The output linked to the
analysis is $W$, which is calculated using equation (4.7). MATLAB then generates a random design sequence of paired $\theta$ and $t_2$ values. A random distribution of the design variables across the design space is desired to create an accurate depiction of the entire design space, which is not well known for this objective. MATLAB then calculates the dependent design variables using the values for $\theta$ and $t_2$ and generates the python script to create and analyze the ABAQUS model, and finally export the results back to MATLAB for analysis.

5.2. Design of Experiment Results

MATLAB and ABAQUS completed the analyses for each configuration specified in the DOE table, using the results to calculate a single final response of each structure as the plastic energy absorption $W$. Table 5.1 displays the raw data obtained from the sequentially run simulations.

<table>
<thead>
<tr>
<th>Run</th>
<th>$t_2$ [mm]</th>
<th>$\theta$ [°]</th>
<th>$W$ [J]</th>
<th>Run</th>
<th>$t_2$ [mm]</th>
<th>$\theta$ [°]</th>
<th>$W$ [J]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.358</td>
<td>57.232</td>
<td>742.3</td>
<td>26</td>
<td>0.492</td>
<td>52.779</td>
<td>760.1</td>
</tr>
<tr>
<td>2</td>
<td>0.251</td>
<td>26.010</td>
<td>678.4</td>
<td>27</td>
<td>0.276</td>
<td>51.819</td>
<td>593.9</td>
</tr>
<tr>
<td>3</td>
<td>0.425</td>
<td>24.085</td>
<td>850.8</td>
<td>28</td>
<td>0.144</td>
<td>10.062</td>
<td>607.4</td>
</tr>
<tr>
<td>4</td>
<td>0.313</td>
<td>2.370</td>
<td>877.8</td>
<td>29</td>
<td>0.203</td>
<td>37.406</td>
<td>628.6</td>
</tr>
<tr>
<td>5</td>
<td>0.240</td>
<td>21.667</td>
<td>691.0</td>
<td>30</td>
<td>0.263</td>
<td>-0.164</td>
<td>857.6</td>
</tr>
<tr>
<td>6</td>
<td>0.476</td>
<td>31.805</td>
<td>842.1</td>
<td>31</td>
<td>0.338</td>
<td>-12.709</td>
<td>1008.5</td>
</tr>
<tr>
<td>7</td>
<td>0.450</td>
<td>35.935</td>
<td>772.8</td>
<td>32</td>
<td>0.205</td>
<td>40.806</td>
<td>668.9</td>
</tr>
<tr>
<td>8</td>
<td>0.320</td>
<td>14.664</td>
<td>977.2</td>
<td>33</td>
<td>0.341</td>
<td>22.502</td>
<td>807.7</td>
</tr>
<tr>
<td>9</td>
<td>0.349</td>
<td>12.558</td>
<td>1019.7</td>
<td>34</td>
<td>0.384</td>
<td>20.994</td>
<td>903.4</td>
</tr>
<tr>
<td>10</td>
<td>0.335</td>
<td>59.099</td>
<td>667.8</td>
<td>35</td>
<td>0.189</td>
<td>52.854</td>
<td>639.1</td>
</tr>
<tr>
<td>11</td>
<td>0.183</td>
<td>-12.170</td>
<td>713.6</td>
<td>36</td>
<td>0.147</td>
<td>30.740</td>
<td>574.2</td>
</tr>
<tr>
<td>12</td>
<td>0.220</td>
<td>51.388</td>
<td>526.5</td>
<td>37</td>
<td>0.219</td>
<td>31.325</td>
<td>611.8</td>
</tr>
<tr>
<td>13</td>
<td>0.288</td>
<td>53.497</td>
<td>559.1</td>
<td>38</td>
<td>0.228</td>
<td>49.458</td>
<td>642.4</td>
</tr>
<tr>
<td>14</td>
<td>0.192</td>
<td>44.714</td>
<td>603.4</td>
<td>39</td>
<td>0.270</td>
<td>45.412</td>
<td>521.5</td>
</tr>
</tbody>
</table>
5.3. Sensitivity Analysis of Design Variables

This section describes the results of the sensitivity study conducted on the two independent parameters, $t_2$ and $\theta$, on the response variable $W$. The independent parameters and interaction effects of these parameters on the response variable are studied using a statistical technique with commercially available software Minitab. The goal of this study is to identify the possibility of eliminating any one of these variables without significantly affecting the response before further analysis and also to reduce computational time in future studies where more design variables can be considered.

The calculation of the effects of the two independent parameters is based on the responses obtained from the DOE. The DOE data is input to Minitab for determining the effects using a model which assumes data to be normally distributed. Hence, a normality check is conducted using Kolmogorov-Smirnov test [36]. The indicator to determine whether or not data is normally distributed is based on the determined ‘p-value’ and comparing it with the confidence interval $\alpha$. The acceptance criterion for a set of data to be normally distributed is $p$-value $> \alpha$. At 95% confidence interval ($\alpha = 0.05$), the computed $p$-value for $t_2$ is 0.122 and hence, it is determined that $t_2$ is normally distributed.
distributed, which is further verified by the normal probability plot in Figure 5.3. The solid blue diagonal line represents a perfect normally distributed data set, meaning that the closer a data set is to alignment with the line, the more normally distributed it is.

![Normal probability plot of $t_2$ from Kolomgorov-Smirnoff test](image)

**Figure 5.3: Normal probability plot of $t_2$ from Kolomogorov-Smirnoff test**

The p-value for $\theta$ is calculated to be greater than 0.150, which shows that its values in the DOE are also normally distributed, as further verified with its normal probability plot in Figure 5.4. Therefore, the distribution of the DOE data can be used to determine the effects of the independent parameters.
The effects of both the independent parameters ($t_2$ and $\theta$) and the interaction between the two on the response are presented in Figure 5.5 using a Pareto chart. The horizontal axis represents the effect, or ‘t’-statistic value, of variation of an independent parameter on the response while the vertical axis represents the different parameters and their possible interactions. The vertical red line at 2.01 is standard marginal error (SME) for 95% confidence interval. Any bar that extends beyond this line indicates that the corresponding parameter has significant effect on the response. Readers may also consult [37,38] for additional information on construction and interpretation of this chart.
This study shows that the angled wall thickness $t_2$ has the highest effect on the response $W$, followed by the cell angle $\theta$ and their interaction. All these parameters are deemed essential for further investigation because each shows an effect with significance greater than the confidence interval.

5.4. Response Surface Model

An RSM is created from the DOE data to approximate the response $W$ as a function of the design variables $t_2$ and $\theta$. An accurate RSM allows for the generalization of a response through the interpolation of the DOE data, rather than having the need to run time-consuming simulations for every configuration within the design space [39].

Radial basis functions (RBF) are chosen to create the RSM because of their ability to interpolate multivariate data well, like those generated from the DOE sequence [39]. RBF’s utilize a radially symmetric function derived from a Euclidian distance to
create a response surface approximation. They have been shown to produce accurate and robust models when limited sample sizes are used and perform well for highly nonlinear problems [40]. Previous work has been done in analyzing the impact energy absorption of out-of-plane honeycomb crushing [34] and vehicle crashworthiness [41] in which RBF’s were able to create accurate response approximations. Because of their overall accuracy and robustness with limited samples sizes, ability to handle multivariable data and their previous use in similar analyses, radial basis functions are deemed an appropriate choice for this analysis.

Isight v. 4.5 is used to generated the RBF model and as the optimization package to determine the maximum energy absorption within the previously mentioned design constraints. The visual representation of the RBF generated by Isight is seen in Figure 5.6. This surface plot shows maximum energy absorption near $\theta = 12.5^\circ$ and $t_2 = .35$ mm, which is also shown in the response from the 9th run in the DOE in Table 5.1. The plot shows the trends in the data, in that the plastic energy absorbed by the structure increases with decreasing cell angle and increasing angled wall thickness, and decreases with increasing cell angle and decreasing angled wall thickness.
The configuration providing lowest energy absorbed, which is the 39th run in the DOE, has $\theta = 45.4^\circ$ and $t_2 = 0.27$mm. These cellular configurations are shown in Figure 5.7, and will be referred to as Type X and Y honeycombs, respectively, from this point forward.

**Figure 5.7: Cellular geometry providing maximum (left – Type X) and minimum (right – Type Y) plastic energy absorption**
The two honeycomb structures exhibit similar crushing responses to the 100 m/s impact, both showing deformation at only the proximal end, as seen in Figure 5.8. The Type Y configuration, however, shows a uniform, complete cellular collapse of its proximal columns of cells in contrast to the Type X configuration.

![Figure 5.8: Type X (left) and Type Y (right) honeycomb response to 100 m/s impact](image)

This behavior explains the greater variation in the proximal and distal forces in the force-displacement curve of the Type X configuration in Figure 5.9. While the Type X curve shows a greater distal reaction force than the Type Y, it also shows a greater proximal reaction force throughout the simulation. Furthermore, while the mass of the two structures are equal, their lengths are not, meaning that the longer Type X structure
(0.578m) experiences a longer duration of impact than does the Type Y structure (0.416m). The volume of the two structures are equal, however, meaning that the same amount of material is crushed in both cases, which can conclude that the Type X structure is more efficient in absorbing the impact energy due to plastic deformation than the Type in terms of both per unit mass and per unit volume.
Figure 5.9: Type X and Type Y response to 100 m/s impact: Proximal (top) and Distal (bottom) reaction forces

5.5 Optimization of Honeycomb Cellular Geometry

One specific use of the method presented in this chapter is the use of the RBF for the optimization of the honeycomb cellular geometric parameters for maximum effective plastic strain energy absorbed by the honeycomb structure. This section includes the optimization of the angled wall thickness $t_2$ and cell angle $\theta$ for the maximum energy absorbed by the structure, using the RBF model seen in Figure 5.6.
The Multi-Island Genetic Algorithm (GA) is selected as the optimization algorithm based on its classification as an exploratory technique and compatibility with discontinuous and non-linear design spaces. While the design space is not predicted to be discontinuous, it is deemed advantageous to account for the possibility of being so. This GA utilizes competing sub-populations to be more effective than a broader search [42]. In it, the population of chromosomes is partitioned into sub-populations, which evolve independently while optimizing the same objective function. Periodically, certain chromosomes are replaced with better ones [42].

Isight is used to optimize the free variables of $t_2$ and $\theta$ to maximize the impact energy absorbed $W$ by the honeycomb structure using the aforementioned Multi-Island Genetic Algorithm on the RBF data. The GA is given 15 islands, providing for 1501 total design evaluations, finding the optimum solution on iteration 1412. These results are then verified by generating a new ABAQUS simulation with the parameters defined by Isight. The optimal parameters found by this algorithm are seen in Table 5.2.

<table>
<thead>
<tr>
<th>$t_2$ [mm]</th>
<th>$\theta$ [deg]</th>
<th>$W$ [J] (predicted)</th>
<th>$W$ [J] (actual)</th>
<th>Error [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.36</td>
<td>10.5</td>
<td>1012</td>
<td>953</td>
<td>5.83</td>
</tr>
</tbody>
</table>

It is found that the energy absorption of the “optimal” configuration as specified by the optimization routine is less than the maximum energy absorbed by the DOE configurations. However, the low error between the approximation and the validation simulation output, as well as the high similarity between the parameters of the optimal
configuration and the Type X honeycomb, validates the RBF and Multi-Island GA used for optimization.

5.6. Improvements in Computational Efficiency

The computational expense of each simulation may decrease the value of the methods described in this chapter, despite the lack of need of user input past the generation of the initial DOE necessary to complete a set of sequentially run simulations. While the algorithm used still generates, analyzes, and reports the results of each simulation with no delay between simulations, the time required to complete the 50 analyses in this chapter is on the order of days, leaving much room for improvement. This limitation generated the need for decreasing the computational expense for each individual simulation by reducing the number of cells within each generated FE model. This process is described in detail in Chapter 6.
While the studies described in the previous chapters provide meaningful results, the high computational expense of running the simulations necessary limits the quantity of data that can be recorded in a reasonable amount of time. As seen in Appendix A, a single simulation of the Type C geometry subjected to 1 m/s impact takes as long as 31 hours to complete while running on six cores and 90% of the available 16 GB of RAM available on the computer, which may make the use of FE for analyses similar to this impractical to the user, especially when attempting to run multiple simulations in sequence. From this limitation arose the potential opportunity to decrease the required time to complete the simulations by reducing the size of the models analyzed while maintaining a high degree of accuracy and negating the boundary effects of the outer rows of cells. It was initially hypothesized that the simulations of the impact of honeycomb meso-structures of reduced numbers of cells, while showing a lower magnitude of the reaction forces at their proximal and distal ends, will show similar trends and crushing mechanisms in their response.

This chapter discusses the efforts made to reduce the required number of rows of cells necessary to produce accurate and repeatable results for forward research, design, and analysis of honeycomb cellular meso-structures. The results find that a reduction in the number of rows allow for highly accurate results, in that the reaction forces at both the distal and proximal end of the structure can be linearly scaled to match those of structures with a higher number of rows within an acceptable degree of error.
6.1. **Methodology**

The method used to decrease the computational resources necessary for the impact simulations is similar to that used by Orr [43]. Her method showed that the size of a FE model of 3D Tweel rolling on lunar sand can be minimized while maintaining negligible boundary effects on the results. Her method lacks the need of a complex design of experiments, but instead approaches the problem in a more logical approach through the controlled increase or decrease of key dimensions and adjusting based on the simulation results. As previously stated, the method and results described in this chapter discuss a similar approach, in which the number of rows of cells is controlled in such a way that allows for informed judgments on the necessary dimensions that keep both computational expense and boundary effects to a minimum, without the use of a complex DOE.

The simulations described in Chapter 5 analyze models with a consistent 74 columns and 21 rows of cells, dimensions that are defined through the re-creation of the research efforts of Zou et al [15]. While the sequential simulation method is both robust and efficient in gathering large amounts of data, a higher degree of efficiency can be achieved through the reduction of the rows of cells, which will reduce the computational time and resources necessary for FEA.

The process includes the expansion of the honeycomb structure to a network of 74 columns and 41 rows of cells for the validation of the initial 74 column by 21 row structures. The structure is then decreased to 74 columns by 9 rows, $74 \times 5$, and finally 74 cells in a single row, as illustrated in Figure 6.1, which displays the Type A honeycomb
structures for each specification. Similar geometries are generated for the Type D cellular geometry for further verification.

Figure 6.1: Type A honeycomb structures used for sizing analysis

Figure 6.2 shows the different size Type D structures simulated for comparison.
All structures for both Type A and Type D geometries are subjected to low, moderate, and high velocity impacts of 1 m/s, 10 m/s, and 100 m/s, respectively, which are chosen because of these three encompass the full spectrum of the three pre-defined crushing mechanisms as discussed in Chapter 4.

6.2 Results

Force-displacement curves similar to those seen in Chapter 4 are generated for each impact velocity to compare the results across the five sizes of the honeycomb
structures. Figure 6.3 displays the reaction forces at the proximal and distal ends of the different size Type A honeycomb structure subjected to 100 m/s impact.

As predicted and can be seen in the figure, the structures with fewer rows of cells provide for less reaction force than do the structures with more rows, while they exhibit similar trends throughout the simulations. The force-displacement curves of the structures
with fewer numbers of cells are then linearly scaled upwards for direct comparison with the 74 column by 41 row structure;

\[ F_x(t_i) = \frac{m}{j} F_a(t_i) \]  

(6.1)

with \( F_a(t_i) \) representing the actual reaction force at a given time step (provided by ABAQUS), \( j \) the number of rows in the current structure, and \( F_a(t_i) \) the scaled reaction force at the time step, and \( m = 41 \), or the number of rows in the largest structure. Each structure is evaluated, comparing its reaction forces to that of the 74 x 41 structure. The error of each scaled model is determined using a normalized Root-Mean-Squared value, which has been used for analysis in similar studies [44];

\[ e = \sqrt{\frac{1}{n} \sum_{t=1}^{n} (F_x(t_i) - F_m(t_i))^2} \]  

(6.2)

with \( F_m(t_i) \) denoting the reaction force of the 74 column, 41 row structure at the current time step, \( n \) the number of time steps in the simulation, and \( F_{\text{max}} \) and \( F_{\text{min}} \) being the maximum and minimum forces recorded across the range of the reaction forces:

\[ F_{\text{max}} = \max \left( F_m(t) \right) \]
\[ F_{\text{min}} = \min \left( F_m(t) \right) \]  

(6.3)

Figure 6.4 displays the error for the Type A and Type D structures at their proximal and distal ends for each impact velocity, when compared to the response data from the 41 row by 74 column structures.
Figure 6.4: Normalized RMS error of reduced-size honeycomb structures compared to 41 row x 74 column honeycomb structure reaction force data

As can be seen in the error plots, the error of each simulation decreases with an increase in the number of rows of cells in the structure, because of a lowered degree of the boundary effects on the response of the structure. The structures with only one row of cells show unacceptable error levels because of the lack of any cell-to-cell interaction outside of the single row. The increased time to complete a simulation with greater numbers of cells, however, may justify an increase in error in the response, depending on the goal of the analysis at hand. As can be seen in Figure 6.5, the time required to run the
simulations increases almost linearly with both the number of rows and the impact velocity.

![Figure 6.5: Type A (left) and Type D (right) computational time required for simulations](image)

The results presented in Figure 6.4 and Figure 6.5 allow an analyst to determine which size model is appropriate for their study, using the trade-off between computational time required and RMS error. For example, exploratory studies of several simulations might find that nine-row structures are adequate, as they still exhibit similar trends to the 41-row structures, as seen in Figure 6.6. The red dotted line in these plots show the response of the nine-row structure scaled using equation (6.3). Higher accuracy on specific simulations can then be obtained using structures with more cells.
Figure 6.6: Scaled proximal (top) and distal (bottom) reaction forces for Type A honeycomb subject to 100 m/s impact

The results presented in this chapter provide an answer to RQ3 by showing that the recorded reaction forces at the proximal and distal ends of a honeycomb meso-structure can be linearly scaled to match those of greater numbers of cells, with an acceptable degree of error. The time saved by analyzing models of fewer numbers of rows of cells can allow for a more rapid collection of trends between various geometric parameters and response collected.
CHAPTER 7 : CONCLUSIONS AND FUTURE WORK

The work described in this thesis is motivated by the lack of a sufficiently reported, generalized method for the finite element analysis of the response of cellular meso-structures to dynamic loading conditions, as identified in Chapter 1. The studies performed were in an attempt to answer three main research questions that are aimed at filling this gap in published research:

RQ1. Does cellular geometry have a significant effect on the response of a honeycomb meso-structure to dynamic impact loading?

RQ2. How can specific responses of honeycomb cellular meso-structures be targeted using the modification of cellular geometry?

RQ3. How small of a model of honeycomb meso-structures can be linearly translated to large-scale models?

Chapter 3 discusses the modeling parameters used to simulate the response of honeycomb cellular structures to impact loads. These parameters are valid not only for the three following chapters, but also for future studies of cellular structures, as they are shown to provide both accurate and reliable results.

Chapter 4 describes the effects of the cell angle on both the crushing mechanisms and reaction forces of the proximal and distal ends of a honeycomb structure. It shows that the energy absorbed by the structure, as defined in equations (4.7) through (4.9), varies with the cell angle, even while the effective modulus of the structures in the loading direction (defined in equation (4.6)) is kept constant. While the results of the
chapter are promising in showing that target values of plastic energy can be absorbed through the design of the structure’s cellular geometry, more data is needed for generalization. This data is attainable through the development and use of the sequential simulation method described in Chapter 5.

Chapter 5 describes a newly developed method to rapidly create and analyze the response of multiple honeycomb structures to impact, using parameters defined by a design of experiments (DOE) sequence. In this particular case, both the angled wall thickness $t_2$ and cell angle $\theta$ are controlled while defining a function for vertical wall thickness $t_1$ to maintain a constant mass of all structures, seen in equation (6.2). The algorithm in Figure 5.2 is used to run the simulations and process their results, which can be used for the generalization of the response of all structures across the domain, as seen in the RBF model in Figure 5.6. Despite the benefit of this method towards the final goal of the forward design of honeycomb cellular meso-structures for dynamic loading conditions, the high computational expense each individual simulation motivates the research of decreasing the necessary size of the structure of which the results can linearly scaled to larger structures for specific applications.

The studies described in Chapter 6 answer RQ3 by showing that the response of smaller-scale honeycomb meso-structures can be linearly scaled to match the response of larger models within a reasonable degree of error. The results show that honeycomb structures of five and nine rows of cells still exhibit similar trends and magnitudes of the reaction forces at the proximal and distal ends of the structure when scaled to match the reaction forces seen in structures of 41 rows of cells. This provides great benefit to an
analyst of the dynamic properties of honeycomb meso-structures, as the smaller models require only a fraction of the computational time to simulate than do the 41-row structures, which allows for greater numbers of simulations to be performed. This in turn permits studies more exploratory in nature, using small-scale models to find areas of interest that can then be analyzed with a higher degree of accuracy with the simulation of larger honeycomb meso-structures.

The largest contribution made by this work to the field of the analysis of the dynamic properties of honeycomb meso-structures is the development of a standard set of modeling parameters that can be used for their further study and design in specific engineering applications. The use of the methods as described in this report will allow analysts to directly compare several studies together, making generalization of the dynamic response of honeycomb meso-structures attainable.

Despite the validation of this method with published work and consistency among the results, it may be further certified with its agreements with recorded data from physical experimentation. Section 7.1 describes two proposed future areas of work to perform such tests.

7.1 Experimental Validation

Extensive data from physical experimentation is needed to support the claims made in this report about the behavior of honeycomb cellular meso-structures. While not discussed in these studies, two high potential concepts for experimental studies are identified.
7.1.1. High-Pressure Piston Acceleration

Steps towards the experimental validation of the simulations discussed in this report have been taken through a design project conducted by a class of senior level undergraduate mechanical engineering students [45]. The project consisted of the design of an apparatus that re-creates the boundary conditions and loads defined in the FE simulations in a physical manner to capture both qualitative and quantitative results that substantiate the findings discussed in this report.

The apparatus, the prototype seen in Figure 7.1, uses compressed air to accelerate a piston that pushes a sleigh to a specified velocity to impact a honeycomb structure that is fixed on the bottom of the fixture. Roller guides on either side of the apparatus are used to constrict motion in the non-loading directions guide the sleigh.
The concept behind this design shows high potential in the ability to re-create the impact environments simulated in this report. However, the design still needs significant development before it can be used for the collection of reliable and repeatable data. Identified areas for improvement include:

Figure 7.1: ME 401 experimental apparatus prototype

- Piston
- Pressurized air tank
- Impact sleigh
- Roller guides
- Honeycomb
1. The maximum impact velocity provided by the current iteration of this apparatus is 15 m/s, which is significantly less than the maximum impact velocities simulated in this report. Extremely high pressures are needed to accelerate the piston to the target velocities, showing a need for a more robust support structure for the apparatus.

2. There are no means of securing the honeycomb structure in the out-of-plane direction while it is being impacted, leaving a potential for the structure to either buckle in the out-of-plane direction or slip from under the impacting sleigh.

While the design still needs significant improvements and refinements, preliminary results show its potential to accurately capture the behavior of honeycomb cellular meso-structures to impact loads. The apparatus was initially tested with the crushing of aluminum beverage at varying impact velocities which show similar crushing mechanisms to honeycomb structures across the tested impact velocity range.

7.1.2. Hopkinson Bar Apparatus

A second, more well-known method for the acquisition of stress and strain data from dynamic impact experimentation is the use of a Hopkinson Bar apparatus. Seen in Figure 7.2 [46], an impacting bullet is projected towards a sample at the head of a Hopkinson bar, which measures the stress and strain at one end of the sample.
Using high-speed cameras and strain gauges, the response of a honeycomb meso-structure can be both quantitatively and qualitatively measured for verification of the numerical simulations as described in this report. While this apparatus has been shown in published studies to provide reliable and accurate data, the cost of assembling such a device may be limiting.

7.2. Future Applications

The following section briefly describes concepts that have been perceived as potential areas for further development using the methods for the simulation of the honeycomb meso-structures subjected to impacting loads. These applications include, but are not limited to, the control of stress and deformation wave propagation throughout a structure, as well as the control of acoustic properties of honeycomb or other cellular materials through the modification of cellular geometric parameters.

7.2.1. Controlled Wave Propagation

One foreseen application of this research is to either control cellular geometric parameters or introduce imperfections, voids, or solid members to honeycomb meso-structures to control the propagation of stress waves through the specimen. Figure 7.3
shows deformation waves induced by stress waves in the Type D structure subjected to a 2 m/s impact. As seen in the figure, the deformation waves propagate throughout the entire structure.

![Deformation waves](image)

**Figure 7.3: Deformation waves in Type D structure subject to 2 m/s impact**

It is predicted that the control of these prominent waves can be attained via the modification of the cellular geometry or the addition of voids or inclusions. For example, removing the cells in the middle of the structure may allow for the manipulation of the shape and direction of the deformation waves as they propagate through the structure.

7.2.2. Acoustic Wave Control

Similar to the control of the stress waves as mentioned in the previous section, a second area for development with the use of honeycomb meso-structures is the control of
the propagation of sound waves through a structure through the design of cellular geometric parameters.

7.3. Application of Honeycomb Analysis Methods to Future Work

While the aforementioned topics have not been thoroughly researched, the methods used to simulate, analyze, and describe the dynamic crushing of honeycomb meso-structures as described in this report will be sufficient for future work into the control of stress and deformation waves within the structures. The modeling approach described in Chapter 3 will allow for the dynamic physical analyses of honeycomb meso-structures, regardless of the desired response. This is validated through the success of the method for modeling honeycomb meso-structures with either constant effective modulus in the loading direction, as described in Chapter 4, or with constant effective density and mass, as discussed in Chapter 5. It is proposed that any future work that takes place to analyze the response of honeycombs use the parameters and methods as defined in this report for both reliable and comparable results.

The design of experiments method described in Chapter 5 can apply to any dynamic, explicit simulation of honeycomb meso-structures, as the code written can be modified to generate models of any user-defined ABAQUS parameters. While the method directly applies to impact simulations, simple modifications to the python script can be made to change the necessary parameters to perform different types of simulations, including acoustic simulations as previously described.
WORKS CITED


APPENDICES
Appendix A. ABAQUS Modeling Descriptions for Chapter 4 Simulations

This section contains the necessary data for the re-creation of the simulations analyzed for this research. Parameters valid for all simulations described in Chapter 4 are defined in Table A-1, while the subsequent tables define the parameters specific to each of the four cellular geometries. All models are generated in Abaqus/CAE version 6.9-1. The input files are then submitted to Abaqus/Explicit version 6.9-1. All jobs are run on a Dell Precision T7400 PC, containing dual Intel® Xeon® E5405 processors @2.00 GHz, 16.0 GB RAM on a 64-bit Microsoft Windows Vista™ Enterprise operating system. One job is run at a time, each allotted the use 6 cores and 90% of the physical memory in the computer.

Table A-1: ABAQUS modeling parameters for all impact simulations

<table>
<thead>
<tr>
<th>Simulation Type</th>
<th>Dynamic, Explicit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Type</td>
<td>3D, deformable</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section Details</th>
<th>Shell / Continuum Shell, Homogeneous Section Integration During Analysis 5 thickness integration points, Simpson’s integration rule All other default settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Material Properties</td>
<td>Aluminum, $\rho = 2700$ kg/m$^3$, $E = 68$ GPa, $\nu = 0.33$ Elastic, perfectly plastic, $\sigma_y = 130$ MPa, $\varepsilon_{ult} = 0.8$</td>
</tr>
<tr>
<td>Element Type</td>
<td>S4R: Explicit, Linear Geometric Order, All other default settings</td>
</tr>
<tr>
<td>Partitions</td>
<td>Partitioned with plane parallel to XZ plane at midpoint of structure to split the faces along the middle</td>
</tr>
</tbody>
</table>

Sets created

- Distal Cells (geometry): the distal column of cells
- Distal Nodes (nodes): the nodes along the outer wall of the distal cells
- Middle Nodes (nodes): All nodes along the middle partition
- Proximal Cells (geometry): the proximal column of cells
- Proximal Nodes (nodes): the nodes along the outer wall of the proximal cells
- Stiff Block Nodes (nodes): all nodes in the stiff impacting body
- Proximal Stiff Nodes (nodes) all nodes on the inner face of the stiff impacting body
Boundary Conditions (on which sets)

- **Impact Velocity (Velocity / Angular Velocity):** Applied to Stiff Block Nodes, V1 = Impact Velocity, all other = 0, Amplitude: Velocity Magnitude
- **Fixed Distal End (Displacement / Rotation):** Applied to the outer faces on the distal end, U1 = 0
- **Out-of-plane Displacement (Displacement / Rotation):** Applied to all nodes in the model, U3 = 0
- **Symmetry (Displacement / Rotation):** Applied to Middle Nodes, U2 = 0

Step Details

- **Apply Load:** All default settings

History Output Request (on which sets)

- Distal Cell Data: RF1 on Set Distal Nodes, Interval: 200
- **H-Output-1:** ALLAE, ALLCW, ALLIE, ALLKE, ALLMW, ALLPD, ALLPW, ALLSE, ALLVD, ETOTAL on Whole Model, Interval: 200
- Proximal Cell Data: RF1 on Set Proximal Stiff Nodes, Interval: 200

Field Output Request

- **F-Output-1:** CSTRESS, EVF, LE, PE, PEEQ, PEEQAVG, PEVAVG, RF, S, SVAVG, U, V on Whole Model, Interval: 20

Interaction Properties

- Frictionless:
  - Tangential Behavior
  - Friction formulation: Frictionless
  - Normal Behavior:
  - Constraint enforcement method: Default
  - Pressure-Overclosure: “Hard” Contact
  - Allow separation after contact

Interactions

- **Contact:**
  - Type: General contact (Explicit)
  - Step: Apply Load
  - Contact Domain:
  - Included Surface pairs: All* with self
  - Attribute Assignments:
  - Global property assignment: Frictionless
  - All other default settings

Impact Velocities and Step Times

<table>
<thead>
<tr>
<th>Impact Velocity</th>
<th>Step Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 m/s</td>
<td>0.004 s</td>
</tr>
<tr>
<td>50 m/s</td>
<td>0.008 s</td>
</tr>
<tr>
<td>10 m/s</td>
<td>0.04 s</td>
</tr>
<tr>
<td>2 m/s</td>
<td>0.2 s</td>
</tr>
<tr>
<td>1 m/s</td>
<td>0.4 s</td>
</tr>
</tbody>
</table>
Table A-2 shows the defined parameters to generate the Type A honeycomb cellular meso-structure and its dependent effective static material properties. These parameters are applicable for all impact velocities.

Table A-2: Type A honeycomb ABAQUS modeling parameters

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Pos_30</th>
</tr>
</thead>
<tbody>
<tr>
<td>θ</td>
<td>+ 30°</td>
</tr>
<tr>
<td>Outer Geometry</td>
<td>0.510 m x 0.127 m x 0.004 m</td>
</tr>
<tr>
<td># Cells Lateral</td>
<td>74</td>
</tr>
<tr>
<td># Cells Transverse</td>
<td>21</td>
</tr>
<tr>
<td>t</td>
<td>0.000346 m</td>
</tr>
<tr>
<td>h</td>
<td>0.004 m</td>
</tr>
<tr>
<td>l</td>
<td>0.004 m</td>
</tr>
<tr>
<td>ρ*</td>
<td>269.7 kg/m³</td>
</tr>
<tr>
<td>E₁₁*</td>
<td>103 MPa</td>
</tr>
<tr>
<td>ε₁₁y*</td>
<td>0.0188</td>
</tr>
<tr>
<td>E₂₂*</td>
<td>103 MPa</td>
</tr>
<tr>
<td>ν₁₂*</td>
<td>1</td>
</tr>
<tr>
<td>ν₂₁*</td>
<td>1</td>
</tr>
</tbody>
</table>

Table A-3 shows the defined parameters to generate the Type B honeycomb cellular meso-structure and its dependent effective static material properties. These parameters are applicable for all impact velocities.

Table A-3: Type B honeycomb ABAQUS modeling parameters

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Pos_15</th>
</tr>
</thead>
<tbody>
<tr>
<td>θ</td>
<td>+ 15°</td>
</tr>
<tr>
<td>Outer Geometry</td>
<td>0.513 m x 0.128 m x 0.004 m</td>
</tr>
<tr>
<td># Cells Lateral</td>
<td>74</td>
</tr>
<tr>
<td># Cells Transverse</td>
<td>21</td>
</tr>
<tr>
<td>t</td>
<td>0.000203 m</td>
</tr>
<tr>
<td>h</td>
<td>0.004 m</td>
</tr>
<tr>
<td>l</td>
<td>0.004 m</td>
</tr>
<tr>
<td>ρ*</td>
<td>269.7 kg/m³</td>
</tr>
<tr>
<td>E₁₁*</td>
<td>103 MPa</td>
</tr>
<tr>
<td>ε₁₁y*</td>
<td>0.0188</td>
</tr>
<tr>
<td>E₂₂*</td>
<td>103 MPa</td>
</tr>
</tbody>
</table>
Table A-4 shows the defined parameters to generate the Type C honeycomb cellular meso-structure and its dependent effective static material properties. These parameters are applicable for all impact velocities.

**Table A-4: Type C honeycomb ABAQUS modeling parameters**

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Neg_15</th>
</tr>
</thead>
<tbody>
<tr>
<td>θ</td>
<td>-15°</td>
</tr>
<tr>
<td>Outer Geometry</td>
<td>0.510 m x 0.128 m x 0.004 m</td>
</tr>
<tr>
<td># Cells Lateral</td>
<td>74</td>
</tr>
<tr>
<td># Cells Transverse</td>
<td>21</td>
</tr>
<tr>
<td>t</td>
<td>0.00017 m</td>
</tr>
<tr>
<td>h</td>
<td>0.004 m</td>
</tr>
<tr>
<td>l</td>
<td>0.004 m</td>
</tr>
<tr>
<td>ρ*</td>
<td>269.7 kg/m³</td>
</tr>
<tr>
<td>E₁₁*</td>
<td>103 MPa</td>
</tr>
<tr>
<td>ε₁₁y*</td>
<td>0.0188</td>
</tr>
<tr>
<td>E₂₂*</td>
<td>101.6 MPa</td>
</tr>
<tr>
<td>ν₁₂*</td>
<td>1</td>
</tr>
<tr>
<td>ν₂₁*</td>
<td>1</td>
</tr>
</tbody>
</table>

Table A-5 shows the defined parameters to generate the Type D honeycomb cellular meso-structure and its dependent effective static material properties. These parameters are applicable for all impact velocities.

**Table A-5: Type D honeycomb ABAQUS modeling parameters**

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Neg_30</th>
</tr>
</thead>
<tbody>
<tr>
<td>θ</td>
<td>-30°</td>
</tr>
<tr>
<td>Outer Geometry</td>
<td>0.519 m x 0.128 m x 0.004 m</td>
</tr>
<tr>
<td># Cells Lateral</td>
<td>74</td>
</tr>
<tr>
<td># Cells Transverse</td>
<td>21</td>
</tr>
<tr>
<td>t</td>
<td>0.000346 m</td>
</tr>
<tr>
<td>h</td>
<td>0.008 m</td>
</tr>
<tr>
<td>l</td>
<td>0.004 m</td>
</tr>
<tr>
<td>ρ*</td>
<td>269.7 kg/m³</td>
</tr>
<tr>
<td>$E_{11}^*$</td>
<td>103 MPa</td>
</tr>
<tr>
<td>$\varepsilon_{11y}^*$</td>
<td>0.0188</td>
</tr>
<tr>
<td>$E_{22}^*$</td>
<td>103 MPa</td>
</tr>
<tr>
<td>$\nu_{12}^*$</td>
<td>-1</td>
</tr>
<tr>
<td>$\nu_{21}^*$</td>
<td>-1</td>
</tr>
</tbody>
</table>
A.1. Visual Descriptions of Results

This section provides visual representation and descriptions of the impact response of the four honeycomb structures to the five impact velocities. Each figure in this section shows the four structures at the initial impact (4% global compression), 20% compression, 40%, and finally 60% compression.
A.1.1. High Velocity Impact – 100 m/s

This section displays the four honeycomb structures’ response to 100 m/s impact and describes the crushing mechanisms observed.

A.1.1.1. Type A Response – Completion Time: 25 min - Figure A- 1

An initial “I”-shaped crushing band forms at the proximal end of the structure, which sequentially collapses adjacent columns of cells throughout the duration of the simulation. A “V” crushing band forms at the distal end near 60% global compression.

Figure A- 1: Type A response to 100 m/s impact
A.1.1.2. Type B Response – Completion Time: 31 min - Figure A- 2

Similar to the Type A structure, an “I” crushing band forms at the proximal end at the initial impact, which sequentially collapses adjacent columns of cells as the crushing progresses. The cells expand outwards as they flatten against the impacting solid. An “X” band forms near 60% compression.

Figure A- 2: Type B response to 100 m/s impact
A.1.1.3. Type C Response – Completion Time: 63 min - Figure A- 3

The proximal end of the structure experiences the bulk of the deformation at the initial impact with an “I” crushing band. The band collapses sequential columns of cells by initially pulling the outer cells inwards towards the centerline of the structure, causing a chaotic mass of fully collapsed cells that grows throughout the simulation. A buckling region forms near the distal end around 25% compression, which forms a second “I” band near 60% compression.

Figure A- 3: Type C response to 100 m/s impact
A.1.1.4. Type D Response – Completion Time: 36 min - Figure A- 4

The proximal cells initially collapse in an “I” band, which, similar to the Type A response, sequentially collapses adjacent columns of cells throughout the duration of the impact. However, because of the negative Poission’s ratio, the honeycomb structure collapses inwards onto itself, rather than outwards. The distal end begins to see some deformation around 50% global compression.

Figure A- 4: Type D response to 100 m/s impact
A.1.2. High Velocity Impact – 50 m/s

This section displays the four honeycomb structures’ response to 50 m/s impact and describes the crushing mechanisms observed.

A.1.2.1. Type A Response – Completion Time: 34 min - Figure A- 5

The proximal end experiences the bulk of the deformation, collapsing in a loose “I” crushing band throughout the impact. A prominent “V” band forms around 40% compression, which grows until the end of the simulation.

Figure A- 5: Type A response to 50 m/s impact
A.1.2.2. Type B Response – Completion Time: 37 min - Figure A- 6

The Type B response is similar to that of Type A, but shows a larger outward expansion at the proximal end from the crushing, and less order to the collapse of the honeycomb cells. A “V” crushing band forms around 40% compression and continues to collapse adjacent columns of cells until the end of the simulation.

Figure A- 6: Type B response to 50 m/s impact
Type C reacted to the initial impact with a loose “I”-crushing band at the proximal end of the structure. The band collapsed adjacent columns of cells into a non-uniform region of fully collapsed cells near the proximal end. A secondary buckling region forms near the distal end of the structure near 10% compression. This becomes more prominent throughout the simulation, initially growing inwards towards the centerline of the structure, then collapsing adjacent columns of cells in the loading direction.

Figure A- 7: Type C response to 50 m/s impact
A.1.2.4. Type D Response – Completion Time: 58 min - Figure A- 8

The Type B structure responds similar to Type A in the formation of the proximal end “I”-shaped crushing band, which grows throughout the simulation. A loose “I” crushing band forms at the distal end at about 20% compression, which slowly grows towards the proximal end as the structure is crushed further. The two crushing bands eventually meet, showing deformation in all cells of the structure.

Figure A- 8: Type D response to 50 m/s impact
A.1.3. Moderate Velocity Impact – 10 m/s

This section displays the four honeycombs’ response to 10 m/s impact and describes the crushing mechanisms observed.

A.1.3.1. Type A Response – Completion Time: 101 min - Figure A- 9

An initial “V”-shaped crushing band forms near the proximal end of the Type A honeycomb structure, then another at the distal end. These bands grow throughout the simulation by collapsing adjacent columns of cells until they meet at near 70% global compression.

Figure A- 9: Type A response to 10 m/s impact
A.1.3.2. Type B Response – Completion Time: 112 min - Figure A-10

The Type B honeycomb structure initially forms an “X”-shaped crushing region at the proximal end of the structure, which collapses to a loose “V” band in front of the fully collapsed proximal cells. The same occurs at the distal end near 30% compression. The “V” bands grow on either end until they meet each other near 70% global compression.

Figure A-10: Type B response to 10 m/s impact
A.1.3.3. Type C Response – Completion Time: 214 min - Figure A- 11

Type C responds to the initial impact with an “l”-shaped crushing band forming at the proximal end simultaneously with a buckling region near the distal end. These two regions then grow through the sequential collapse of adjacent columns of cells, until all cells within the structure collapse around 60% compression.

Figure A- 11: Type C response to 10 m/s impact
A.1.3.4. Type D Response – Completion Time: 160 min - Figure A- 12

Both the proximal end distal ends of the Type D structure collapse inwards throughout the simulation, collapsing adjacent columns of cells to grow in size. These two regions meet at near 50% compression, then begin to further densify into tight “I”-shaped columns until the structure is fully crushed near 80% compression.

Figure A- 12: Type D response to 10 m/s impact
A.1.4. Low Velocity Impact – 2 m/s

This section displays the four honeycomb structures’ response to 100 m/s impact and describes the crushing mechanisms observed.

A.1.4.1. Type A Response – Completion Time: 9 hr 7 min - Figure A- 13

The Type A structure forms an initial “X”-shaped crushing band near the proximal end of the structure, which transforms to a “V” band as it crushes further. The distal end begins to form an additional “V” band near 30% compression. These two bands sequentially collapse adjacent “V”-shaped columns of cells, crushing outwards until the structure densifies near 80% global compression.

Figure A- 13: Type A response to 2 m/s impact
A.1.4.2. Type B Response – Completion Time: 8 hr 32 min - Figure A-14

A prominent “X”-shaped crushing region initially forms in the Type B structure, then collapses outwards, forming loose “V” bands with a chaotic crushing region at its apex. This region grows throughout the simulation through the collapse of adjacent columns of cells, leaving the distal end free from deformation until near 70% global compression.

Figure A-14: Type B response to 2 m/s impact
A.1.4.3. Type C Response – Completion Time: 16 hr 51 min - Figure A-15

Type C responds with an initial “I” band with an interior buckling region near the centerline of the structure. This band collapses adjacent columns of cells at the proximal end throughout the simulation. The distal end remains free of deformation until near 60% global compression.

Figure A-15: Type C response to 2 m/s impact
A.1.4.4. Type D Response – Completion Time: 11 hr 48 min - Figure A- 16

Both the proximal and distal ends of the Type D structure deform at constant rates throughout the simulation. The proximal end collapses adjacent columns of cells at a rate of nearly twice that of the distal end. The structure begins to densify near 50% global compression, when the crushing bands from the proximal and distal ends meet.

Figure A- 16: Type D response to 2 m/s impact
A.1.5. Low Velocity Impact – 1 m/s

The following sections describe the crushing mechanisms observed in the simulation of 1 m/s impact on the honeycomb meso-structures.

A.1.5.1. Type A Response – Completion Time: 16 hr 38 min - Figure A-17

An “X”-shaped crushing region forms at the proximal end of the Type A structure due to the initial impact, which transforms to a “V” band as it is crushed further. A second “X” crushing region is formed near the middle of the structure in the loading direction, near 30% compression, which expands outwards by collapsing adjacent columns of cells in either direction until the structure is fully crushed, leaving the distal end free from deformation until 80% compression.

Figure A-17: Type A response to 1 m/s impact
A.1.5.2. Type B Response – Completion Time: 16 hr 34 min - Figure A-18

The Type B structure initially responds with an “X”-shaped crushing region at the proximal end, which collapses on itself to form a loose “V” band as it is further crushed. The cells adjacent to the impacting body expand outwards, while the “V” band grows through the crushing of sequential columns of cells. The distal end remains free from deformation until the end of the simulation.
The Type C structure’s initial response consists of both a proximal-end “I” band and a buckling region near the middle of the structure in the loading direction. The proximal end experiences the majority of the deformation through the sequential collapse of adjacent columns of cells, until the columns within the buckling region collapse near 50% global compression. These two regions join together, then continue to collapse adjacent columns of cells until the collapse of nearly all columns within the structure near 60% compression.

Figure A-19: Type C response to 1 m/s impact
A.1.5.4. Type D Response – Completion Time: 22 hr 45 min - Figure A- 20

The proximal end of the Type D structure experiences the entirety of the deformation during the simulation via the inwards collapse of columns of cells. The partially collapsed columns of cells reach the distal end near 50% global compression, then begin to densify into a fully collapsed configuration.

Figure A- 20: Type D response to 1 m/s impact
Appendix B. Sequential Simulation Source Code

This Appendix includes the source code written to implement the sequential simulation method described in Chapter 5. It is divided into two sections; the first is the code written in MATLAB to generate a Design of Experiments (DOE) table based on the desired inputs, calculate the dependent modeling parameters, edits the two python scripts that generate, run, and process the results of each ABAQUS job, and finally calculates the total plastic energy absorbed by each honeycomb structure. The second part is an example of the python scripts that are edited.

B.1. MATLAB Source Code

This section provides the code written in MATLAB to generate the DOE based on the input parameters $t_2$ and $\theta$, calculates the necessary dependent modeling parameters for generating the ABAQUS simulation models.

B.1.1. DOE.m

This program calls the functions listed in this section to create a DOE table, run each simulation, and store the results in an output matrix. It runs the DOE without the need of a third-party optimization program such as modeFRONTIER or Isight.

```matlab
number = 50; % # of runs in the DOE
data1 = generateDOE(number); % Calculates the DOE table
output = zeros(number,4); % Sets up the results table
for j = 1:number % Writes the results with each iteration
t_2 = data1(j,2);
angle = data1(j,3);
run = j;
Main
output(j,:) = [run t_2 angle W]; % Runs the main program
end
```
B.1.2. generateDOE.m

This function will generate the DOE table based on the specified number of runs.

This case uses uniformly distributed random values within the specified domain.

```matlab
function [data1] = generateDOE(number)
  t_2_bounds = [0.0001 0.0005]; % angled wall thickness domain
  theta_bounds = [75 150]; % cell angle domain
  data1 = zeros(number,3); % sets up the DOE table
  data1(:,1) = (1:number)'; % experiment #
  data1(:,2) = random('unif',t_2_bounds(1),t_2_bounds(2),number,1)'; % thickness
  data1(:,3) = random('unif',theta_bounds(1),theta_bounds(2),number,1)'; % angle
end
```

B.1.3. Main.m

The main program calls the two functions that edit the python scripts, runs ABAQUS from a .dos prompt, and interprets the results as the plastic energy absorbed by each honeycomb configuration.

```matlab
% Set the current folder, allowing ABAQUS to write the results in the same % folder.
cd('C:\Users\Jesse\Documents\Thesis\Optimization Code & Results')
% -read run number on line 443 to find the job name-------------------------% replaceLine = 443;
fid = fopen('python_template_final.py','r');
for k=1:(replaceLine-1);
  fgetl(fid);
end;
fseek(fid, 0, 'cof');
r1 = fread(fid, 15,'*char');
a = r1(14); b = r1(15);
run = str2num(strcat(a,b));
fclose(fid);
% Assemble the initial python script---------------------------------------% [length] = python_assembly(angle,t_2,run);
% Run python script in ABAQUS to generate model and submit job-----------% dos('abq691 cae noGUI=python_template_final.py')
% Assemble the get_data python script-------------------------------------% getdata(run);
% Run python script in ABAQUS to save the recorded data from the .odb file% dos('abq691 cae noGUI=Get_data_template.py')
% Import the recorded Proximal and Distal Force reactions
Proximal = importfile('Proximal.dat');
Proximal = data(:,2);
Distal = importfile('Distal.dat');
Distal = data(:,2);
D_max = max(Distal);
% Calculate the Work done by the honeycomb structure (Objective Function)
W = 0;
for i = 1:201
  Diff = Proximal(i)+Distal(i);
  W = W + Diff*length/200;
end
```
B.1.4. Python\_assembly.m

This function reads and edits a python template (seen in section B.1.7) to generate an ABAQUS model for each cellular configuration, using the input parameters from the DOE table.

```matlab
function [length] = python_assembly(angle,t_2,run)
%Constants============================================
%=l = .004; %Length (constant)
%=h = .004; %Height (constant)
%=V\_honeycomb = 2.581E-5; %Volume (constant)
%=v = 100; %Impact Velocity (constant)
%Calculate values for spacings and time period based on inputs---------%
format long
[vert\_space,hor\_space,angle,Time\_period] = honeycomb(l,h,angle);
d = hor\_space;
%Thickness 1 (based off of Thickness 2)
t_1 = V\_cell/(h\_1) - 2\_t\_2;
%--------Matlab will write the python script, adding in the---------%
%----------calculated/changed values at select lines----------%
%replace line 46 with new value for the angle----------%
replaceLine = 46;
 fid = fopen('python\_template\_final.py','r+');
 for k=1:(replaceLine-1);
 fgetl(fid);
 end;
 fseek(fid, 0, 'cof');
 fprintf(fid, ' -0.135464668273926), value=%8.3f', angle(1));
 fclose(fid);
%replace line 48 with new value for the angle----------%
replaceLine = 48;
 fid = fopen('python\_template\_final.py','r+');
 for k=1:(replaceLine-1);
 fgetl(fid);
 end;
 fseek(fid, 0, 'cof');
 fprintf(fid, ' -0.13254602253437), value=%8.3f', angle(1));
 fclose(fid);
%replace line 59 with new value for vertical spacing----------%
replaceLine = 59;
 fid = fopen('python\_template\_final.py','r+');
 for k=1:(replaceLine-1);
 fgetl(fid);
 end;
 fseek(fid, 0, 'cof');
 fprintf(fid, ' spacing2=%8.6f, angle2=90.0)', vert\_space(1));
 fclose(fid);
%replace line 68 with new value for horizontal spacing----------%
replaceLine = 68;
 fid = fopen('python\_template\_final.py','r+');
 for k=1:(replaceLine-1);
 fgetl(fid);
 end;
 fseek(fid, 0, 'cof');
 fprintf(fid, ' number1=74, spacing1=%8.6f, angle1=0.0, number2=1, spacing2=0.012,',
 hor\_space(1));
 fclose(fid);
%replace line 220 with new value for t\_1----------%
replaceLine = 220;
 fid = fopen('python\_template\_final.py','r+');
 for k=1:(replaceLine-1);
```

115
fgetl(fid);
end;
fclose(fid);

% replace line 224 with new value for t_2-----------------------------
replaceLine = 224;
for k=1:(replaceLine-1);
    fgetl(fid);
end;
fseek(fid, 0, 'cof');
fprintf(fid,'    material=''Aluminum'', thicknessType=UNIFORM, thickness=%8.6f, t_2(1));
fclose(fid);

% replace line 287 with new value for timePeriod----------------------
replaceLine = 287;
for k=1:(replaceLine-1);
    fgetl(fid);
end;
fseek(fid, 0, 'cof');
fprintf(fid,'    previous=''Initial'', timePeriod=%8.6f)', Time_period(1));
fclose(fid);

% length of the crush
length = Time_period*v;
% replace line 404 with new job name----------------------------------
replaceLine = 404;
for k=1:(replaceLine-1);
    fgetl(fid);
end;
fseek(fid, 0, 'cof');
fprintf(fid,'mdb.Job(name=''Run%02g'', model=''Model-1'', description='', run(1));
fclose(fid);

% replace line 443 with new job name-----------------------------------
replaceLine = 443;
for k=1:(replaceLine-1);
    fgetl(fid);
end;
fseek(fid, 0, 'cof');
fprintf(fid,'mdb.jobs[''Run%02g''].submit(consistencyChecking=OFF)', run(1));
fclose(fid);

B.1.5. Honeycomb.m

This function calculates the spacing necessary between cells for the generation of
the ABAQUS models used for simulation.

function [vert_space,hor_space,angle,Time_period] = honeycomb(l,h,angle)
angle1 = angle-90;
theta = angle1*2*pi/360;
vert_space = 2*h + 2*l*sin(theta);
hor_space = 2*l*cos(theta);
Time_period = hor_space*74*0.8/100;
B.1.6. Getdata.m

This function reads and edits a second python script (seen in section B.1.8) that
opens the ABAQUS results .odb file and exports the results to .txt files.

```plaintext
function[] = getdata(run)
% replace line 22 with new file name----------------------------------------
replaceLine = 22;
fid = fopen('Get_data_template.py','r+');
for k=1:(replaceLine-1);
  fgetl(fid);
end;
fsseek(fid, 0, 'cof');
fprintf(fid,'    name=''C:/Users/Jesse/Documents/Thesis/Optimization Code & Results/Run%02g.odb''), run(1));
close(fid);
replaceLine = 32;
fid = fopen('Get_data_template.py','r+');
for k=1:(replaceLine-1);
  fgetl(fid);
end;
fsseek(fid, 0, 'cof');
fprintf(fid,'odb = session.odbs[''C:/Users/Jesse/Documents/Thesis/Optimization Code & Results/Run%02g.odb''], run(1));
close(fid);
```

B.1.7. Python_template.py

This is an example python script that can be used to generate a honeycomb impact
simulation similar to those described in Chapter 5. The template is initially generated by
recording the steps taken to generate a simulation using ABAQUS CAE. Specific lines of
the template are then edited per the input parameters from the DOE table,

```plaintext
# -*- coding: mbcs -*-
#
# Abaqus/CAE Release 6.9-1 replay file
# Internal Version: 2009_04_16-15.31.05 92314
# Run by Jesse on Mon Nov 22 15:09:27 2010
#
# from driverUtils import executeOnCaeGraphicsStartup
# executeOnCaeGraphicsStartup()
#: Executing "onCaeGraphicsStartup()" in the site directory ...
from abaqus import *
from abaqusConstants import *
session.Viewport(name='Viewport: 1', origin=(0.0, 0.0), width=274.968475341797,
  height=253.361892700195)
session.viewports['Viewport: 1'].makeCurrent()
session.viewports['Viewport: 1'].maximize()
from caeModules import *
from driverUtils import executeOnCaeStartup
executeOnCaeStartup()
```
Mdb()

#: A new model database has been created.
#: The model "Model-1" has been created.
s = mdb.models['Model-1'].ConstrainedSketch(name='__profile__', sheetSize=1.0)
g, v, d, c = s.geometry, s.vertices, s.dimensions, s.constraints
s.setPrimaryObject(option=STANDALONE)
s.Line(point1=(-0.159874156117439, -0.138588175177574), point2=(-0.159874156117439, -0.135))
s.VerticalConstraint(entity=g[2])
s.Line(point1=(-0.159874156117439, -0.135), point2=(-0.15799112178534, -0.13997323770523))
s.Line(point1=(-0.15799111278534, -0.13397323770523), point2=(-0.156122997403145, -0.134994566440582))
s.Line(point1=(-0.156122997403145, -0.138611480593681), point2=(-0.157708063721657, -0.140024334192276))
s.Line(point1=(-0.157708063721657, -0.140024334192276), point2=(-0.158588175177574, -0.138611480593681))
s.VerticalConstraint(entity=g[5])
s.Line(point1=(-0.158588175177574, -0.138611480593681), point2=(-0.157708063721657, -0.136869030952454))
s.VerticalConstraint(entity=g[8])
s.EqualLengthConstraint(entity1=g[2], entity2=g[8])
s.EqualLengthConstraint(entity1=g[3], entity2=g[7])
s.AngularDimension(line1=g[2], line2=g[7], textPoint=(-0.175915062427521, -0.135464668273926), value=75.167)
s.AngularDimension(line1=g[3], line2=g[2], textPoint=(-0.178080677986145, -0.13254602253437), value=75.167)
s.SymmetryConstraint(entity1=g[3], entity2=g[4], symmetryAxis=g[8])
s.SymmetryConstraint(entity1=g[2], entity2=g[5], symmetryAxis=g[8])
s.ObliqueDimension(vertex1=v[0], vertex2=v[1], textPoint=(-0.164727658033371, -0.136520460247993), value=0.004)
s.ObliqueDimension(vertex1=v[0], vertex2=v[1], textPoint=(-0.162802934646606, -0.13416929721832), value=0.004)
s.delete(objectList=(d[0], d[1], d[2], d[3])))
s.delete(objectList=(c[4], c[11], c[18], c[19], c[20], c[21]))
s.linearPattern(geomList=(g[2], g[3], g[4], g[5], g[6], g[7], g[8]), vertexList=(), number1=1, spacing1=0.1, angle1=0.0, number2=11, spacing2=0.007733, angle2=90.0)
s.linearPattern(geomList=(g[3], g[4], g[5], g[6], g[7], g[8], g[10], g[11], g[12], g[13], g[14], g[15], g[17], g[18], g[19], g[20], g[21], g[22], g[24], g[25], g[26], g[27], g[28], g[29], g[31], g[32], g[33], g[34], g[35], g[36], g[38], g[39], g[40], g[41], g[42], g[43], g[45], g[46], g[47], g[49], g[50], g[52], g[53], g[55], g[56], g[57], g[59], g[60], g[61], g[62], g[63], g[64], g[66], g[67], g[68], g[69], g[70], g[71], g[73], g[74], g[75], g[76], g[77]), vertexList=(), number1=74, spacing1=0.007733, angle1=0.0, number2=1, spacing2=0.012, angle2=90.0)
p = mdb.models['Model-1'].Part(name='Honeycomb', dimensionality=THREE_D, type=DEFORMABLE_BODY)
p = mdb.models['Model-1'].parts['Honeycomb']
p.BaseShellExtrude(sketch=s, depth=0.004)
s.unsetPrimaryObject()
p = mdb.models['Model-1'].parts['Honeycomb']
delete mdb.models['Model-1'].sketches['__profile__']
sl = p.features['Shell extrude-1'].sketch
p = mdb.models['Model-1'].parts['Honeycomb']
f = p.faces
faces = f.getSequenceFromMask(mask=('[#800800 #200000 #800 #800000 #2000000 #8000000 #80000000 ]', ), )
p.Set(faces=faces, name='Distal Faces')
#: The set 'Distal Faces' has been created (11 faces).
p = mdb.models['Model-1'].parts['Honeycomb']
f = p.faces
faces = f.getSequenceFromMask(mask=([#6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #7db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #3deb6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #3deb6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #3deb6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #3deb6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #3deb6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6, #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6 #6db6db6]),
   p.Set(faces=faces, name='Angled Walls')
#: The set 'Angled Walls' has been created (3256 faces).
  p = mdb.models['Model-1'].parts['Honeycomb']
  f = p.faces
faces = f.getSequenceFromMask(mask=([#49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249, #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249 #49249249]),
   p.Set(faces=faces, name='Vertical Walls')
#: The set 'Vertical Walls' has been created (1565 faces).
  elemType1 = mesh.ElemType(elemCode=S4R, elemLibrary=EXPLICIT, secondOrderAccuracy=OFF, hourglassControl=DEFAULT)
  elemType2 = mesh.ElemType(elemCode=S3R, elemLibrary=EXPLICIT)
  p = mdb.models['Model-1'].parts['Honeycomb']
  f = p.faces
faces = f.getSequenceFromMask(mask=([#6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66, #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66 #6b6db66]),
   p.Set(faces=faces, name='Vertical Walls')
#: The set 'Vertical Walls' has been created (1565 faces).
  elemType1 = mesh.ElemType(elemCode=S4R, elemLibrary=EXPLICIT, secondOrderAccuracy=OFF, hourglassControl=DEFAULT)
  elemType2 = mesh.ElemType(elemCode=S3R, elemLibrary=EXPLICIT)
pickedRegions = (faces, )
p.setElementType(regions=pickedRegions, elemTypes=(elemType1, elemType2))
elemType1 = mesh.ElemType(elemCode=S4R, elemLibrary=EXPLICIT, secondOrderAccuracy=OFF, hourglassControl=DEFAULT)
elemType2 = mesh.ElemType(elemCode=S3R, elemLibrary=EXPLICIT)
p = mdb.models['Model-1'].parts['Honeycomb']
f = p.faces
faces = f.getSequenceFromMask(mask=(' #b6db6db6 #6db6db6d #db6db6db #b6db6db6 #6db6db6d #db6db6db #36db6db6', ' #db6db6db #b6db6db6 #b6db6cf7 #6db6db6d #6db3dedb #db6db6db #cf7b6db6', ' #6db6db6d #7b6db6db #6db6db6c #6db6db6c #6db6db6b #6cf7b6d', ' #6db6db6d #3deb6d6 #6db6db6b #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6', ' #6deb6d6 #cf7b6db6 #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6', ' #6deb6d6 #b3deb6d6 #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6', ' #6deb6d6 #6cf7db6b #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6 #6deb6d6', ' #f7b6db6b #f7b6db6b #f7b6db6b #f7b6db6b #f7b6db6b #f7b6db6b #f7b6db6b', ' #cb7b5bb3 #aade7bde #aaaa #9ef3ffff #c79e7bc7 #79e79ef3 #c79e79e7', ' #79e79e79 #effde79e #79e79e79 #79e79e79 #effde79e #79e79e79 #79e79e79', ' #8f3cf3cd #3cf379e7 #f3cde78f #cf378c7 #79e79e79 #79e79e79'), )
pickedRegions = (faces, )
p.setElementType(regions=pickedRegions, elemTypes=(elemType1, elemType2))
p = mdb.models['Model-1'].parts['Honeycomb']
p.seedPart(size=0.002, deviationFactor=0.1)
p = mdb.models['Model-1'].parts['Honeycomb']
p.generateMesh()

mdb.models['Model-1'].Material(name='Aluminum')

mdb.models['Model-1'].materials['Aluminum'].Density(table=((2700.0, ), ))

mdb.models['Model-1'].materials['Aluminum'].Elastic(table=((68000000000.0, 0.33), ))

mdb.models['Model-1'].materials['Aluminum'].Plastic(table=((130000000.0, 0.0), (130000000.0, 0.8)))

mdb.models['Model-1'].HomogeneousShellSection(name='t1', preIntegrate=OFF, material='Aluminum', thicknessType=UNIFORM, thickness=0.004, thicknessField='', idealization=NO_IDEALIZATION, poissonDefinition=DEFAULT, thicknessModulus=None, temperature=GRADIENT, useDensity=OFF, integrationRule=SIMPSON, numIntPts=5)

mdb.models['Model-1'].HomogeneousShellSection(name='t2', preIntegrate=OFF, material='Aluminum', thicknessType=UNIFORM, thickness=0.002, thicknessField='', idealization=NO_IDEALIZATION, poissonDefinition=DEFAULT, thicknessModulus=None, temperature=GRADIENT, useDensity=OFF, integrationRule=SIMPSON, numIntPts=5)
thicknessField='', idealization=NO_IDEALIZATION, integrationRule=SIMPSON, numIntPts=5)
mdb.models['Model-1'].sections['t2'].setValues(preIntegrate=OFF,
material='Aluminum', thicknessType=UNIFORM, thickness=0.000499,
thicknessField='', idealization=NO_IDEALIZATION, integrationRule=SIMPSON,
umIntPts=5)
s = mdb.models['Model-1'].ConstrainedSketch(name='__profile__', sheetSize=1.0)
g, v, d, c = s.geometry, s.vertices, s.dimensions, s.constraints
s.setPrimaryObject(option=STANDALONE)
s.rectangle(point1=(0.0, 0.12), point2=(0.005,
-0.165))
s.ObliqueDimension(vertex1=v[0], vertex2=v[1], textPoint=(
-0.0742377936840057,
0.0314871370792389), value=0.25)
s.DistanceDimension(entity1=g[2], entity2=g[4], textPoint=(
0.0120153576135635,
0.103029355406761), value=0.004)
p = mdb.models['Model-1'].Part(name='Steel Bar', dimensionality=THREE_D,
type=DEFORMABLE_BODY)
p = mdb.models['Model-1'].parts['Steel Bar']
p.BaseSolidExtrude(sketch=s, depth=0.004)
s.unsetPrimaryObject()
session.viewports["Viewport: 1"].assemblyDisplay.setValues(step='Apply Load')
mdb.models['Model-1'].historyOutputRequests.changeKey(fromName='H-Output-1',
toName='Energy')
pl = mdb.models['Model-1'].parts['Honeycomb']
p = mdb.models['Model-1'].parts['Honeycomb']
n = p.nodes
defines = n.getSequenceFromMask(mask=(
    ' #c00003c0 #3c00003 #3c00000 #3c00000 #0 #3c00',
    ' #3c00000 #0 #3c00000 #0 #3c00000 #0',
    ' #0:191 #1d0 #1d00 #1d000 #0 #740',
    ' #d000000 #1 #0 #1d0 #0 #1d00 #0:2',
    ' #740 #0:2 #1d #0:2 #1d000 #0:382 #800000',
    ' #8008008 #80002000 #8000000 #8000000 #2000 #800 #8000 ', ), )
p.Set(nodes=defines, name='Distal Nodes')
#: The set 'Distal Nodes' has been created (99 nodes).
pp = mdb.models['Model-1'].parts['Honeycomb']
n = pp.nodes
nodes = n.getSequenceFromMask(mask=(
    ' #0:211 #40500000 #1 #0:2 #a02800000 #0:4 #40500000',
    ' #1 #0:4 #500000000 #140 #0:5 #50a000000 #2',
    ' #0:5 #50140000 #0:6 #a0280 #0:6 #14050 #0:5',
    ' #a00000 #28 #0:5 #501400000 #0:6 #a02800000 #0:6',
    ' #14050 #0:5 #a0000000 #280 #0:5 #14000000 #5',
    ' #0:5 #a028000 #0:6 #14050000 #0:6 #280a #0:5',
    ' #14000000 #50 #0:5 #a02800000 #0:6 #140500000 #0:6',
    ' #a050 #0:5 #400000000 #501 #0:5 #28000000 #a',
    ' #a050 #1 #405000000 #6 #280a00000 #0:6 #50140000',
    ' #280000000 #a0 #0:5 #405000000 #1 #0:5 #280a00000',
    ' #0:6 #50140 #0:8 #a0280 #0:7 #14050 #0:6',
    ' #a028000 #0:5 #14000000 #5 #0:4 #a02800000 #0',
    ' #28140a0 #5 #0:15 #1400 #0:151 #9000000000 #0',
    ' #240 #9000000000 #4 #0 #120000',
    ' #0 #480000000 #1200000000 #0:2 #480000000 #0:2 #120000',
    ' #480000000 #1200000000 #0:2 #480000000 #0:2 #120000',
    ' #480000000 #0:2 #1200000000 #0:2 #480000000 #0:2 #120000',
    ' #1200000000 #0:2 #480000000 #0:2 #1200000000 #0:2 #480000000',
    ' #0 #1200000000 #0:2 #480000000 #0 #1200000000 #0:2 #480000000',
    ' #1200000000 #0:2 #480000000 #0 #1200000000 #0:2 #480000000',
    ' #480000000 #0:2 #1200000000 #0:2 #480000000 #0:2 #1200000000',
    ' #249 #0:5 #1 ', ), )
p.Set(nodes=defines, name='Middle Nodes')
#: The set 'Middle Nodes' has been created (225 nodes).
pp = mdb.models['Model-1'].parts['Honeycomb']
n = pp.nodes
nodes = n.getSequenceFromMask(mask=(
    ' #ffffffff:759 #1ff ', ), )
p.Set(nodes=names, name='Honeycomb Nodes')
#: The set 'Honeycomb Nodes' has been created (24297 nodes).
session.viewports["Viewport: 1"].partDisplay.setValues(mesh=OFF)
a = mdb.models['Model-1'].rootAssembly
b = mdb.models['Model-1'].rootAssembly
b.DatumCsysByDefault(CARTESIAN)
p = mdb.models['Model-1'].parts['Honeycomb']
a.Instance(name='Honeycomb-1', part=p, dependent=ON)
a = mdb.models['Model-1'].rootAssembly
b.translate(instanceList=('Honeycomb-1',), vector=(0.159874, 0.076588, -0.002))
#: The instance Honeycomb-1 was translated by 159.874E-03, 76.588E-03, -2.E-03 with respect to the assembly coordinate system
session.viewports["Viewport: 1"].view.setValues(session.views['Iso'])
a = mdb.models['Model-1'].rootAssembly
b = mdb.models['Model-1'].parts['Steel Bar']
a.Instance(name='Steel Bar-1', part=p, dependent=ON)
session.viewports["Viewport: 1"].view.setValues(nearPlane=0.963179, farPlane=1.45966, width=0.262211, height=0.195847, viewOffsetX=0.124756, viewOffsetY=0.0435168)
a = mdb.models['Model-1'].rootAssembly
The instance Steel Bar-1 was translated by -8.E-03, 40.E-03, -2.E-03 with respect to the assembly coordinate system.

regionDef = mdb.models['Model-1'].rootAssembly.instances['Honeycomb-1'].sets['Distal Nodes']

mdb.models['Model-1'].HistoryOutputRequest(name='Distal Forces', createStepName='Apply Load', variables=('RF1', ), region=regionDef, sectionPoints=DEFAULT, rebar=EXCLUDE)

regionDef = mdb.models['Model-1'].rootAssembly.instances['Honeycomb-1'].sets['Honeycomb Nodes']

mdb.models['Model-1'].HistoryOutputRequests['Energy'].setValues(region=regionDef, sectionPoints=DEFAULT, rebar=EXCLUDE)

mdb.models['Model-1'].HistoryOutputRequest(name='Proximal Forces', createStepName='Apply Load', variables=('RF1', ), region=regionDef, sectionPoints=DEFAULT, rebar=EXCLUDE)

mdb.models['Model-1'].ContactProperty('Frictionless')

mdb.models['Model-1'].interactionProperties['Frictionless'].TangentialBehavior(formulation=FRICTIONLESS)

mdb.models['Model-1'].interactionProperties['Frictionless'].NormalBehavior(pressureOverclosure=HARD, allowSeparation=ON, constraintEnforcementMethod=DEFAULT)

The interaction property "Frictionless" has been created.

mdb.models['Model-1'].ContactExp(name='Contact', createStepName='Apply Load')

mdb.models['Model-1'].interactions['Contact'].includedPairs.setValuesInStep(stepName='Apply Load', useAllstar=ON)

mdb.models['Model-1'].interactions['Contact'].contactPropertyAssignments.appendInStep(stepName='Apply Load', assignments=((GLOBAL, SELF, 'Frictionless'), ))

The interaction "Contact" has been created.

a = mdb.models['Model-1'].rootAssembly
region = ainstances['Honeycomb-1'].sets['Honeycomb Nodes']

mdb.models['Model-1'].DisplacementBC(name='Out of plane displacement', createStepName='Apply Load', region=region, u1=UNSET, u2=UNSET, u3=0.0, ur1=0.0, ur2=0.0, ur3=UNSET, amplitude=UNSET, fixed=OFF, distributionType=UNIFORM, fileName='', localCsys=None)

a = mdb.models['Model-1'].rootAssembly
region = a.instances['Honeycomb-1'].sets['Middle Nodes']

mdb.models['Model-1'].DisplacementBC(name='Symmetry', createStepName='Apply Load', region=region, u1=UNSET, u2=0.0, u3=UNSET, ur1=0.0, ur2=0.0, ur3=0.0, amplitude=UNSET, fixed=OFF, distributionType=UNIFORM, fileName='', localCsys=None)

a = mdb.models['Model-1'].rootAssembly
region = a.instances['Honeycomb-1'].sets['Distal Faces']

mdb.models['Model-1'].DisplacementBC(name='Fixed Distal Faces', createStepName='Apply Load', region=region, u1=0.0, u2=UNSET, u3=UNSET, ur1=UNSET, ur2=UNSET, ur3=UNSET, amplitude=UNSET, fixed=OFF, distributionType=UNIFORM, fileName='', localCsys=None)

a = mdb.models['Model-1'].rootAssembly
region = a.instances['Steel Bar-1'].sets['Steel Nodes']

mdb.models['Model-1'].VelocityBC(name='Impact Velocity', createStepName='Apply Load', region=region, v1=100.0, v2=0.0, v3=0.0, vr1=0.0, vr2=0.0, vr3=0.0, amplitude=UNSET, localCsys=None, distributionType=UNIFORM, fieldName='')

mdb.Job(name='Run53', model='Model-1', description='', type=ANALYSIS, atTime=None, waitMinutes=0, waitHours=0, queue=None, memory=90, memoryUnits=PERCENTAGE, getMemoryFromAnalysis=True, explicitPrecision=SINGLE, nodalOutputPrecision=SINGLE, echoPrint=OFF, historyPrint=OFF, userSubroutine='', scratch='', multiprocessingMode=DOMAIN, parallelizationMethodExplicit=DOMAIN, numDomains=6, numCpus=6)

p1 = mdb.models['Model-1'].parts['Honeycomb']
session.viewports['Viewport: 1'].setValues(displayedObject=p1)

p = mdb.models['Model-1'].parts['Honeycomb']
region = p.sets['Vertical Walls']
p = mdb.models['Model-1'].parts['Honeycomb']
p.SectionAssignment(region=region, sectionName='t1', offset=0.0,
p = mdb.models['Model-1'].parts['Honeycomb']
region = p.sets['Angled Walls']
p = mdb.models['Model-1'].parts['Honeycomb']
p.SectionAssignment(region=region, sectionName='t2', offset=0.0, offsetType=MIDDLE_SURFACE, offsetField='')
session.viewports['Viewport: 1'].partDisplay.setValues(sectionAssignments=OFF, engineeringFeatures=OFF)
p = mdb.models['Model-1'].parts['Steel Bar']
session.viewports['Viewport: 1'].setValues(displayedObject=p)
session.viewports['Viewport: 1'].partDisplay.setValues(sectionAssignments=ON, engineeringFeatures=ON)
p = mdb.models['Model-1'].parts['Steel Bar']
cells = c.getSequenceFromMask(mask=('[#3 ]', ), )
region = regionToolset.Region(cells=cells)
p = mdb.models['Model-1'].parts['Steel Bar']
p.SectionAssignment(region=region, sectionName='Steel Bar', offset=0.0, offsetType=MIDDLE_SURFACE, offsetField='')
a = mdb.models['Model-1'].rootAssembly
mdb.saveAs(pathName='C:/Users/Jesse/Documents/Thesis/Optimization Code & Results/Model')
session.viewports['Viewport: 1'].view.setValues(nearPlane=1.09183, farPlane=1.21933, width=0.662445, height=0.494784, cameraPosition=(0.218377, 0.0146117, 1.15558), cameraTarget=(0.218377, 0.0146117, 0))

B.1.8. Get_data_template.py

This script is used for opening the ABAQUS results .odb file and exporting the results to .txt files for data processing in MATLAB. The script below is a specific example for a single run, but can be edited for each individual simulation.

```python
# -*- coding: mbcs -*-
#
# Abaqus/CAE Release 6.9-1 replay file
# Internal Version: 2009.04.16-15.31.05 92314
# Run by Jesse on Tue Nov 30 15:21:46 2010
#
# from driverUtils import executeOnCaeGraphicsStartup
# executeOnCaeGraphicsStartup()
# Executing "onCaeGraphicsStartup()" in the site directory ...
# from abaqus import *
# from abaqusConstants import *
# from caeModules import *
# from driverUtils import executeOnCaeStartup
# executeOnCaeStartup()
#* KeyError: C:/Users/Jesse/Documents/Classes/Fall 2010/ME 871/Project/MATLAB
#* files/Run00.odb
#* File "Get_data.rpy", line 1, in ?
#*   odb = session.odbs['C:/Users/Jesse/Documents/Classes/Fall 2010/ME
#* 871/Project/MATLAB files/Run00.odb']
#* 1 - session.openOdb{
#*   name='C:/Users/Jesse/Documents/Thesis/Optimization Code & Results/Run53.odb')
#* session.viewports['Viewport: 1'].setValues(displayedObject=1)
#* Model: C:/Users/Jesse/Documents/Classes/Fall 2010/ME 871/Project/MATLAB
#* files/Run00.odb
#* Number of Assemblies: 1
```
#: Number of Assembly instances: 0
#: Number of Part instances: 2
#: Number of Meshes: 2
#: Number of Element Sets: 3
#: Number of Node Sets: 9
#: Number of Steps: 3

odb = session.odbs['C:/Users/Jesse/Documents/Thesis/Optimization Code & Results/Run53.odb']

xy0 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7 in NSET DISTAL NODES',
suppressQuery=True)

xy1 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 8 in NSET DISTAL NODES',
suppressQuery=True)

xy2 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 9 in NSET DISTAL NODES',
suppressQuery=True)

xy3 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 10 in NSET DISTAL NODES',
suppressQuery=True)

xy4 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 31 in NSET DISTAL NODES',
suppressQuery=True)

xy5 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 32 in NSET DISTAL NODES',
suppressQuery=True)

xy6 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 33 in NSET DISTAL NODES',
suppressQuery=True)

xy7 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 34 in NSET DISTAL NODES',
suppressQuery=True)

xy8 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 55 in NSET DISTAL NODES',
suppressQuery=True)

xy9 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 56 in NSET DISTAL NODES',
suppressQuery=True)

xy10 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 57 in NSET DISTAL NODES',
suppressQuery=True)

xy11 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 58 in NSET DISTAL NODES',
suppressQuery=True)

xy12 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 79 in NSET DISTAL NODES',
suppressQuery=True)

xy13 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 80 in NSET DISTAL NODES',
suppressQuery=True)

xy14 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 81 in NSET DISTAL NODES',
suppressQuery=True)

xy15 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 82 in NSET DISTAL NODES',
suppressQuery=True)
xy16 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 115 in NSET DISTAL NODES',
suppressQuery=True)
xy17 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 116 in NSET DISTAL NODES',
suppressQuery=True)
xy18 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 117 in NSET DISTAL NODES',
suppressQuery=True)
xy19 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 118 in NSET DISTAL NODES',
suppressQuery=True)
xy20 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 151 in NSET DISTAL NODES',
suppressQuery=True)
xy21 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 152 in NSET DISTAL NODES',
suppressQuery=True)
xy22 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 153 in NSET DISTAL NODES',
suppressQuery=True)
xy23 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 154 in NSET DISTAL NODES',
suppressQuery=True)
xy24 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 199 in NSET DISTAL NODES',
suppressQuery=True)
xy25 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 200 in NSET DISTAL NODES',
suppressQuery=True)
xy26 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 201 in NSET DISTAL NODES',
suppressQuery=True)
xy27 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 202 in NSET DISTAL NODES',
suppressQuery=True)
xy28 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 247 in NSET DISTAL NODES',
suppressQuery=True)
xy29 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 248 in NSET DISTAL NODES',
suppressQuery=True)
xy30 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 249 in NSET DISTAL NODES',
suppressQuery=True)
xy31 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 250 in NSET DISTAL NODES',
suppressQuery=True)
xy32 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 307 in NSET DISTAL NODES',
    suppressQuery=True)
xy33 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 308 in NSET DISTAL NODES',
    suppressQuery=True)
xy34 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 309 in NSET DISTAL NODES',
    suppressQuery=True)
xy35 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 310 in NSET DISTAL NODES',
    suppressQuery=True)
xy36 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 367 in NSET DISTAL NODES',
    suppressQuery=True)
xy37 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 368 in NSET DISTAL NODES',
    suppressQuery=True)
xy38 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 369 in NSET DISTAL NODES',
    suppressQuery=True)
xy39 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 370 in NSET DISTAL NODES',
    suppressQuery=True)
xy40 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 439 in NSET DISTAL NODES',
    suppressQuery=True)
xy41 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 440 in NSET DISTAL NODES',
    suppressQuery=True)
xy42 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 441 in NSET DISTAL NODES',
    suppressQuery=True)
xy43 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 442 in NSET DISTAL NODES',
    suppressQuery=True)
xy44 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6565 in NSET DISTAL NODES',
    suppressQuery=True)
xy45 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6567 in NSET DISTAL NODES',
    suppressQuery=True)
xy46 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6568 in NSET DISTAL NODES',
    suppressQuery=True)
xy47 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6569 in NSET DISTAL NODES',
    suppressQuery=True)
xy48 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6601 in NSET DISTAL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6783 in NSET DISTAL NODES',
suppressQuery=True)
xy66 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6784 in NSET DISTAL NODES',
suppressQuery=True)
xy67 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6785 in NSET DISTAL NODES',
suppressQuery=True)
xy68 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6853 in NSET DISTAL NODES',
suppressQuery=True)
xy69 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6855 in NSET DISTAL NODES',
suppressQuery=True)
xy70 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6856 in NSET DISTAL NODES',
suppressQuery=True)
xy71 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6857 in NSET DISTAL NODES',
suppressQuery=True)
xy72 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6925 in NSET DISTAL NODES',
suppressQuery=True)
xy73 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6927 in NSET DISTAL NODES',
suppressQuery=True)
xy74 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6928 in NSET DISTAL NODES',
suppressQuery=True)
xy75 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 6929 in NSET DISTAL NODES',
suppressQuery=True)
xy76 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7015 in NSET DISTAL NODES',
suppressQuery=True)
xy77 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7017 in NSET DISTAL NODES',
suppressQuery=True)
xy78 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7018 in NSET DISTAL NODES',
suppressQuery=True)
xy79 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7019 in NSET DISTAL NODES',
suppressQuery=True)
xy80 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7105 in NSET DISTAL NODES',
suppressQuery=True)
xy81 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7107 in NSET DISTAL NODES',
suppressQuery=True)
xy82 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7108 in NSET DISTAL NODES', suppressQuery=True)
xy83 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7109 in NSET DISTAL NODES', suppressQuery=True)
xy84 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7213 in NSET DISTAL NODES', suppressQuery=True)
xy85 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7215 in NSET DISTAL NODES', suppressQuery=True)
xy86 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7216 in NSET DISTAL NODES', suppressQuery=True)
xy87 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 7217 in NSET DISTAL NODES', suppressQuery=True)
xy88 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19480 in NSET DISTAL NODES', suppressQuery=True)
xy89 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19492 in NSET DISTAL NODES', suppressQuery=True)
xy90 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19504 in NSET DISTAL NODES', suppressQuery=True)
xy91 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19516 in NSET DISTAL NODES', suppressQuery=True)
xy92 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19534 in NSET DISTAL NODES', suppressQuery=True)
xy93 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19552 in NSET DISTAL NODES', suppressQuery=True)
xy94 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19576 in NSET DISTAL NODES', suppressQuery=True)
xy95 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19600 in NSET DISTAL NODES', suppressQuery=True)
xy96 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19630 in NSET DISTAL NODES', suppressQuery=True)
xy97 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19660 in NSET DISTAL NODES', suppressQuery=True)
xy98 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: HONEYCOMB-1 Node 19696 in NSET DISTAL NODES', suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 64 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy10 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 65 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy11 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 66 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy12 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 67 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy13 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 68 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy14 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 69 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy15 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 70 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy16 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 71 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy17 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 72 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy18 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 73 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy19 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 74 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy20 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 75 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy21 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 76 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy22 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 77 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy23 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 78 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy24 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 79 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy25 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 80 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy26 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 81 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy27 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 82 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy28 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 83 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy29 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 84 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy30 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 85 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy31 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 86 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy32 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 87 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy33 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 88 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy34 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 89 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy35 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 90 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy36 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 91 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy37 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 92 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy38 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 93 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy39 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 94 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy40 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 95 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy41 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 96 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy42 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 97 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
suppressQuery=True)
xy43 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 98 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy44 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 99 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy45 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 100 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy46 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 101 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy47 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 102 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy48 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 103 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy49 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 104 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy50 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 105 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy51 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 106 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy52 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 107 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy53 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 108 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy54 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 109 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy55 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 110 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy56 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 111 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy57 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 112 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy58 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 113 in NSET PROXIMAL
    STEEL NODES',
    suppressQuery=True)
xy59 = xyPlot.XYDataFromHistory(odb=odb,
outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 162 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy60 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 163 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy61 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 164 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy62 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 165 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy63 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 166 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy64 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 167 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy65 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 168 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy66 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 169 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy67 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 170 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy68 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 171 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy69 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 172 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy70 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 173 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy71 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 174 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy72 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 175 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy73 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 176 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy74 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 177 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy75 = xyPlot.XYDataFromHistory(odb=odb, outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 178 in NSET PROXIMAL STEEL NODES', suppressQuery=True)
xy76 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 179 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy77 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 180 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy78 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 181 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy79 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 182 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy80 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 183 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy81 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 184 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy82 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 185 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy83 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 186 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy84 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 187 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy85 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 188 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy86 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 189 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy87 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 190 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy88 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 191 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy89 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 192 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy90 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 193 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy91 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 194 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
xy92 = xyPlot.XYDataFromHistory(odb=odb,
    outputVariableName='Reaction force: RF1 PI: STEEL BAR-1 Node 195 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
suppressQuery=True)
xy93 = xyPlot.XYDataFromHistory(odb=odb,
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 196 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 197 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 198 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 199 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 200 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 201 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 202 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 203 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 204 in NSET PROXIMAL STEEL NODES',
    suppressQuery=True)
outputVariableName='Reaction force: RF1 PI: STEEL BAR
    - 1 Node 5 in NSET PROXIMAL STEEL NODES, Reaction force: RF1 PI: STEEL BAR-1 Node 6 in NSET PROXIMAL STEEL NODES,

138
c1 = session.Curve(xyData=xy_result)
xyp = session.xyPlots['XYPlot-1']
chartName = xyp.charts.keys()[0]
chart = xyp.charts[chartName]
chart.setValues(curvesToPlot=(c1, ))
x0 = session.xyDataObjects['Proximal']
session.writeXYReport(fileName='Proximal.dat', appendMode=OFF, xyData=(x0, ))
x0 = session.xyDataObjects['Distal']
session.writeXYReport(fileName='Distal.dat', xyData=(x0, ))