MICRO-CT IMAGE-BASED MESH GENERATION AND FINITE ELEMENT ANALYSIS OF BONE

A Thesis

by

ZACHARY LEE KOHN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Harry A. Hogan
Committee Members,	Susan A. Bloomfield
	Anastasia Hanifah Muliana
Head of Department,	Andreas Polycarpou

May 2020

Major Subject: Mechanical Engineering

ABSTRACT

This study investigated the reduced platen compression (RPC) test of the distal femur metaphysis (DFM) of adult male rats. The objectives were to develop a finite element (FE) mesh generation algorithm that would take existing μCT images and create a model for use in commercial finite element analysis (FEA) software, and to investigate the RPC test using different boundary conditions (BC) for comparison with modern bone FEA studies.

The MATLAB Image Processing Toolbox was used to manipulate the μCT images from their exported format into a usable volume in the form of a 3D binary image. The voxels in this image were directly converted to hexahedral elements in a FE mesh using custom developed MATLAB scripts. The meshes were imported to ABAQUS for FE processing. Post-processing was done in both ABAQUS and MATLAB. ABAQUS was used to visualize the results in contour plots. The stress and strain values for each element were exported for distribution analysis in MATLAB.

To validate the FEA techniques developed in this study, two animals were chosen based on their RPC mechanical testing results; one with significantly above average and one with significantly below average maximum load. Both whole specimen compression (WSC) and RPC tests were simulated using rough and smooth BCs. The reaction force results of the simulations mirrored past mechanical testing, but the outlier effect was muted due to greater consistency in platen sizing in the simulations than in RPC testing. The contour plots and strain distributions indicated that the RPC test has little influence from the cortical tissue, supporting the assumption that the RPC test effectively measures cancellous tissue properties.

The computational requirements of the image processing and mesh generation techniques, along with the FE processing in ABAQUS are analyzed for future study feasibility. Computation time is not a pressing issue, but memory becomes the limiting factor for both simulation processing and result storage.

Overall, the mesh generation technique developed in this study is powerful due to its generality; any binary image can be converted in a FE mesh. Future possibilities include extending the analysis to nonlinear FEA and incorporating tissue failure models.

ACKNOWLEDGEMENTS

This work may be my own, but I was in no way alone. To the many teachers who led me on this path and the friends I gained along the way, there are not enough pages here to properly express my appreciation. Many helped in this journey, and for all of you I am eternally grateful.

I want to thank Dr. Muliana for serving on my thesis committee. I was fortunate to be a student in her class multiple times, as an undergraduate and graduate student. CVEN 305 in the spring of 2015 was the class that made me sure of my future; I knew I was in the right place. Thank you to Dr. Bloomfield for serving on my thesis committee as well. Her input and tutelage during my graduate study has certainly influenced my work and this project, ensuring that I knew the biology behind the engineering on screen. Her knowledge and passion for science is truly unrivaled. I would like to thank Dr. Hogan for serving as my committee chair and overseeing this work. He has served as a teacher and mentor throughout my time at A&M, both inside and outside the lab. Since we first met in Spain Dr. Hogan has been an unending resource, helping me grow as a student and person. Thank you for all the opportunities you have provided me, both known and unknown.

I want to thank all the students I have worked with in the Bone Biomechanics and Bone Biology labs. There is a unique bond forged while working under the red lights of LARR, and I hope our skills of measuring 15g of food by eye is transferrable.

Finally, I would be remiss not to thank my family. I have enjoyed an endless stream of support from my gracious parents Stacy and Greg, my brother and ever patient roommate Nick, and my wonderful fiancée Angela, even as graduation dates came and went. I know we are all looking forward to what the future has in store.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a thesis committee consisting of Professor Harry Hogan and Professor Anastasia Muliana of the Department of Mechanical Engineering and Professor Susan Bloomfield of the Department of Health and Kinesiology.

The CT scan data used for this analysis was provided by Professor Larry Suva of the Department of Veterinary Physiology and Pharmacology whose students were responsible for the scans.

Mechanical testing data - used for comparison to the simulations in this study - was collected in previous studies by Jenny Kosniewski, Jeremy Black, Jessica Brezicha, Jon Paul Elizondo, and Scott Lenfest.

All other work for conducted for this thesis was completed by the student independently.

Portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

Funding Sources

Graduate study was supported in part by a Teaching Assistantship from the J. Mike Walker ’66 Department of Mechanical Engineering at Texas A&M University.

NOMENCLATURE

AMPS	Absolute Maximum Principal Strain
BC	Boundary Condition
BMD	Bone Mineral Density
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CDF	Cumulative Density Function
CPU	Central Processing Unit
CSV	Comma Separated Value
CT	Computed Tomography
DFM	Distal Femur Metaphysis
ECA	Element Connectivity Array
FE	Finite Element
FEA	Finite Element Analysis
FEM	Finite Element Modelling/Method
GPU	Graphical Processing Unit
GUI	Graphical User Interface
HPRC	High-Performance Research Computing
HU	Hindlimb Unloading
INP	Input File (ABAQUS Filetype)
μCT	Micro-Computed Tomography
NCA	Nodal Coordinate Array
ODB	Output Database (ABAQUS Filetype)
pQCT	Peripheral Quantitative Computed Tomography
PTM	Proximal Tibia Metaphysis
RAM	Random Access Memory
RF	Reaction Force
RPC	Reduced Platen Compression
RR	RPC Rough
RS	RPC Smooth
VOI	Volume of Interest
WR	Whole Rough
WS	Whole Smooth
WSC	Whole Specimen Compression
3PB	Three Point Bending

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

More than 50 million men and women in the United States suffer from osteoporosis or low bone density. Almost half of Americans over 50 are at risk of fracture. With an aging population, the healthcare system is estimated to see an economic impact of more than $25 billion per year in the coming decade [1].

While some humans are aging on earth, an increasing number are spending time far from our planet. Exploration of ever more extreme environments has highlighted the limits of our skeletal system. In the early years of spaceflight it became apparent that human bone experienced changes while in the gravitational unloading and radiation environment of low earth orbit. Astronauts and Cosmonauts alike saw decreases in bone mineral density (BMD) when they returned to earth [2]. Successful countermeasures for this bone loss have been deployed on modern missions to the International Space Station, namely resistance exercise and antiresorptive medication [3]. With Mars on the horizon, however, future missions will be of longer duration and take place in partial gravity environments where the previous countermeasures may not be feasible [4].

In the last decade, enormous strides have been made in both the speed of computing and the quality of medical imaging. The confluence of these two subjects has allowed increasingly complex analysis of the human body and how it reacts with its environment. Finite element analysis (FEA) of bone tissue has benefitted uniquely from these technological advances.

In the early 1970s, the newly developed finite element method (FEM) used in structural analysis and civil engineering made its way into the world of biomechanics. Bone researchers quickly realized the potential FEA had to a field where the structures of interest were highly complex geometrically and made up of multiple materials. Precise bone geometry could be

modeled, and stress fields could be determined for simulated load scenarios. Early models were very coarse and limited in their formulation due to the amount of computation involved. By the mid-1980s, "refined" models with high density meshes were on the order of 10,000 degrees of freedom (DOF) [5]. Computer processing speed has become many thousands of times faster in the decades since these early models were created [6]. Today, models of similar complexity take seconds to calculate on a personal computer.

As the capabilities of FEA have advanced so, too, have the ambitions of its applications. micro-computed tomography (μCT) can generate discrete models of bone sections with millions of elements. However, μCT data is widely used solely for densitometric analysis and structural characterization. The potential for the existing scans to be used in FEA is often overlooked.

In 1995, van Rietbergen introduced the idea of using direct voxel conversion from CT images for micromechanical models of trabecular bone [7]. This discrete modelling is limited to regions where high resolution CT images can be acquired. Popular today are CT-informed continuum models that assign material properties to elements based on apparent density in the CT scans [8]. These models do not fully resolve the trabecular structure, allowing for lower cost scans. However, these continuum models are limited to specimen where the scale of the loading and stress field is sufficiently larger than the underlying trabecular structure, satisfying the assumption that the cancellous bone can be treated as a continuum [9]. For sites where this scale is on the borderline, such as the rat distal femur metaphysis (DFM) in the present study, discrete trabecular models offer more accuracy.

FEA is fully reliant on its input data. Thus, mechanical testing is still required to determine the tissue properties to use in a model. Many methods are presently used to find these properties for bone tissue, from nanoindentation to atomic force microscopy [10, 11]. Larger

samples are tested in compression and tension to determine structural and whole bone properties. These experiments show a wide range of values for the mechanical properties of bone in different species and in different locations throughout the body [12].

The reduced platen compression (RPC) test has been devised to assess the strength of cancellous tissue without physically removing the cortex [13]. Previous studies have optimized the RPC process through image analysis and with the use of continuum FEA [14]. The present study develops a method for discrete trabecular structure FEA to further model and study the RPC test.

Complicating the process of FEA using μCT data is the need to transform the output from the CT scan to a format usable in FEA packages. Most modern scanners come with attached software that allows for this transition, with some even performing FEA using their own solvers and post-processors. These solvers are often very limited in the loads that can be applied and the degree to which the results can be analyzed. In the case of the Scanco μCT 50 (Scanco Medical, AG, Switzerland) used in the present study, the FEA capabilities are limited to linear analysis [15]. In order to model non-standard problems like the RPC test and to have capabilities that extend beyond linear analysis, an alternative solution method must be used. Commercial solvers like ABAQUS and ANSYS are widely available and have robust FEA capabilities. ABAQUS is used in the present study because of its availability to users at Texas A&M University.

The primary complication in using these commercial FEA packages is having a 3D part or volume with which a mesh can be generated. Many commercial CAD packages support 3D part files that are easily imported to ABAQUS. This is not the case with volumes generated by CT. Common software packages used for the visualization and manipulation of CT images in the medical community are often freely available. These packages allow for segmentation of CT

images and visualization of volumes, but their export capabilities are limited to STL formats or other surface models. These surfaces could then be filled using other programs, but the workflow for FEA using this method becomes cumbersome and ineffective, as the fidelity of the models deteriorates with each format conversion. Commercial software packages have been developed to bridge this gap (e.g., ScanIP/FE, Materialise Mimics, etc.), but they are extremely cost prohibitive; licenses will run into the low six figures. Laboratories that use CT-based FEA often employ custom-developed codes to generate meshes [16]. The practice of direct voxel conversion from CT data to a input file for ABAQUS has been previously demonstrated in Python [17]. MATLAB will be used in the present study for image analysis and mesh generation, as it is a cost-effective program and freely available at many universities. Making the algorithms and source code developed herein publicly available will allow those without the resources to acquire expensive CT manipulation software to extend their capabilities for CT-based FEA.

1.1 Objectives

There will be two focus areas for this thesis: developing a method for processing μCT images into a finite element (FE) mesh; and investigating the FEA parameters that may benefit future studies involving the RPC test. Prior studies have investigated the RPC test using continuum models of cancellous tissue. The present study will generate discrete recreations of the trabecular structure, increasing the size and computational cost of the analysis by several orders of magnitude. The objective of this study may be summarized as follows:

1. To develop a mesh generation algorithm using readily available software that:
    a. Uses μCT images as an input.
    b. Outputs a mesh for use in commercial FEA software.
2. To understand the effects on simulated whole specimen compression (WSC) and reduced platen compression (RPC) tests of common FEA parameters including:
    a. Different loading and boundary condition cases.
    b. Different volume geometry cases (i.e., with or without the cortical shell).
3. To investigate the computational feasibility of using the FE models in larger studies.

2. BACKGROUND

2.1 Bone Biology and Structure

The skeleton is a critical organ and the bones in it play a multitude of roles. Mechanically, they support the body and transmit loads from the environment and from within the body. Bones act as levers, amplifying the forces generated by muscles. This minimizes the length over which muscles must contract to move the body around or to manipulate the world [18]. Bones also protect more sensitive internal organs such as the ribs around the lungs and heart. Biologically, bone acts as a bank for minerals like calcium and phosphate to be deposited or released for the body to maintain long term homeostasis. Some bone sites even form red blood cells [19].

Bone tissue is primarily comprised of inorganic minerals, collagen, and water. The collagen molecules group together to form fibrils, which cross-link with one another. Mineral crystals, specifically hydroxyapatite, fill the space between and around the cross-linked collagen fibrils, forming sheets known as lamellae. In the long bones of humans, these lamellae organize into concentric annular cylinders called osteons. These osteons house canals for blood flow and nervous tissue. The organization of the microscale structure within bone is shown in Figure 1.

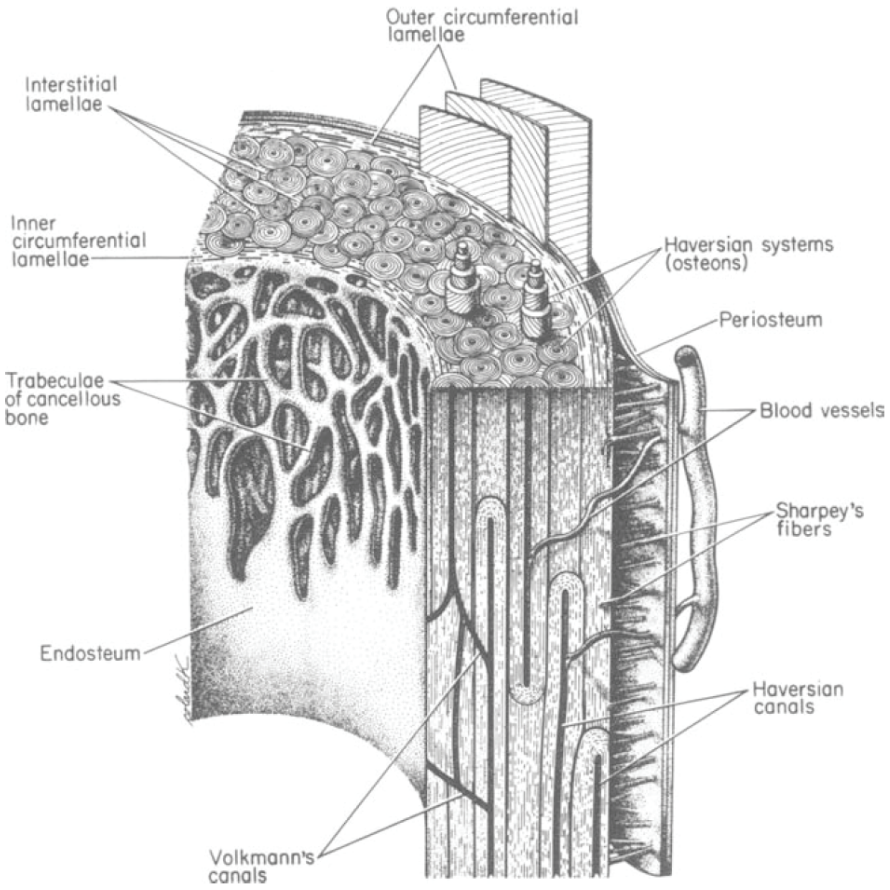


Figure 1. Important Features of Human Long Bones. Reprinted From [18].

From a macroscale perspective, bone has two main structural components. The dense outer shell is known as compact or cortical bone. The less dense internal bone known as spongy or cancellous bone is found in the ends of long bones, as well as other locations such as the vertebrae and ribs. The locations of each bone type in long bones can be seen in Figure 2.

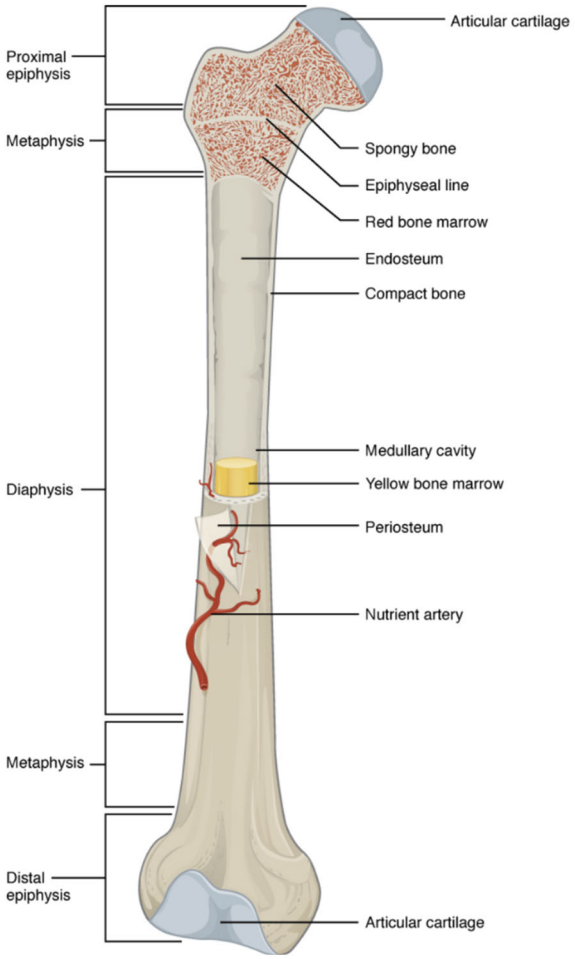


Figure 2. Structural Anatomy of Human Femur. Reprinted From [20].

Cancellous bone is comprised of rods and plates on the order of 200 μm thick, referred to as trabeculae, only filling approximately 30% of the bone cavity. The remaining volume is occupied by marrow [19]. The terms cancellous and trabecular bone are often used interchangeably. Herein, cancellous bone will be used to describe the macroscale spongy bone and its structural properties, and trabecular bone will refer to the individual rods and plates and their characteristics.

The primary mechanical role of cancellous bone is to absorb loads and transmit them to the cortical shell. This explains why cancellous bone is located at the ends of long bones where the loads encountered during motion are applied. The trabecular structure distributes these surface loads to a larger volume of material, reducing the maximum stresses encountered. Due to this load transmission, the trabecular structure is often found aligned with the stresses experienced in loading [21]. The diaphysis of long bones (Figure 2) is void of cancellous bone as the loads experienced in this region are more evenly distributed axial, bending, and torsional loads for which the hollow cross section of cortical bone is well adapted to handle.

Bone can grow and adapt in response to both physiological and mechanical stimuli. The factors that influence these behaviors and their interactions are complex and this area is still undergoing active research. The idea of a relationship between bone's structure and function was introduced in 1892 in the form of "Wolff's Law". This law stated simply that changes in the function of bone was followed by a change in its architecture, subject to mathematical laws. Much of Wolff's interpretation has since been refuted, both the merit in bearing his name and its status as a law in the first place, but the term is still widely used to mean the functional adaption bone displays in remodeling [22]. Little attention was given to the mechanical behavior of bones until the 1950s when experiments began to show possible ways in which bone could relay signals in response to mechanical stimulus, for instance the discovery of bone having piezoelectric properties [23]. In the 1990s a model known as the Mechanostat was proposed for bone adaptation based on the mechanical strains the tissue experiences during loading [24]. Crucial to this idea is the role of muscle strength in skeletal development, as it provides the growth inducing mechanical strains. In short, the model suggests there exists a critical level of strain that once reached, signals the remodeling process. If the strain remains inside some range

of values encountered during normal loading, the shape would remain the same. These thresholds of strain are illustrated in Figure 3.

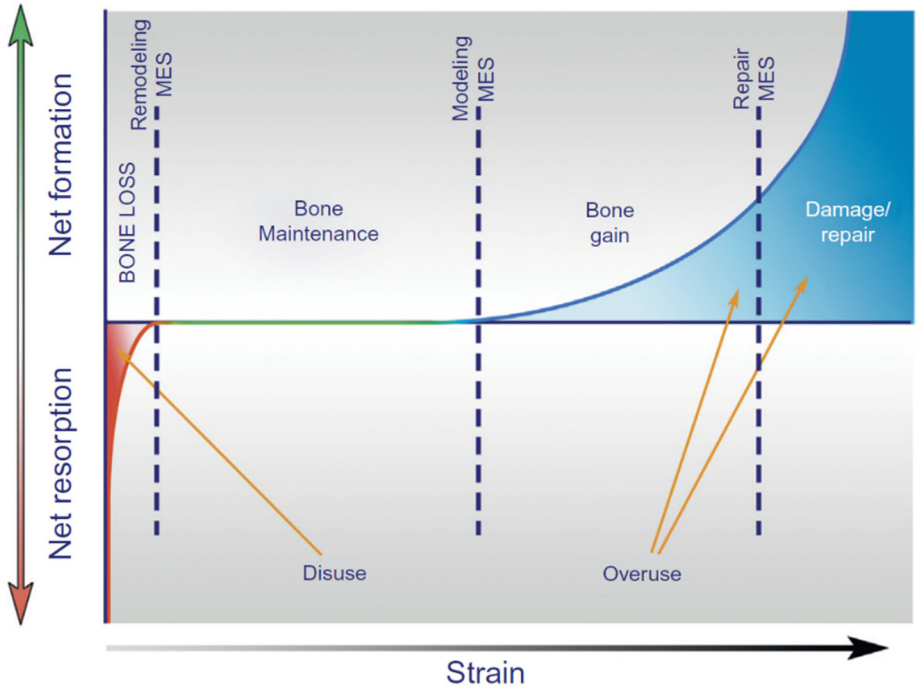


Figure 3. Mechanostat Strains.

The Mechanostat model has been supported in multitudes of experimental data [25], and is more recently being investigated on a molecular and chemical level to identify the signal processes involved. The OPG/RANKL/RANK proteinic system has be shown to react to tensile and compressive strains by changing concentrations of the proteins involved, which would regulate the metabolism of bone tissue in a manner consistent with the Mechanostat model [26].

2.2 Mechanical Properties of Bone

Bone's complex microstructure leads not only to its unique geometry, but to complex material properties as well. Bone acts as a composite material, whose elastic properties and failure behavior are functions of its constituent components. The hydroxyapatite crystals are very strong and stiff. Collagen is comparatively quite soft and ductile. The stiffness of bone lies between that of the mineral and collagen, as expected for a composite material. The benefit of this arrangement is apparent the overall strength of bone, as the composite structure is stronger than its individual components. The collagen fibrils stop cracks from propagating through the mineral preventing brittle fracture. Conversely, the mineral crystals bear load and prevent the collagen from yielding [27]. In this way, the mineral is providing the elastic behavior of the material, and the collagen is driving the post-yield and energy absorption properties.

A delineation needs to be made between the internal tissue level and the average structural properties of bone. The difference between the two is most pressing for cancellous bone. When looking at the material properties of individual trabeculae, there is little difference from cortical bone. If the overall structure of the cancellous framework is considered, bulk properties become of interest. Cancellous bone has been shown to be 35% less stiff than cortical bone [12]. While the material level behavior is certainly important, the structural behavior has more influence on the load carrying capabilities of the full bone.

2.3 Mechanical Testing of Bone

Destructive mechanical testing allows for a material and its failure mechanisms to be classified. Carefully controlling the geometry of the specimen and the loads to which it is subject allows various material constants to be calculated. These constants can then be used to predict the behavior of the material when subject to additional loads or estimate the response to a problem that cannot be directly tested. Tensile and compressive testing are examples of these types of experiments.

A brief note of importance is the difference between intrinsic and extrinsic properties that can be determined from mechanical testing. Direct quantities measured in a tensile test, for instance, are the reaction force of the specimen and the displacement to which it is subjected. A larger specimen (in width) will generate more force if it experiences the same axial displacement as a smaller specimen. When normalized by the geometry of the specimen, the force and displacement become stress and strain, respectively. These normalized intrinsic properties allow for comparison between specimens and of different materials. The extrinsic properties are not without application, however. In the case of whole bones, it is useful to know the overall force that can be sustained as this is more relevant to the life of the animal (or human) to whom the bone belongs.

The results of a typical tensile test are shown in Figure 4. The red line indicates the linear-elastic portion of the deformation. Next, a critical point is reached where yield occurs, and the material then undergoes plastic deformation. The highest load which the specimen bears throughout the test is marked as the maximum load, and the point at which fracture occurs is shown.

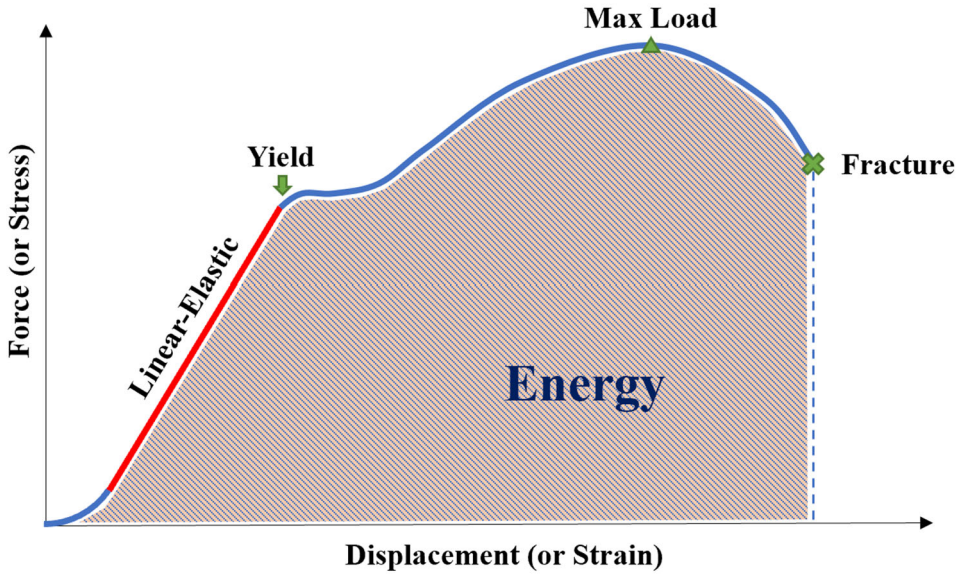


Figure 4. Example Stress-Strain Curve.
Energy is the area beneath the curve. If force and displacement are normalized for specimen geometry, they produce stress and strain, respectively.

The linear-elastic region is where the characteristic material constants can be determined. For instance, the slope of the linear portion of the curve is the stiffness of the specimen. When normalized for geometry, the intrinsic stiffness of the material is known and its Young's Modulus (E). The area beneath the load-displacement curve represents the amount of energy absorbed by the material before fracture, or put differently, the amount of work done on the material that caused it to fail. This is often referred to as the material's toughness.

2.3.1 RPC Testing

Most mechanical properties of bone, and many other materials for that matter, are acquired by testing specific sections in tension. However, the primary loading for most bones occurs in compression or in bending. The failure modes for these load cases have been shown to differ [28]. In homogenous isotropic materials like metal, the mechanical data found in tensile tests can be used to accurately predict behavior in other load cases. This extension is not as simple for complex and composite materials like bone. For this reason, numerous experiments have been used to determine the response of bone to different load cases [29].

Complicating further the matter of direct mechanical testing is the difficult nature of working with cancellous bone. The structure of cancellous tissue becomes an issue when attempting to isolate it from the surrounding cortical material. With tedious care taken in generating test specimens, tensile tests have shown that the elastic behavior of cancellous tissue is similar in compression and tension up to the point of yielding, after which the failure mechanisms differ and produce different stiffening behavior [30]. This data does have limitations since the experiments have removed the trabecular architecture from the cortex. The outer layer of trabeculae has been disconnected from the surrounding structure, so surfaces that would have carried load *in situ* are now free, creating a boundary layer effect on these test specimens. While this data is certainly useful in understanding the material behavior of cancellous tissue, it is an ideal loading situation and does not fully represent the *in vivo* loading scenarios and interactions that cancellous bone would experience.

The interaction between cancellous and cortical tissue has been investigated using whole specimen compression. This involves taking a cross section of a bone and compressing it

between two platens. In this case, both the cortical shell and cancellous core are loaded, so the interaction between them is included in the test.

A more sensitive compression test has been developed where only a reduced section of the cancellous core is loaded, known as the reduced platen compression (RPC) test. In whole section compression, the cortical shell bears most of the applied load. This means that for sections where the strength of the cancellous core is significantly different between specimens, that difference is muted by the cortical shell. The RPC method loads only the cancellous section, while maintaining its interaction with the cortical shell. The RPC test has been shown to be twice as sensitive to changes in cancellous strength than whole section compression for similar specimens [13]. The difference in loading between the RPC and whole specimen method is shown in Figure 5.

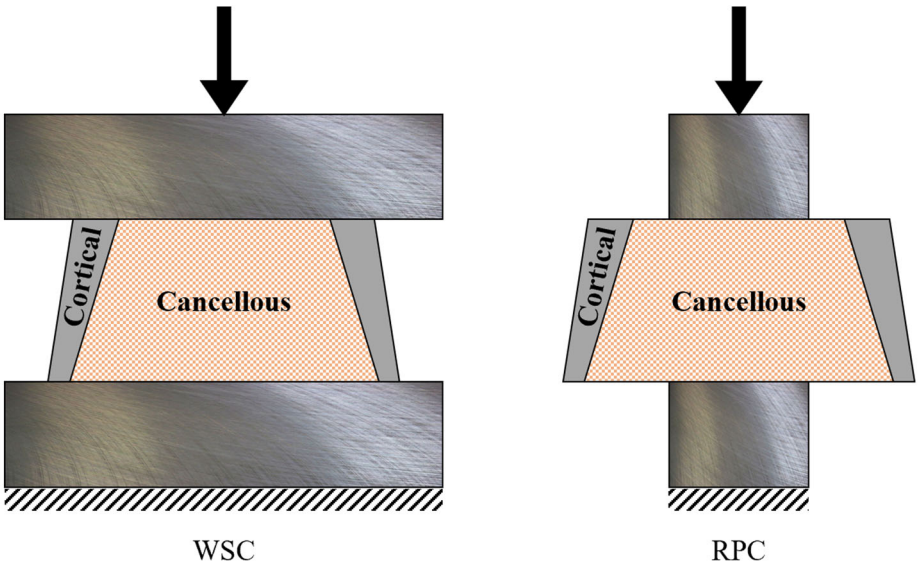


Figure 5. Whole Specimen Compression (WSC) and Reduced Platen Compression (RPC) Test.
Bone illustrated is a side view of a sample cross section of a femur or tibia metaphysis that has been removed from the bone. In the case of the distal femur metaphysis in the present study, the top surface here would be the most proximal face.

Figure 6 provides a 3D view of the RPC test, illustrating how the platens contact the cancellous tissue without loading the cortical shell. The contact area for the RPC platens is chosen to be 70% of the diameter of a circle that fits entirely inside the cortical wall of the cut specimen, known as the endocortical circle. The platen sizing is illustrated in Figure 7 for both the distal femur metaphysis (DFM) and proximal tibia metaphysis (PTM). More information on the location and preparation of the specimens is available in previous studies [31, 32].

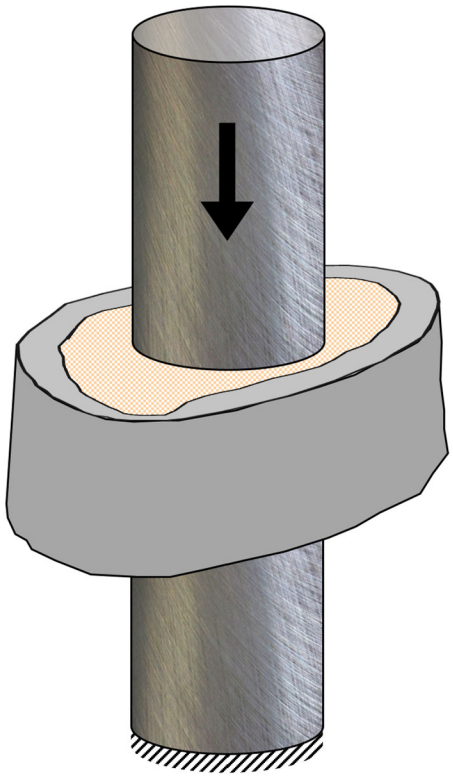


Figure 6. 3D Representation of RPC Test.
Bone illustrated is a 2mm thick section of a rat distal femur metaphysis.

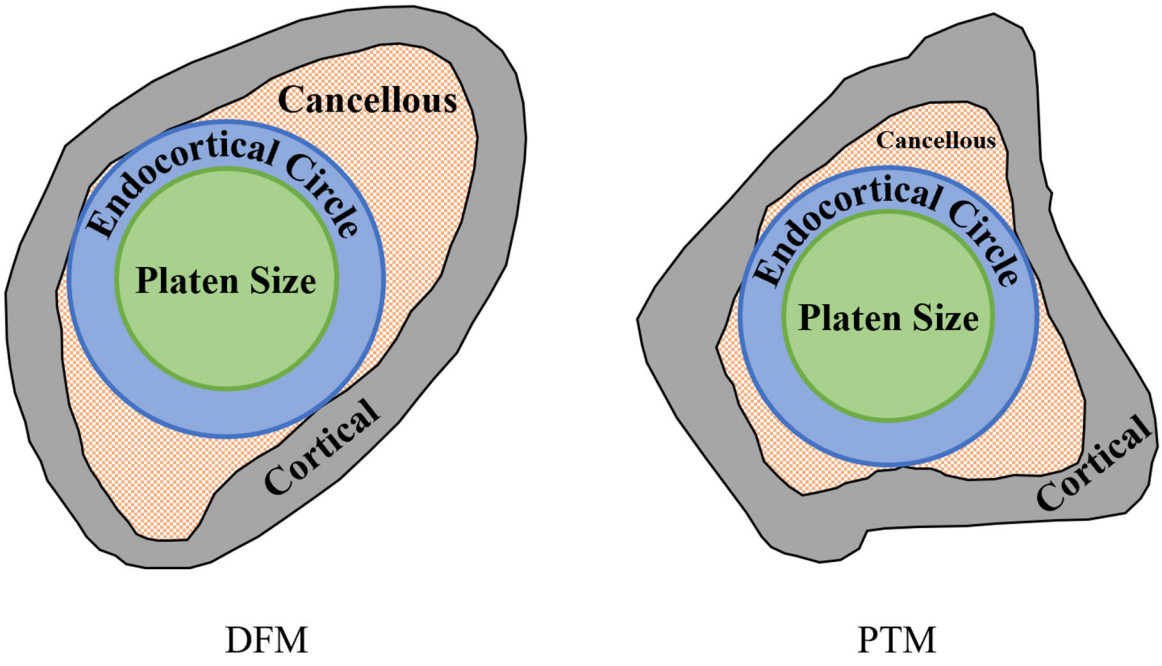


Figure 7. RPC Platen Sizing for Distal Femur Metaphysis (DFM) and Proximal Tibia Metaphysis (PTM). Endocortical circle is defined as the largest circle that will fit entirely inside the cortical shell. Platen size is 70% of the diameter of the endocortical circle.

2.3.2 Three-Point Bending

Other mechanical testing techniques have been used to determine the strength of cortical bone and whole bones that do not remove a specific test section. This produces more accurate *in situ* loading conditions. To investigate cortical tissue strength, three-point bending (3PB) of the mid-diaphysis of the femur and tibia have been employed in past studies. A generic 3PB test setup is shown in Figure 8. More information on the specimen preparation is available in past studies [32].

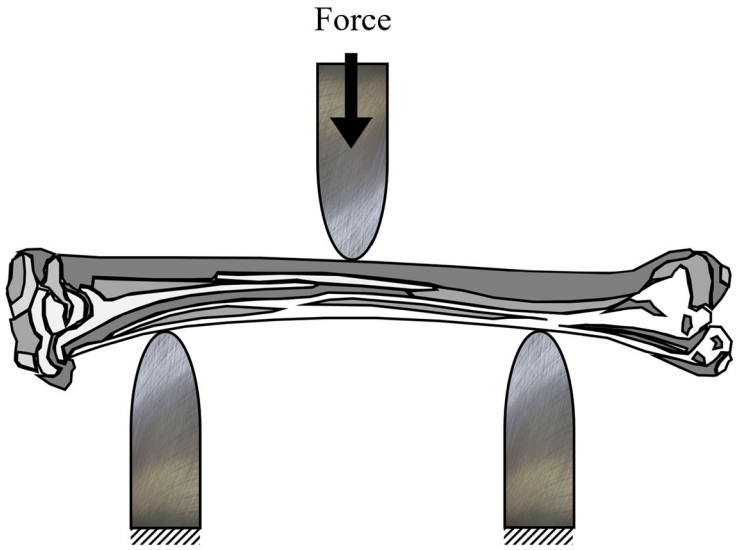


Figure 8. Three-Point Bending (3PB) Diagram.
Rat tibia shown in illustration. 3PB mechanical test also performed on femur.

2.3.3 Femoral Neck Test

Another mechanical test commonly employed to asses whole bone strength is femoral neck fracture. This method attempts to load the head of the femur in a manner consistent with normal use, or in some cases the direction of impact encountered during a fall. The experimental setup for the femoral neck test used in past studies is illustrated in Figure 9. More information on the test conditions and specimen preparation in available in past studies [33].

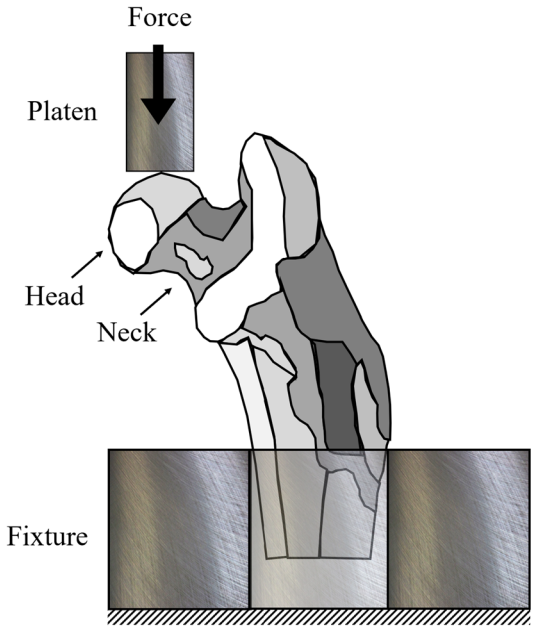


Figure 9. Femoral Neck Test Diagram.
Proximal end of rat femur illustrated in diagram.

2.4 Bone Imaging Using Computed Tomography

Computed tomography (CT) is widely used to generate 2D and 3D images of biological tissue and other materials. The fundamental physics behind CT is the interaction of X-ray radiation with matter. As a photon passes through an object, it interacts with the atoms inside. The photon will collide with other particles and exchange energy with the electrons of the atoms in its path. When the photon reemerges from the object, it will be at a different energy state than when it entered.

CT scanners use this principle of photon-electron interaction to generate images of biological tissue. An X-ray beam packed full of high energy photons is directed at the sample being scanned. These photons exchange energy with the tissues in the sample and then run into the detector on the opposite side. The diagram in Figure 10 illustrates the fundamental setup of a CT scanner.

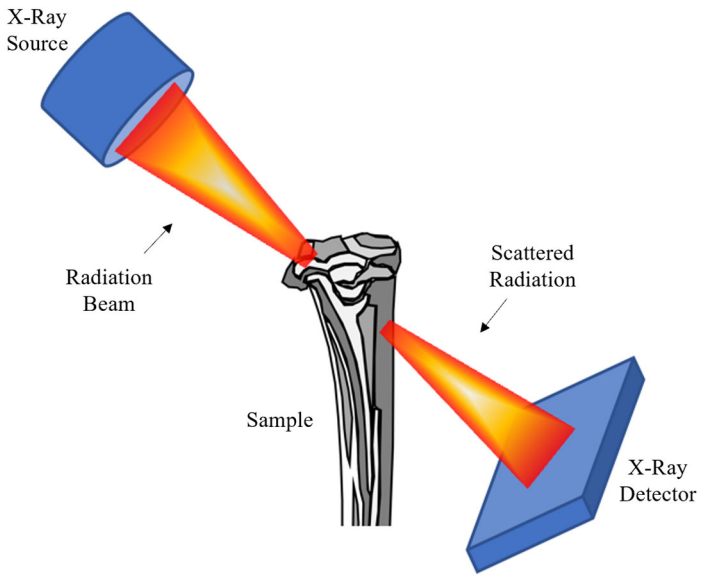


Figure 10. CT Scanner Diagram.

The detector measures the energy level of the photons after passing through the specimen, getting a snapshot of the shadow of the sample from that angle. The source and detector move around the specimen to read the shadows from many angles. Each of these scans is then assembled by a computer to generate a usable image of the sample [34].

2.4.1 pQCT and μCT

CT scans using X-ray radiation are limited in their application by the dose of radiation they give the patient. The higher the resolution or the larger the area of a scan, the more time it takes; therefore, the radiation dose is higher. Thus, scans with high enough resolution to resolve trabecular bone are limited to the extremities: hands, wrists, ankles, etc. These higher resolution scans can be used to generate quantitative measures of bone and other tissues and are consequently name peripheral quantitative computed tomography (pQCT).

Higher resolution scans than pQCT are achievable, but with doses of radiation exceeding that which would be safe for living organisms. These scans can resolve tissue components on the micron level and are thus named micro-CT (μCT). Due to the high radiation dose and compact scanner sizes, μCT is limited to *ex vivo* scans of animal tissue. μCT is also used in other disciplines interested in microscopic structures, such as composites in materials science [35].

2.5 The Finite Element Method

The first requirement in understanding any physical phenomenon is the development of a mathematical model. Assumptions are made about the way a process works and are then used to develop equations that characterize the observed behavior. This allows for the quantification and estimation of processes in nature, which can be validated with experiments. Complications arise when the equations in a model are difficult to solve (differential, integral, simultaneous, etc.) or they are imposed over a complex domain. Throughout history, models had to be abstracted to a point where analytical solutions could be realistically determined. Intricate geometry or nonuniform material makeup often precluded an analytical solution.

The finite element method (FEM) is a numerical method of analysis that allows complex mathematical models to be solved. It has the dual benefits of increasing the complexity of geometry that can be analyzed, as well as reducing the order of equations that need to be solved to gain an accurate solution to a model. The FEM derives its name from the fundamental process involved, subdividing the domain into smaller components. This process is illustrated in Figure 11.

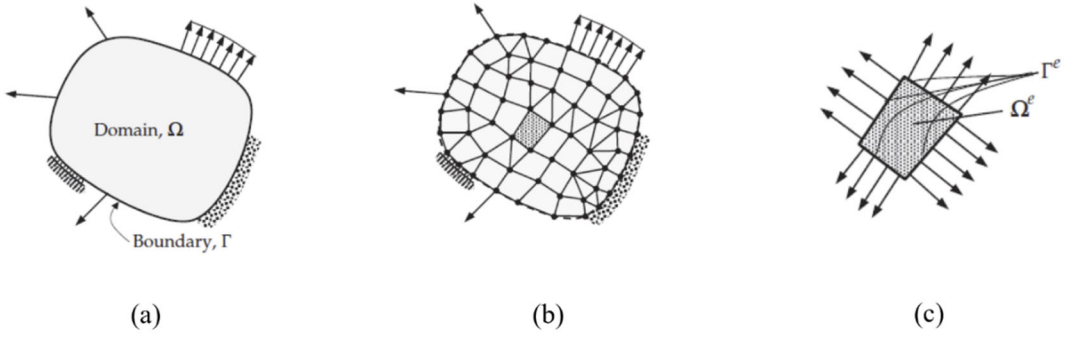


Figure 11. Finite Element Method Discretization. Adapted From [36].
(a) Generalized body with applied loads and boundary conditions, (b) discretized body with elements and nodes, (c) domain and boundary of single element.

Figure 11 (a) shows a generalized domain with various boundary conditions and applied loads. In Figure 11 (b), the domain is discretized into multiple subdomains, known as elements. The elements are connected to one another at points referred to as nodes. These nodes are the points throughout the domain where unknown quantities of interest (e.g., displacement) are calculated. Together, the nodes and elements are called the finite element mesh. Now that the domain has been discretized, the equations of the mathematical model can be solved over the simplified domain, as in Figure 11 (c). Imposing continuity at the nodes (i.e., the calculated values for each node must be the same between elements) and interpolating between these values allows the solution to be approximated for the entire domain.

As with any numerical method, increased discretization allows for higher fidelity representation of higher order nonlinear functions. So too in FEM generating a finer mesh will capture more details. Finer meshes can be used in areas of high gradients to gain a more accurate solution.

The sources for error in a finite element model are threefold: the approximation of the domain, the approximation of the solution, and numerical computation errors (e.g. round off, numerical integration) [36]. The first of these, domain approximation error, is minimized during mesh refinement. Solution approximation is partially addressed in mesh refinement but will often require analysis of the types and formulations of elements used in the model. Computation errors are present in any numerical method.

The errors which have been addressed are those between the finite element solution and the exact solution to a mathematical model. This numerical accuracy is often labelled as the accuracy of a model. The true "Accuracy" however, is how well the model describes the physical behavior of the process it is intended to analyze. A finite element model can be error

free from the exact analytical solution but arrive at meaningless results if the underlying equations are not properly based on a physical understanding of the problem at hand. This is the lens through which any finite element results should be viewed.

2.6 FEM Applied to Bone

In 1995, van Rietbergen introduced the idea of using direct voxel conversion from CT images for micromechanical models of trabecular bone [7]. The elements in that study were 40.26 x 28.44 x 40 μm, in a specimen from the human proximal femur. By 1998, different meshing techniques were being explored, and Ulrich compared both the coarseness of a mesh in relation to the overall structure and the type of mesh used (hexahedral or tetrahedral) [37]. The results of that study showed that hexahedral elements were sufficient for determining bulk properties such as stiffness and strength, particularly when fine enough meshes were used. Tetrahedral meshes can sustain trabecular connections below the resolution of the image used to generate them, an advantage compared to hexahedral meshes, but the algorithms required to generate those meshes are much more complicated.

2.7 Animal Models

Few humans spend time in space; thus, to adequately study the effects of microgravity on bone and potential countermeasures, a ground-based model is necessary. Hindlimb Unloading (HU) in rodents has long been accepted as an analog to the inhibited bone formation observed in humans during spaceflight [38]. Illustrated in Figure 12 is a rat undergoing HU via the tail-traction method.

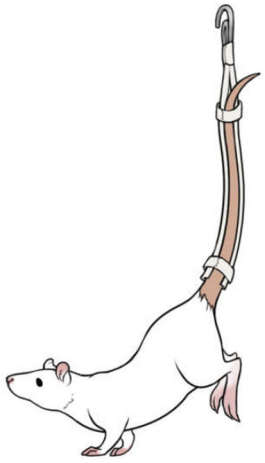


Figure 12. Hindlimb Unloaded Rat

Multiple drug treatments have been investigated as a pretreatment for spaceflight and as osteoporosis medication [3]. The two animals in the present study both received injections of Risedronate prior to HU. More information on the HU protocol and drug pretreatment used on the animals whose bones are used in this thesis is available in previous studies [31, 32].

3. METHODS

3.1 Image Processing

The femurs in this study were previously scanned using a Scanco μCT 50 (Scanco Medical AG, Switzerland) using a 20 μm nominal[1] voxel size (55 kVp, 145 μA, 400 ms, 0.5 mm Al filter). The data from the CT scans was exported in slices as DICOM and TIFF images. DICOM is a standard medical imaging format that includes patient and scan information. TIFF is a more widely used image format that contains only pixel intensity information. The pixel intensity values of the DICOM images were signed 16-bit integers, supporting values from -32768 to 32767. The TIFF image pixel intensities were unsigned 8-bit integers ranging from 0 to 255. In the case of the TIFF images, the CT voxels with the highest bone mineral density (BMD) are mapped to a pixel intensity of 255, the lowest are mapped to 0, and the remainder are interpolated along that scale.

Applications within the MATLAB Image Processing Toolbox were used to identify the slices comprising the Volume of Interest (VOI) for each specimen. The DICOM Browser was used first to import the series of DICOM images from a single folder. The volume was then exported to the Volume Viewer and Video Viewer applications to identify the starting and ending slices of the VOI. The Volume Viewer is used to inspect the whole volume for alignment of the bone and to identify a rough estimate of the starting frame for the VOI.

[1] The CT scan raw data files and DICOM result files contain conflicting information. The CT scan parameters indicate a 19 μm nominal voxel size, which is reported in other publications as 19, 20, or somewhere between (e.g. 19.8 μm) for studies whose scans were generated using the Scanco® μCT 50 – the same model used in the present study. The DICOM files report a pixel size and slice thickness of 0.020 mm. The author theorizes that the DICOM file pixel size and slice thickness fields do not support more than three decimal places. Thus, if a true voxel size of 19.8 μm (or any other non-integer between 19 and 20 μm) was used, the 0.0198 mm value would be rounded to 0.020 mm. This would not significantly alter the methods or results of the present study, as the only effect of changing the user specified voxel width in mesh generation is a uniform dilation of the volume.

Illustrated in Figure 13 is the location of the distal femur metaphysis (DFM) test section used in the present study.

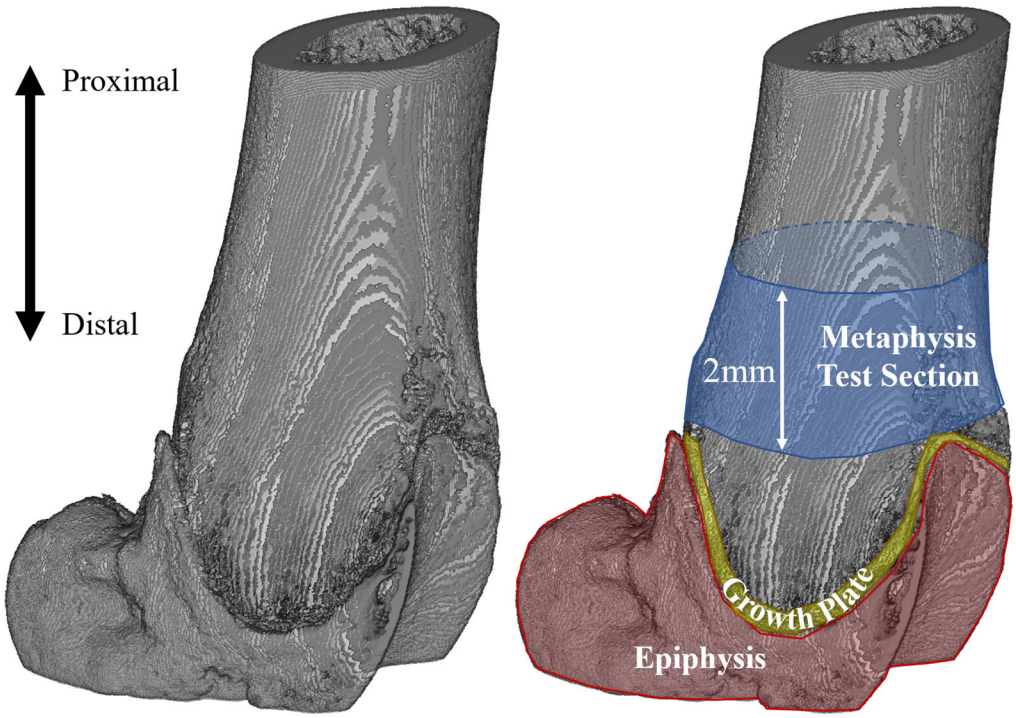


Figure 13. Test Section Location in Rat Distal Femur.
Test section is defined as 2mm thick starting at the most distal point without epiphyseal tissue.

The Video Viewer app is then used to identify the start and end slice for the VOI. The video that is created is composed of each of the DICOM images as a single frame, which can be viewed at a selected frame rate. The video is paused and can be navigated on a frame by frame basis. The most distal (in the case of the femur) slice where the epiphysis is no longer visible is treated as the beginning of the VOI. In Figure 14, the epiphysis can be seen in the first image, and it disappears as the slices progress. Slice 250 would be considered the start of the VOI.

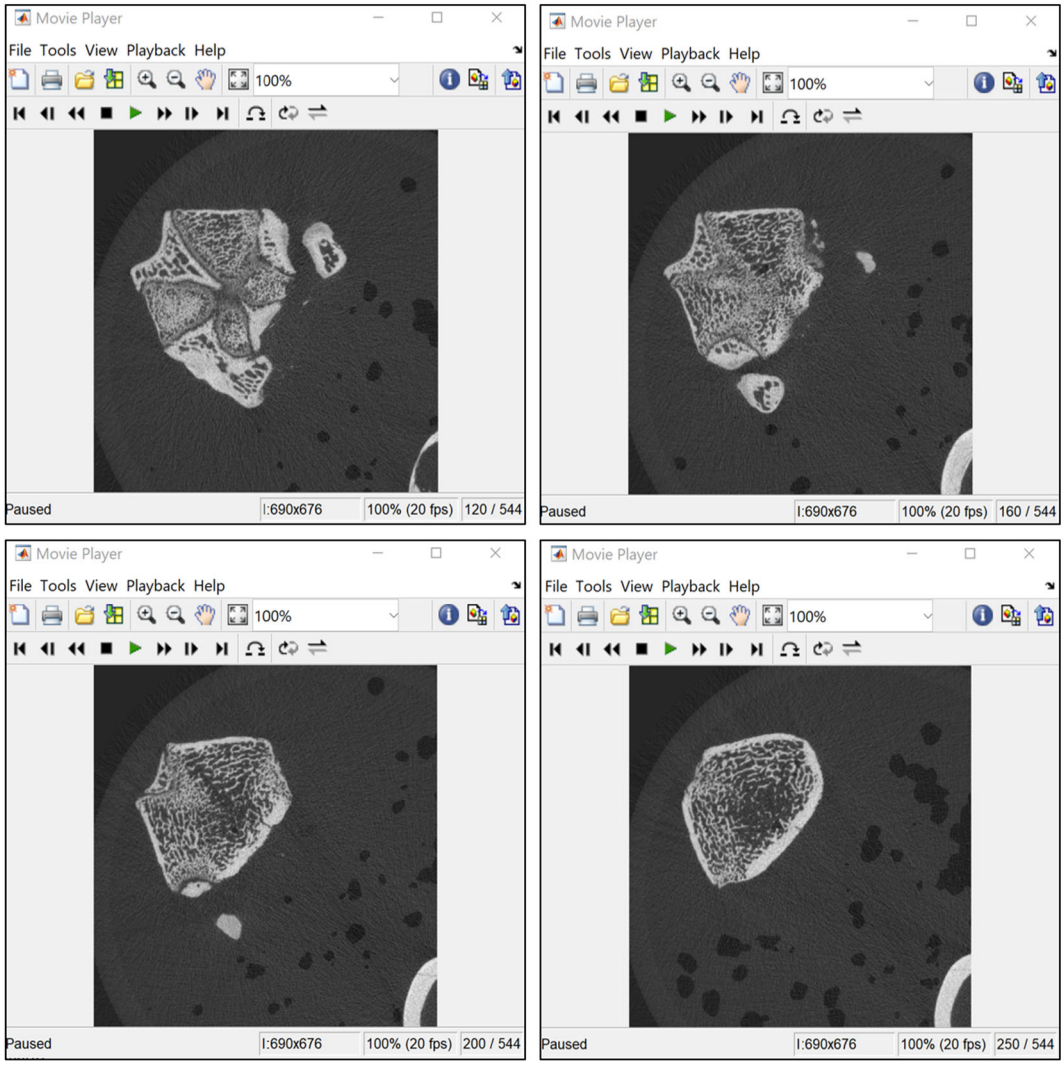
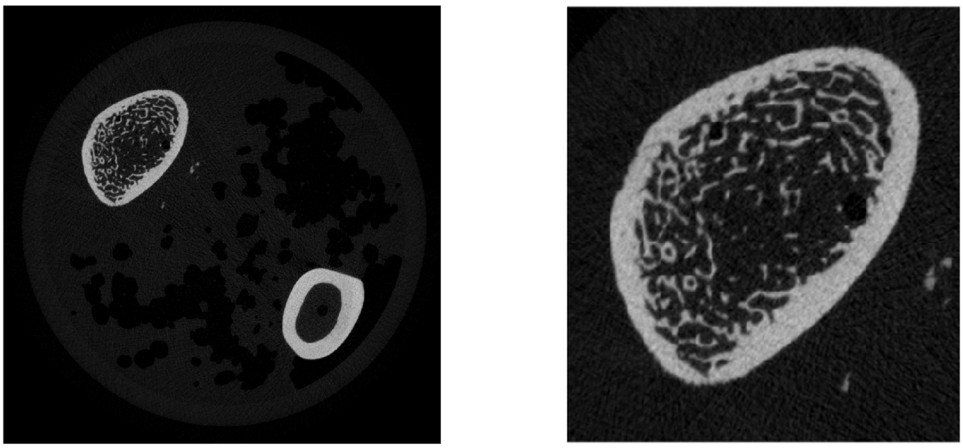


Figure 14. Video Viewer App for Volume of Interest (VOI) Slice Selection.
Images are cross-sections of a rat distal femur, with the top left being the most distal slice and the bottom right being the most proximal. Top left image contains both epiphyseal (brighter) and metaphyseal (darker, smaller trabeculae) tissue. The first slice with no epiphyseal tissue (250 in above images) is considered the beginning of the VOI. The next 100 slices in the proximal direction are selected as the VOI (20 µm slices, 2 mm total).

Now that the slices for the VOI have been identified, the corresponding TIFF images are imported into MATLAB. Each image is 1024x1024 pixels, but only the portion containing the specimen needs to be kept. Cropping will reduce the file size of the generated volume and allow the user to select only the bone of interest if there are multiple present in the same scan. The

edges of the VOI are identified in the top and bottom slices, and a buffer of 15 pixels is added to the ranges in each dimension. These ranges are then used to crop all the images in the series. Figure 15 shows an original 1024x1024 grayscale image and a sample 315x281 cropped image.

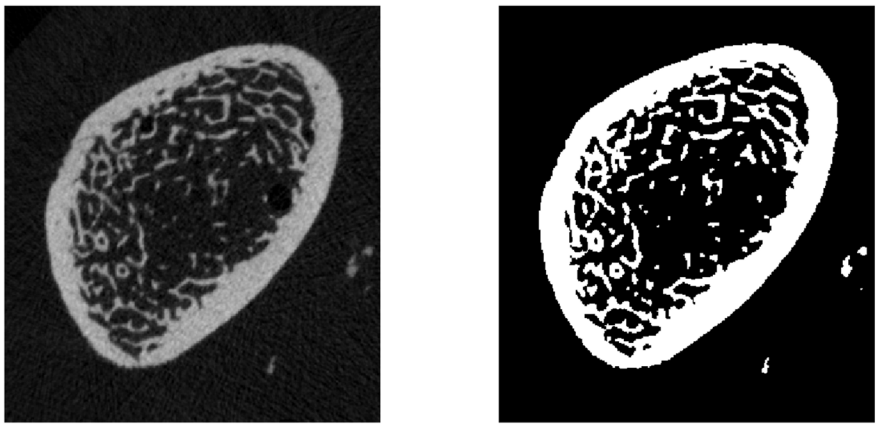


(a) (b)

Figure 15. TIFF Image Cropping
(a) original grayscale image, 1024x1024 pixels. (b) cropped image, 315x281 pixels.

The series of cropped images are then stacked to create a volume. For the 315x281 images in this example, the result would be a 315x281x100 three-dimensional grayscale image. The intensity values of each pixel in the 2D images now represent the intensity of a 3D cube, known as a voxel, having an assigned thickness equal to the side lengths of the square pixels. Here, the intensity values in these stacked images represents a 20x20x20 μm cube. The thickness of the slices is arbitrary and can be defined by the user; at this point the layers are known by their matrix indices which will be given a unit in 3D space later. Care must be taken to adhere to the resolution and slice thickness used in the original CT scan data.

Thresholding is applied to the stacked volume to differentiate bone from non-bone material and empty space. Voxel intensity values in the 3D image above an applied threshold are treated as bone and are mapped to 1; values below the threshold are treated as non-bone and are mapped to 0. The result is a new image, a binary copy of the original grayscale stack of TIFFs. A 2D representation of this binarization process is shown in Figure 16.

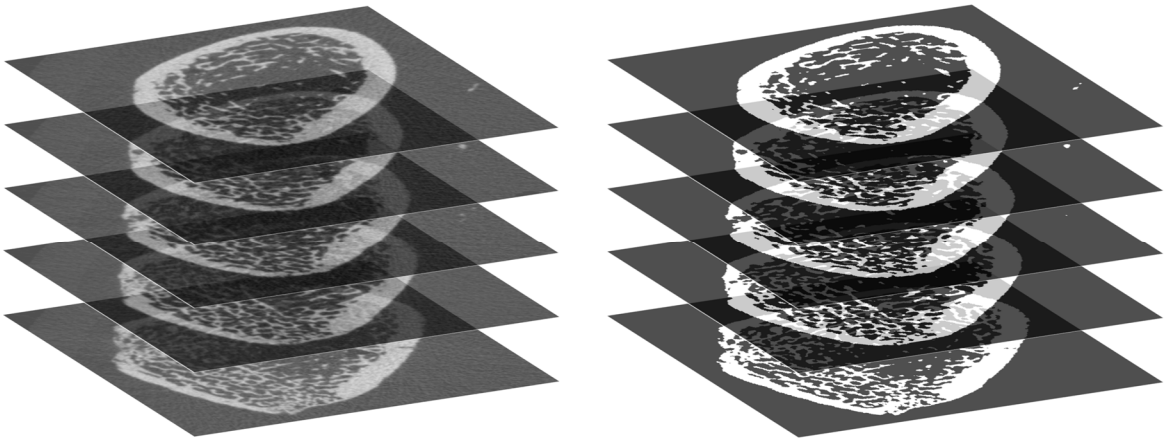


(a) (b)

Figure 16. Binarization of Grayscale Image.
(a) grayscale image with pixel values ranging from 0-202, (b) binarized image containing only 0s and 1s. The grayscale image does not span the full 0-255 as another portion of the image that was cropped out contained the highest intensity values. The binarization threshold was arbitrarily defined to be 100.

In actual implementation, the thresholding is applied to 3D the stacked volume. This allows for the same threshold value to be used throughout, rather than on a per slice basis. Functions in the MATLAB Image Processing Toolbox was used to binarize[2] the volume. The stacking and binarization is illustrated in Figure 17.

[2] The imbinarize() function was used. This applies Otsu's Method of threshold selection, which uses the grayscale value histogram to select a threshold value that minimizes the intraclass variance of the black and white pixels. This method provides a high-fidelity recreation of the bone material, as verified by visual inspection of selected frames. See the CT_to_Volume script in Appendix C for implementation details.



(a) (b)

Figure 17. Stacking and Binarization
(a) grayscale stack of images with voxel intensity values ranging from 0-202, (b) binarized stack of images containing only 0s and 1s. The grayscale stack does not span the full 0-255 as another portion of the image that was cropped out contained the highest intensity values. The binarization threshold was arbitrarily defined to be 100. 100 slices (20 μm thickness) are used to form the 3D image.

The result of this stacking and binarization process is a volumetric representation of the scanned bone. Utilizing the Volume Viewer, the visualization in Figure 18 of the assembled bone image can be verified.

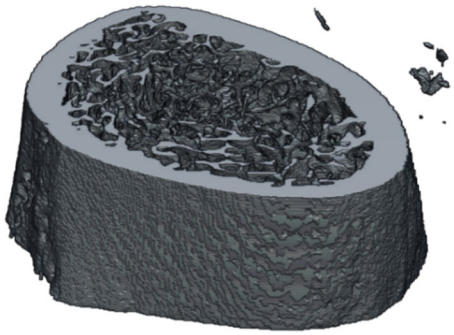


Figure 18. Assembled Binarized Volume
3D binary image visualized with MATLAB Volume Viewer. Volume shown is a 2mm thick section of a rat distal femur metaphysis, with the top face being the most proximal. Floating pieces on the upper right are inclusions from the CT scan and do not have any significance.

The volume that results may contain multiple bodies. Bone fragments or other inclusions from the CT scan can be seen floating in the top right corner of Figure 18; disconnected trabeculae are also present although hidden from view. The loads to be applied for FEA in the present study require a single, well connected body. Otherwise, rigid body motion can occur, in which case the numerical analysis will fail to converge.

A connectivity filter[3] is used to remove the unconnected parts, keeping only the largest single body. Face connectivity is imposed, as corner and edge connectivity will produce poor model performance for FEA. Only the largest connected component is kept for further analysis. The body generation process is summarized in Figure 19. The removal of exterior objects as well as interior, poorly connected trabeculae can be observed.

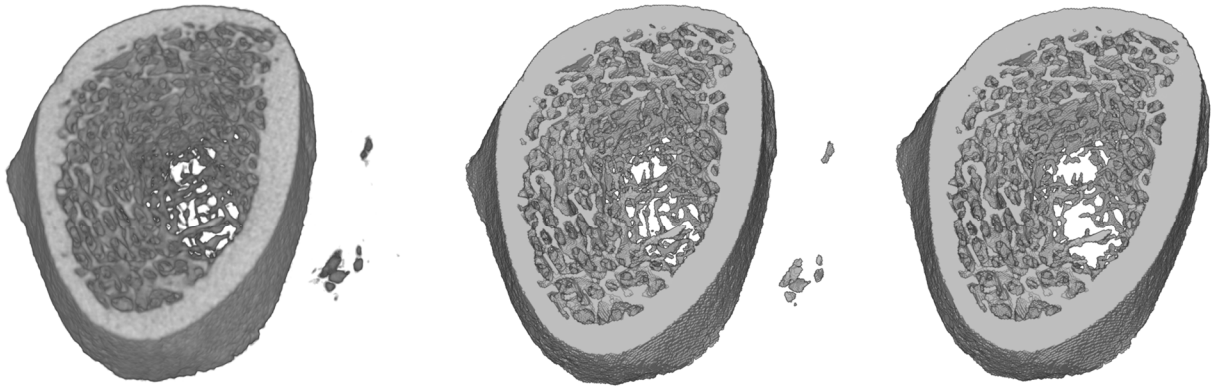


Figure 19. Connected Body Generation Process
(a) Original stack of grayscale images, (b) binarized image obtained using Otsu's Method, (c) single connected body after connectivity filter is applied. Some minimal thresholding has been applied to (a) for viewing, without which the image would appear as a black prism. (b) contains 260+ unconnected bodies, (c) is the largest of them. Volume shown is a 2mm thick section of a rat distal femur metaphysis, with the top face being the most proximal.

[3] The bwconncomp() function in the MATLAB Image Processing Toolbox is used with a connectivity parameter of 6, indicating connectivity only in the three-dimensional six-connected neighborhood. This will define two cubes as connected only if they share a face, rather than an edge or corner. See the Conn_Filt script in Appendix C for implementation details.

3.1.1 Separation of Cortical Shell

An isolated body of the trabecular structure is generated to allow for FEA simulations without the cortical shell. An image opening process[4] is applied to the volume, which isolates the cortical shell by removing the regions of the image that are below a defined thickness threshold. The isolated cortex is then subtracted from the original image, producing a representation of the trabecular structure without the cortical shell. Face connectivity[5] is imposed once again to ensure the final product is a single body suitable for use in FEA. Figure 20 shows the progression of the various volumes throughout the trabecular isolation process. A more rigorous version of this algorithm has been employed elsewhere [39].

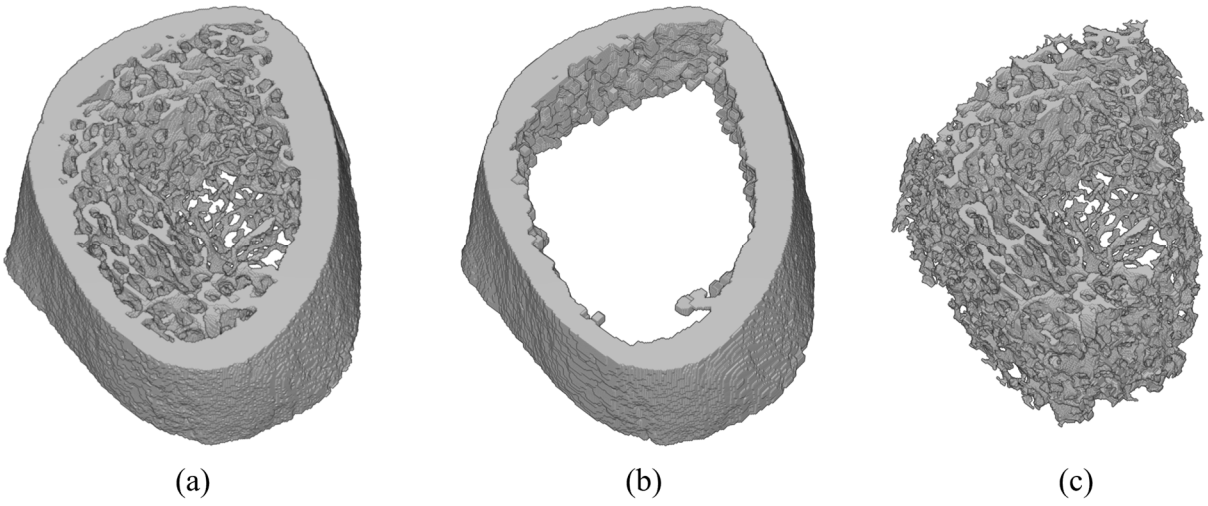


Figure 20. Trabecular Isolation Process
(a) Fully connected body, (b) isolated cortical shell, (c) isolated trabecular structure. Volume shown is a 2mm thick section of a rat distal femur metaphysis, with the top face being the most proximal.

[4] The imopen() function in the MATLAB Image Processing Toolbox is employed using a 5x5x5 cubic structural element. See the Isolate_Trab script in Appendix C for implementation details.
[5] The bwconncomp() function in the MATLAB Image Processing Toolbox is used with a connectivity parameter of 6, indicating connectivity only in the three-dimensional six-connected neighborhood. This will define two cubes as connected only if they share a face, rather than an edge or corner. See the Conn_Filt script in Appendix C for implementation details.

3.2 FEA Pre-processing

After proper thresholding is applied and a single body has been isolated, the volume can be used to generate a finite element mesh. The binarized volumetric image of the bone is effectively a 3-dimensional matrix of 1s and 0s. As illustrated in Figure 21 these matrix entries can be represented as a cube at that location in space, the value of the entry corresponding to an active or inactive voxel. By defining these cubes as elements and placing nodes at each of their corners, the voxels can be converted directly to hexahedral elements in a FE mesh.

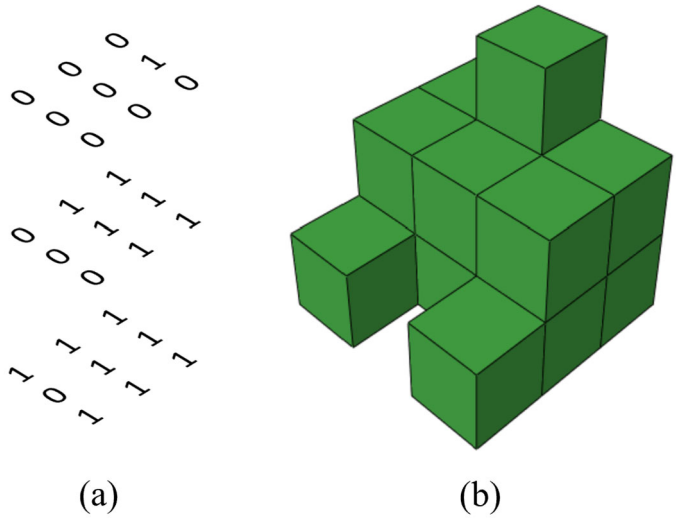


Figure 21. Sample Assembly of Cubes from 3D Binary Matrix
(a) 3D binary matrix, (b) volumetric representation of cubes assembled from the binary matrix. Zeros are treated as empty space and ones are treated as bone material.

3.2.1 Mesh Generation

For a mesh to be properly defined, the following parameters need to be determined: unique identification numbers for each node and element, spatial coordinates for each node, and the nodal connectivity of each element. Only the active voxels should be included in the final mesh. To aid in this process, an element and node numbering scheme was devised such that the index of the matrix entry - V(i, j, k) - could be used to assign a unique element number to each voxel (Equation 1). Figure 22 shows the element numbering convention on a sample 3D matrix. A feature of this numbering scheme worth mentioning is that both the voxels that contain 1s and those that contain 0s are assigned an element number.

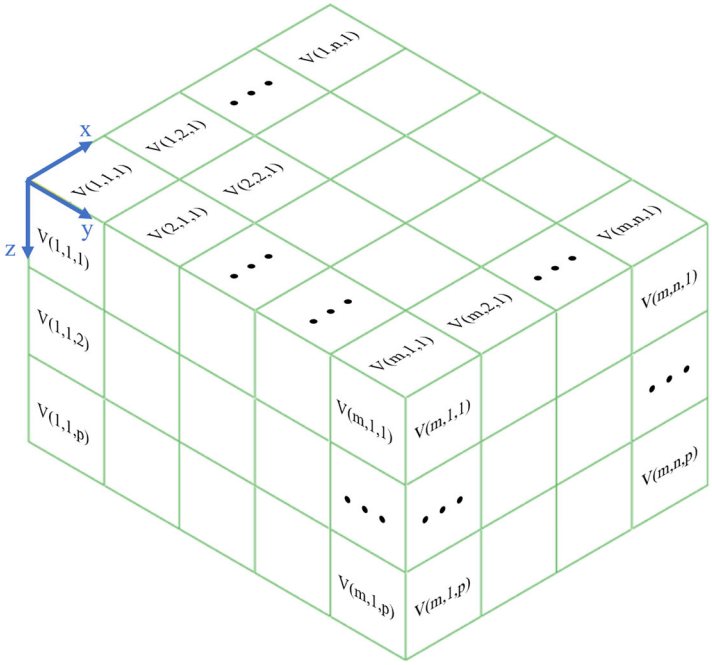


Figure 22. Matrix Index Element Numbering Convention

Size of matrix is m×n×p, index of each entry - V(i,j,k) - is used to assign an element number to the voxel. The m index of the matrix corresponds to the y direction of the model, the n index corresponds to the x direction, and the p index corresponds to the z direction.

$$Element\ Number\ =\ n(i-1)\ +\ j\ +mn(k-1) \tag{1}$$

$$for\ V(i,j,k), \qquad size(V) = m \times n\ \times p$$

A similar numbering scheme was used for each of the nodes at the corners of the elements. The nodes can be visualized as another matrix, with one additional entry in each of the three dimensions, shown in Figure 23. Thus, using the new matrix index N(I, J, K), each node can be given a unique node number. Using the formulas in Equations 2-6, the spatial coordinates for each node can be determined based on the matrix index.

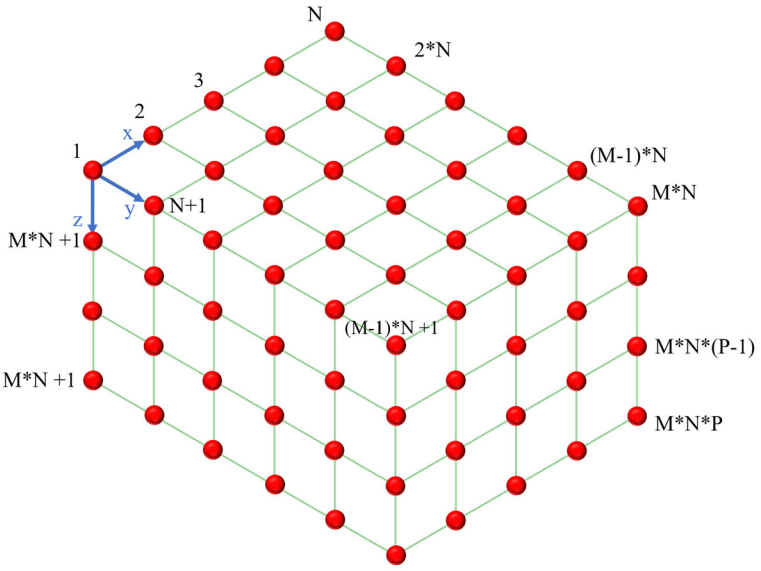


Figure 23. Matrix Index Node Numbering Convention
The nodes can be treated as a matrix with one additional entry in each direction, with respect to the volume image.

$$Node\ Number\ =\ N(I-1)\ +\ J\ +MN(K-1) \tag{2}$$

$$x_n = w_{pixel} * (J-1) \tag{3}$$

$$y_n = w_{pixel} * (I-1) \tag{4}$$

$$z_n = w_{pixel} * (K-1) \tag{5}$$

$$M = m+1, \qquad N = n+1, \qquad P = p+1 \tag{6}$$

To determine the nodal connectivity of each element, the numbering convention in Figure 24 is used. The numbers of each of the nodes can be directly determined from the index – V(i,j,k) – of the element. The nodal values based on the matrix index are listed in Equations 7-14.

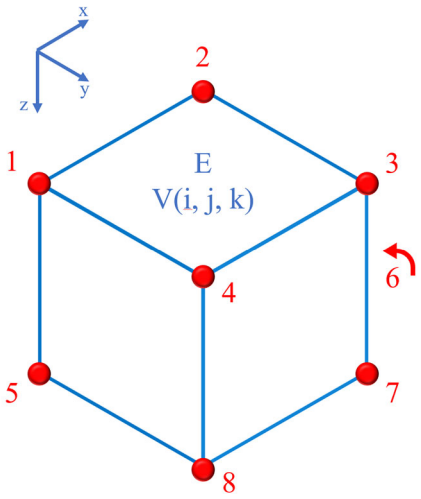


Figure 24. Element Nodal Connectivity Convention
Red numbers are the local nodes numbers for the element; the global node numbers are calculated and stored in the element connectivity array. The global node numbers can be determined directly from the matrix entry, V(i,j,k). The local numbering convention is that found in ABAQUS element documentation.

$$N_1 = (n+1)(i-1) + j + (m+1)(n+1)(k-1) \quad (7)$$

$$N_2 = N_1 + 1 \quad (8)$$

$$N_3 = N_2 + (n+1) \quad (9)$$

$$N_4 = N_1 + (n+1) \quad (10)$$

$$N_5 = N_1 + (m+1)(n+1) \quad (11)$$

$$N_6 = N_2 + (m+1)(n+1) \quad (12)$$

$$N_7 = N_3 + (m+1)(n+1) \quad (13)$$

$$N_8 = N_4 + (m+1)(n+1) \quad (14)$$

For the element connectivity, the mesh generation algorithm loops through each of the voxels in the 3D image. If the value is 1, the element number and connected nodes get written to the Element Connectivity Array (ECA). If the voxel value is 0, the element is skipped and not included in the ECA[6]. The 2D array generated has 9 columns and the number of rows is equal to the number of active elements in the mesh. The entries in each row correspond to values in Equation 15.

$$[\ Element\ Number \quad N_1 \quad N_2 \quad N_3 \quad N_4 \quad N_5 \quad N_6 \quad N_7 \quad N_8\ ] \tag{15}$$

Similarly, for the nodal coordinates the algorithm loops through each node and checks if any of the surrounding elements are active. For a node in the internal volume of the image, the values for each of the eight voxels that it touches are added together. If this sum is non-zero, then one or more of the elements to which the node belongs are active. In this case, the node and its coordinates are written to the Nodal Coordinate Array (NCA). If the sum of that node's voxels is 0, no active elements touch the node and it is not included in the NCA[7]. The NCA is a 2D array consisting of 4 columns and the number of rows is equal to the number of active nodes in the model. The entries in each row correspond to values in Equation 16.

$$[\ Node\ Number \quad x_n \quad y_n \quad z_n\ ] \tag{16}$$

[6] See Elem_Conn function in Appendix C for implementation details.
[7] See Node_Coord function in Appendix C for implementation details.

Generating the NCA is more complex programmatically than the ECA. The nodes on the corners of the matrix only touch 1 element, edge nodes touch 2 elements, face nodes touch 4 elements, and internal nodes touch 8 elements. Thus, each case needs to be treated differently. The algorithm employed herein starts at the corners and then proceeds through the remaining classes: edges, faces, and volume. This means that as the NCA is written, the nodes are not in numerical order. For the purposes of import to ABAQUS, this is not an issue, but generating node sets and handling the results in post-processing are simplified by having the NCA and ECA in ascending order. The ECA is already sorted due to the algorithm used in its generation, so only the NCA needs additional sorting by node number[8].

Once the sorted ECA and NCA have been generated, they are written to an ABAQUS input file (INP) with proper formatting so that the mesh can be imported into ABAQUS. Since some nodes and elements that had unique numbers were not included due to the voxels having a zero value (non-bone material), the resulting mesh has element and node number values that are non-consecutive. For instance, the elements included in the mesh may be numbers 6, 12, and 372 - rather than 1, 2, 3. This does not impact the result, as the assignments used are arbitrary. ABAQUS has no issue importing the mesh with non-consecutive values, so long as the nodes have proper coordinates and the element connectivity is defined with the proper number of nodes.

[8] This is accomplished using the sortrows() function in MATLAB. See Node_Coord function in Appendix C for implementation details.

3.2.2 Node and Element Set Generation

To simplify the application boundary conditions (BCs), node and element sets are defined in MATLAB to be included in the INP file. These sets can then be called by name, as opposed to writing the BCs for each node or element, which would substantially increase the file size and input processing time for the simulations.

The two classes of mechanical tests to be simulated on each volume are whole specimen compression (WSC) and reduced platen compression (RPC) tests. To apply the necessary BCs for WSC, all the nodes on the top and bottom faces need to be included in Node Sets. This is done by taking the NCA and selecting only those nodes with a z-coordinate inside a specified range. New smaller NCAs are generated comprising these node sets[9].

For the RPC simulations, the nodes inside the area of contact with the platen need to be identified. This is done by calculating the in-plane distance of each node from a user-defined center point. The points that are within a specified radius are kept as a new node set[10]. To mimic the platen selection process used during RPC testing, an endocortical circle measurement needs to be made. Image processing and analysis functions[11] are used on the top face of the specimen to isolate the trabecular area, which is approximated as an ellipse. The minor axis of this ellipse is treated as the diameter of the endocortical circle passing through the centroid of the trabecular area. As in actual RPC testing, 70% of this diameter is used as the platen size. Figure 25 shows the resulting areas for the present example. The inner circle contains the nodes which were assigned to the RPC Node Set.

[9] This algorithm employs the unique() function in MATLAB to identify a user defined number of layers in the mesh. See Layer_Nodes function in Appendix C for implementation details.
[10] See RPC_Nodes function in Appendix C for implementation details.
[11] The cancellous compartment is isolated using the bwconncomp(), imfill(), imclose(), and regionprops() functions in the MATLAB Image Processing Toolbox. See the Find_RPC_Area script in Appendix C for implementation details.

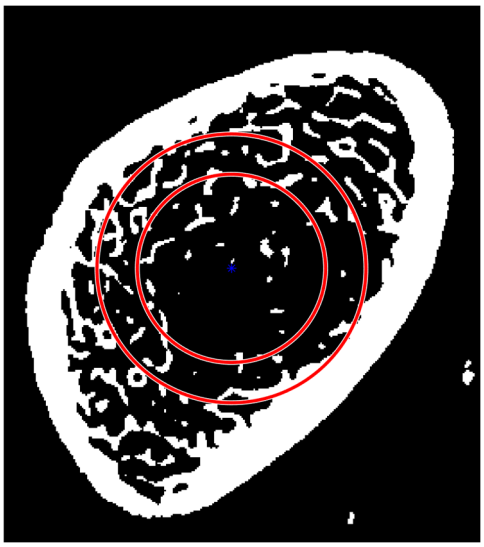


Figure 25. RPC Area Selection
Outer red circle is simulated endocortical circle as calculated using the centroid and minor diameter of the cancellous area. The inner red circle has 70% of the endocortical diameter and represent the platen contact area. Image is a cross section of a rat distal femur metaphysis, 2mm proximal to the end of the growth plate.

For equilibrium of forces, a circle of the same size and location is used on the bottom face of the specimen. It should be noted that the present image analysis method is sufficient for these DFM specimens but would not be effective on tibial RPC specimens as the cancellous compartment is not elliptical in shape.

Selecting only the nodes on the top and bottom faces can produce models that do not accurately mimic the behavior of a compression specimen. Trabeculae that are slightly below surface and do not contain surface nodes would be excluded; after loading they may protrude through the plane representing a physical platen. This issue is addressed by selecting 3 layers of nodes in each of the node sets. Similar practices have been used in other studies [40].

3.2.3 Boundary Conditions

The compression test simulation in each of the loading scenarios is accomplished by specifying a displacement for the top node set and fixing the displacement on the bottom. A 0.1mm displacement in the negative z-direction is specified for the top face, representing a 5% strain for these 2mm specimens. Both a smooth and rough case were simulated for the WSC and RPC test. The rough cases had the z-displacement specified as mentioned, as well as having the x and y-directions fixed for each of the nodes in the top and bottom node sets. This would simulate a compression test where the ends of the specimen were glued to the platens. Conversely, the smooth cases had the same applied z-displacement, but all the nodes could move in the x and y-directions. This represents a compression test using lubricated platens. In the case of the smooth models, a single node in the bottom node set was fixed in the x and y-direction to prevent rigid body motion. Each of the boundary conditions applied in the present study are listed and abbreviated in Table 1 and are illustrated in Figure 26 on a sample cubic volume. The orange arrows represent specified degrees of freedom.

Table 1. Boundary Condition Abbreviations

Compression Test	Friction	Abbreviation
Whole Specimen	Rough	WR
Whole Specimen	Smooth	WS
RPC	Rough	RR
RPC	Smooth	RS

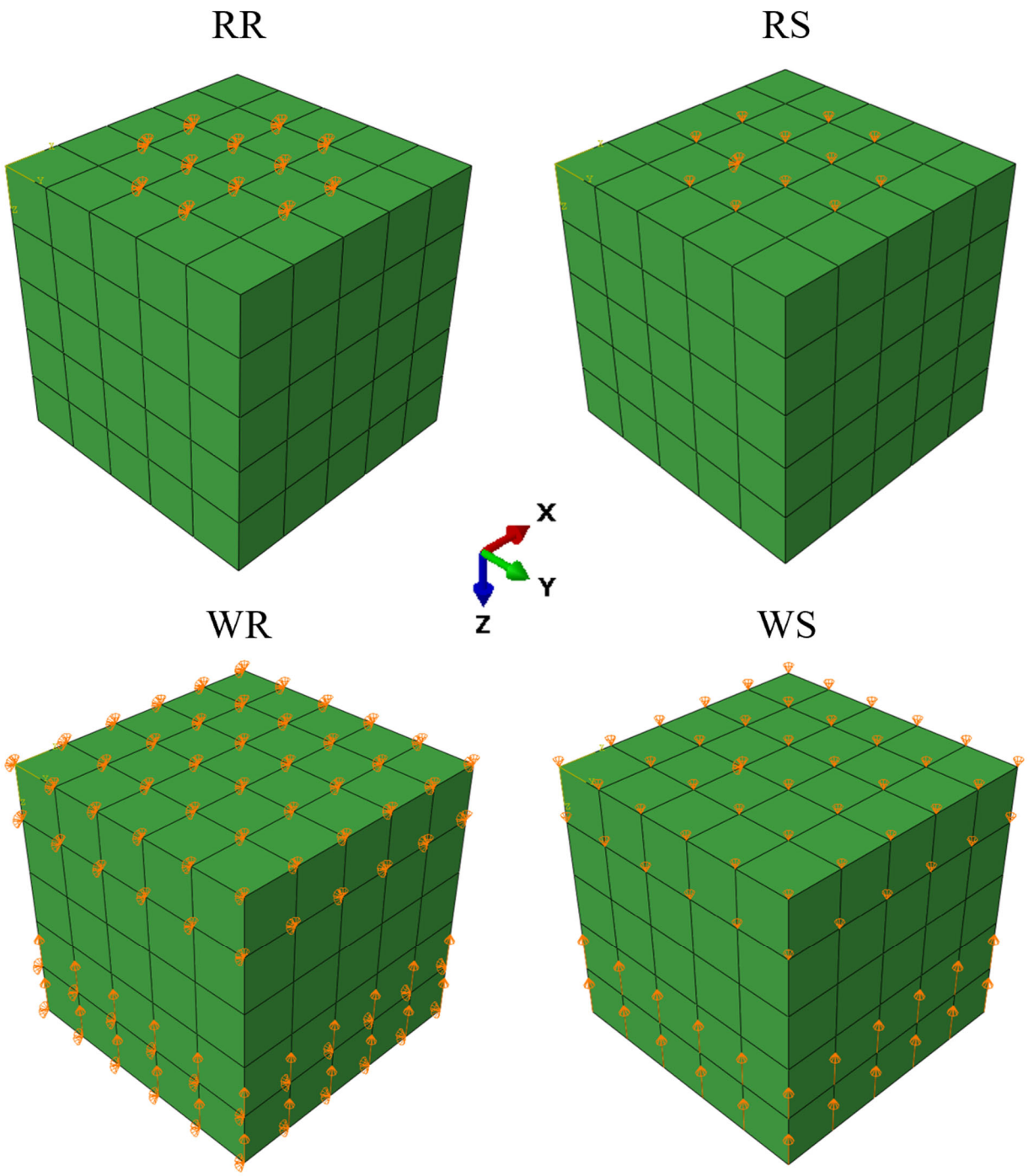


Figure 26. Applied Boundary Conditions.

Boundary conditions (BC) are illustrated for the reduced platen compression (RPC) and whole specimen compression (WSC) simulations. RR = RPC Rough, RS = RPC Smooth, WR = Whole Rough, WS = Whole Smooth. Each of these is simulated separately for every volume. Rough means the displacement in fixed in the x and y, with applied displacement in the z direction. Smooth means applied displacement in the z direction, and free to move in the x and y. The orange cones represent specified (fixed or otherwise) degrees of freedom. The illustrations are "upside down" relative to RPC specimens, in that the "top" (most proximal) surface has the applied displacement and the "bottom" (most distal) surface is fixed. Two layers of nodes are shown with applied BCs, the specimens had three layers selected for the simulations.

3.2.4 Input File Generation and Model Specifications

There are multiple options for defining the rest of the FE model. The ABAQUS CAE graphical user interface (GUI) can be used to generate the boundary conditions (BCs) and to assign material properties. With models of this size (2 million+ elements), the GUI can be cumbersome to use. Alternatively, the BCs can be written to the input file from MATLAB either manually or using a repeatable script. Either way, the output is a completed input file for the ABAQUS solver. For the models herein, low level file I/O functions in MATLAB[12] were used to write the complete ABAQUS input files (INP) line by line.

The material properties assigned to each element were a Young's Modulus of 10 GPa and Poisson's ratio of 0.3 [37, 41]. See the discussion in section 5.3 for suggestions regarding these material properties. The element type assigned was ABAQUS linear hexahedral C3D8.

3.3 FEA Processing

After the input file has been generated, it is submitted for solving in ABAQUS 2018. The linear analysis is performed in ABAQUS Standard. For models of this size, the direct sparse solver in ABAQUS requires extensive memory resources and many hours of computation time to arrive at a solution. The iterative solver requires less memory and performs much more quickly, and for these reasons was selected for the present analyses. The simulations were run on the Ada supercomputer housed by Texas A&M University High Performance Research Computing (TAMU HPRC). Only one CPU was utilized, with varying amounts of memory reserved for each type of simulation. The simulations required 30-35 MB RAM and took 1-2.5 hours to solve

[12] fprintf() was employed to write the necessary information to a text file, which was saved with the .inp ABAQUS file ending. Additional scripts were developed to assign different boundary conditions to each model. See the Inputs_Generation script and Input_Write function in Appendix C for implementation details. A sample ABAQUS input file generated with this technique is available in Appendix D.

for the whole specimen. The trabecular bodies were less computationally expensive, requiring 12.5-13 MB RAM and 20-40 minutes to complete.

3.4 FEA Post-processing

After the simulations were completed, the output databases (ODBs) were opened in the ABAQUS CAE GUI via remote access on the Ada cluster's web portal. For each model, a report file was generated containing the appropriate stress and strain quantities for further processing. This included the principal stresses and strains, as well as the full stress and strain tensor components[13]. The values were calculated and reported at the centroid of each element, as opposed to the nodes or integration points; this saves considerable time in generating and memory in storing the output data. An additional report was generated for the reaction force (RF) values at each node. The reports were generated in comma separated value (CSV) format to ease the import of the data into MATLAB[14].

The total reaction force was determined by summation of all nodes in the BC node sets. These RF values can be compared to the yield force measurements from mechanical testing in past experiments of the same bones used to generate the CT scans.

For a more detailed analysis of the difference between specimens, the element values are used to compare strain distributions. Strain energy density was calculated using the principal stress and strain values in the table for each element. Equation 17 shows the formula for strain energy density that was used. With this strain energy density, an effective strain was calculated using Equation 18.

[13] Only the principal values were used in calculation and contour plotting, so future simulations need not spend the resources to write and store the full tensors.
[14] The CSV report files were imported into MATLAB using the readtable() function, with the opts parameter adjusted to only include the desired columns.

$$U = \frac{1}{2}(\sigma_{max}\epsilon_{max} + \sigma_{mid}\epsilon_{mid} + \sigma_{min}\epsilon_{min}) \quad (17)$$

$$\epsilon_{eff} = \sqrt{\frac{2U}{E}} \quad (18)$$

U is strain energy density, calculated using the principal stresses and strains. ε_{eff} is an effective strain calculated using the strain energy density and Young's Modulus (E). These values are calculated in MATLAB using the element centroidal stress and strain values exported from ABAQUS.

Both the strain energy density and effective strain were added as columns to the original MATLAB tables, and the new tables were saved for future use. For comparison between loading cases and different specimen, the distribution of effective strain is visualized in histograms and cumulative density functions (CDF). To make these comparisons more meaningful, new tables were generated containing the strain values from the whole specimen (with cortex) simulations for only the elements that were also present in the isolated trabecular structures. This means that the same volume can be compared between the load cases that include the cortex and those with it removed. This is explained in more detail in the RF results section.

Since the effective strain was calculated in MATLAB and does not exist in the ABAQUS output database (ODB), there is no way to generate contour plots of the bone using this value. However, the E_max_principal_Abs value – which is the maximum principal strain in terms of magnitude, not just the largest positive number – is present in the ODB and correlates well with effective strain (Appendix B, Figure 57). The contour plots generated using the absolute maximum principal strain (AMPS) are therefore representative of the effective strain. If failure is determined from a percentage of the elements having reached a predefined effective strain

[42], then the regions of high strain in the AMPS plots will contain those elements in failure. The added benefit of using AMPS rather than effective strain is that the regions in compression and tension are distinguished. The effective strain as calculated from strain energy does not provide this delineation.

It is worth noting that due to size of the models and ODBs, an appropriate workstation is needed for postprocessing. When making use of the Texas A&M High Performance Research Computing (HPRC) portal, a computer with at least 90 Mbps internet connection is necessary for model manipulation. A computer with a local license of ABAQUS would likely need 16+ GB of RAM and a discrete GPU. For sufficient image quality when taking screen shots of the contour plots, a visual display greater than 1080p is recommended.

3.5 Animal Selection for FEA Validation

To validate the FEA method developed in this study, two animals were selected for simulation[15]. R227 and R223 were chosen based on their RPC mechanical testing results from a previous study, where more information on the animals can be found [32]. R223 had a maximum load in the DFM RPC test that was considerably below average, and R227 was considerably above average. This difference was not well supported by pQCT and μCT densitometric quantities (Appendix A).

[15] These specific animals were six-month-old male Sprague-Dawley rats that received an injection of Risedronate prior to 4 weeks of hindlimb unloading (HU). R227 was terminated at the completion of HU, R223 was terminated after an additional 6-week recovery period beyond HU. Body weight information is available in Appendix B.

4. RESULTS

The following section compares the volume results for the images from the scans of the two bones selected for the present study to investigate differences in their structure. Additionally, the FEA reaction forces (RF) are presented and compared with past mechanical testing results from the reduced platen compression (RPC) test and other anatomical sites in order to determine potential reasons for the outlier behavior of the selected animals in mechanical testing.

Following the force results, contour plots and strain distributions are presented to compare the applied boundary conditions (BC). The final section of these results details the computational requirements for the processes developed in this study.

4.1 Image Processing

4.1.1 Effect of Thresholding on Generated Body

The threshold value selected during binarization determines how much bone material is included in the binarized images. These effects in 2D are shown in Figure 27.

Figure 27. 2D Image Thresholding Value Comparison.
(a) Original grayscale image with intensity values from 0-202; (b) binary image using 135, or 2/3 of the maximum value, as a threshold; (c) binary image using 101, or 1/2 of the maximum value, as a threshold; and (d) binary image using 67, or 1/3 of the maximum value, as a threshold. The lower the threshold value, the more material is included in the binarized image.

A high threshold value produces an overly thin recreation. A well-suited threshold leads to an accurate binary image. A threshold that is too low results in an overly thick cortex and a cancellous region that is too dense. In 3D this process is similar; however, the effects are exacerbated by the connectivity filter. Lower thresholds produce thinner trabeculae, often leading to more disconnected bodies which are removed in filtration. Figure 28 shows three bodies that were generated using the same set of grayscale images, but with different binarization thresholds. After the connectivity filter is applied, the large difference in included trabecular structure is evident.

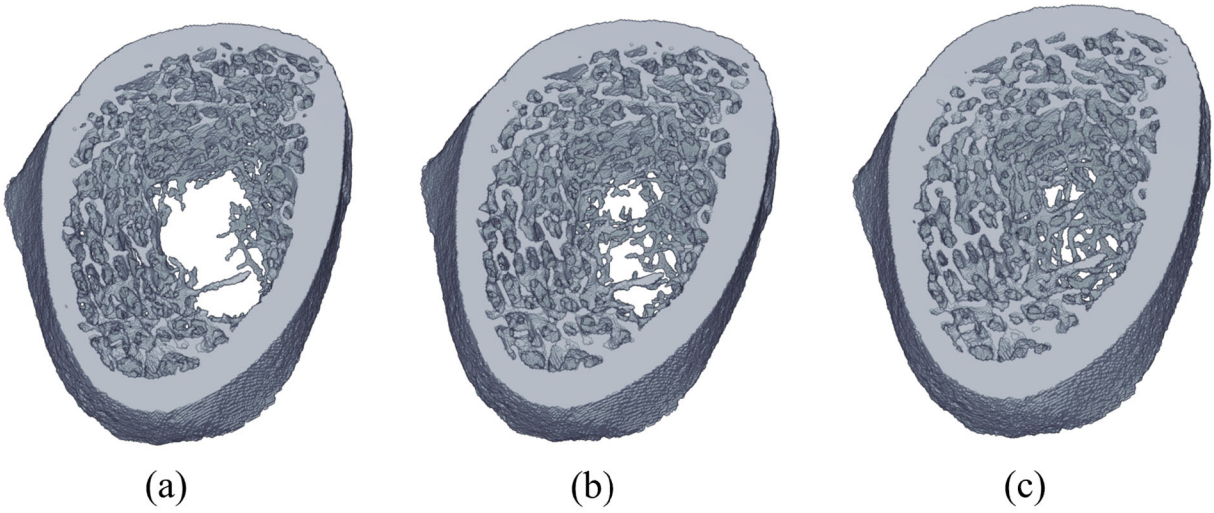


Figure 28. Thresholding Effect on Final Connected Volume.
Images are connectivity filtered volumes that were binarized using: (a) too high a threshold value, producing thin trabeculae that end up getting removed; (b) an appropriate threshold value, producing a representative structure; and (c) too low a threshold value, producing an overly thick specimen. The bone segment shown is a 2mm section of the rat distal femur metaphysis, the top surface being the most proximal.

A closer inspection shows a difference in the features even after the connectivity filter is applied. Figure 29 magnifies the upper right corner of the different generated volumes and the same section of the original grayscale image representing the top surface.

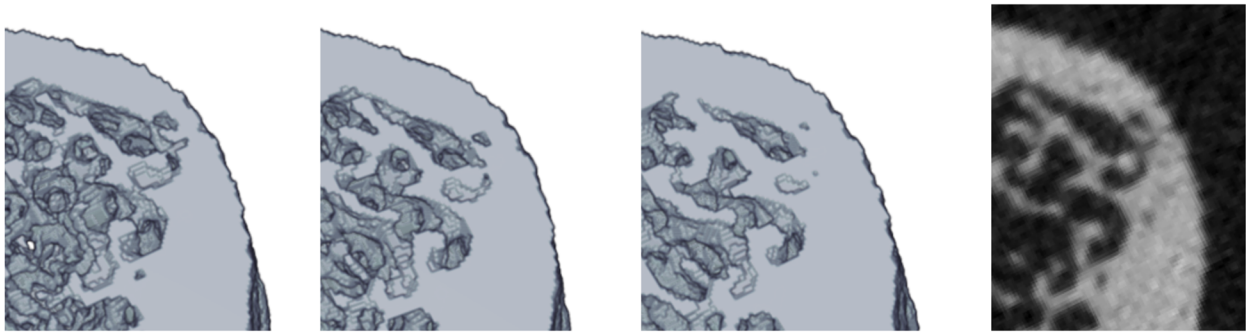


Figure 29. Enhanced View for Thresholding Comparison
Images are upper right corner of the volumes shown in Figure 28; produced with (a) too high a threshold value, producing thin trabeculae; (b) an appropriate threshold value, producing a representative structure; and (c) too low a threshold value, producing an overly thick specimen. (d) is the same region in the original grayscale image of the top slice. The size of the trabeculae and holes in the cortex can be visually compared, indicating (b) is the closest recreation. The bone segment shown is a 2mm section of the rat distal femur metaphysis, the top surface being the most proximal.

As the threshold is lowered, geometric features like holes and pores become less pronounced. If the threshold is too high, the included trabeculae are overly thin; this will produce FE meshes that do not transmit loads properly and inaccurately predict the stiffness and strength of the actual bones.

4.1.2 Specimen Attributes

The volumes generated herein for the purposes of FEA are approximations of real bones. As previously discussed, the traits of these structures can be manipulated through changes in the image acquisition and manipulation processes. They should thus be viewed as representations of their physical counterparts, but not perfect recreations. The rendered volumes will be referred to by the names of their CT scans, T160416 and T160435. The real animals from which these bones came were R227 and R223, respectively. Any results derived from the physical bones, such as mechanical testing data, will be referenced with respect to the animal numbers (R22X). Simulated strength quantities from FEA will be referenced to the volume T-numbers.

The volumes created for the two specimens in this study are shown in Figure 30. For any volumes rendered here and throughout, quantitative comparisons cannot be made visually as the volumes are not to scale. However, qualitative observations regarding the connectedness and prevalence of the trabecular structure are valid. The trabecular structure of both specimens is more directly observed from the top views in Figure 31.

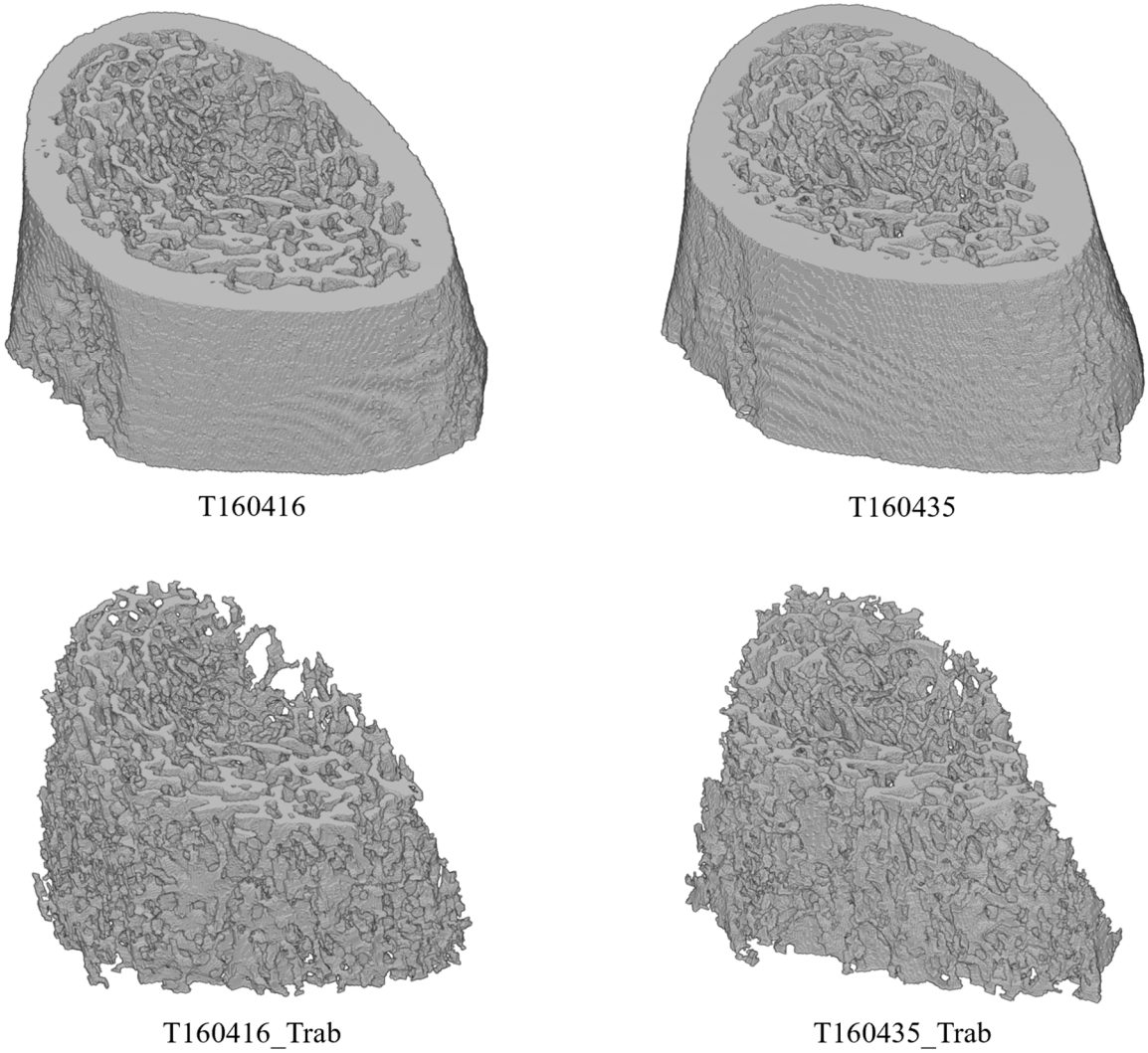


Figure 30. Volume Images Used for FEA Meshing.
Volumes shown are from bones of two animals, a 2mm section of the rat distal femur metaphysis, the top surface being the most proximal. Volumes with "_Trab" in the name were generated by removing the cortical shell from the original image through an automated image opening process (Figure 20). Thresholding was done using Otsu's Method. Images are not to scale, but the top and bottom images are registered to be from the same viewpoint.

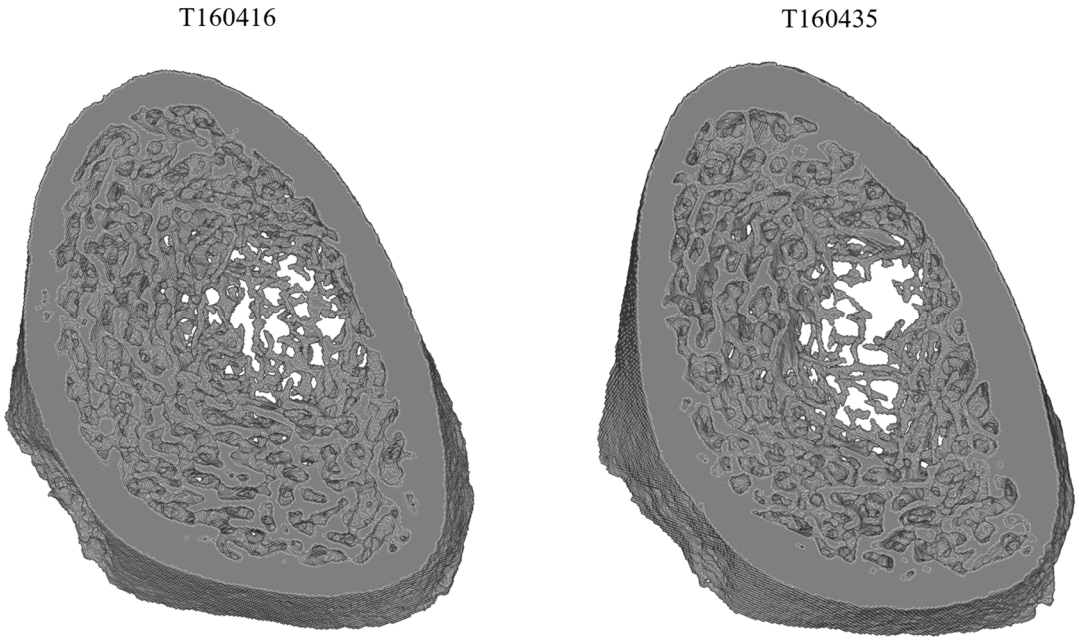


Figure 31. Top View of Whole Specimen Volume Images.
Top view illustrates the comparatively increased "hollowness" of T160435, with less well-connected trabeculae in the center regions of the cancellous compartment. Volumes shown are from bones of two animals, a 2mm section of the rat distal femur metaphysis, the top surface being the most proximal. Thresholding was done using Otsu's Method. Images are not to scale.

T160435 has a larger "hollow" space in the center of the cancellous region than does T160416. This can be seen quantitatively in the number of nodes included in the node sets for each volume, which is presented in the next section.

4.2 FEA Pre-processing Element and Node Sets

This section presents the number of nodes and elements generated for each volume. The nodes sets used for the application of boundary conditions (BC) are included for comparison between the specimens.

Figure 32 shows that T160435 contains 7% more elements than T160416. This means it is a "larger" bone, in the sense that more bone mineral is present in the 2mm test section used in this study. The difference in nodes is only 5%. The difference in nodes being less than the difference in elements indicates that T160435 has a higher percentage of cortical tissue than does T160416. Elements in the cortex have less free surface nodes, since they are surrounded on more sides by other elements; therefore, a higher node to element ratio implies more trabecular tissue, as there are more free nodes belonging to only one element. The differences in the trabecular volumes for each specimen are minimal, but this once again shows that T160435 has a higher percentage of cortical tissue, since more elements are removed to isolate the trabecular structure.

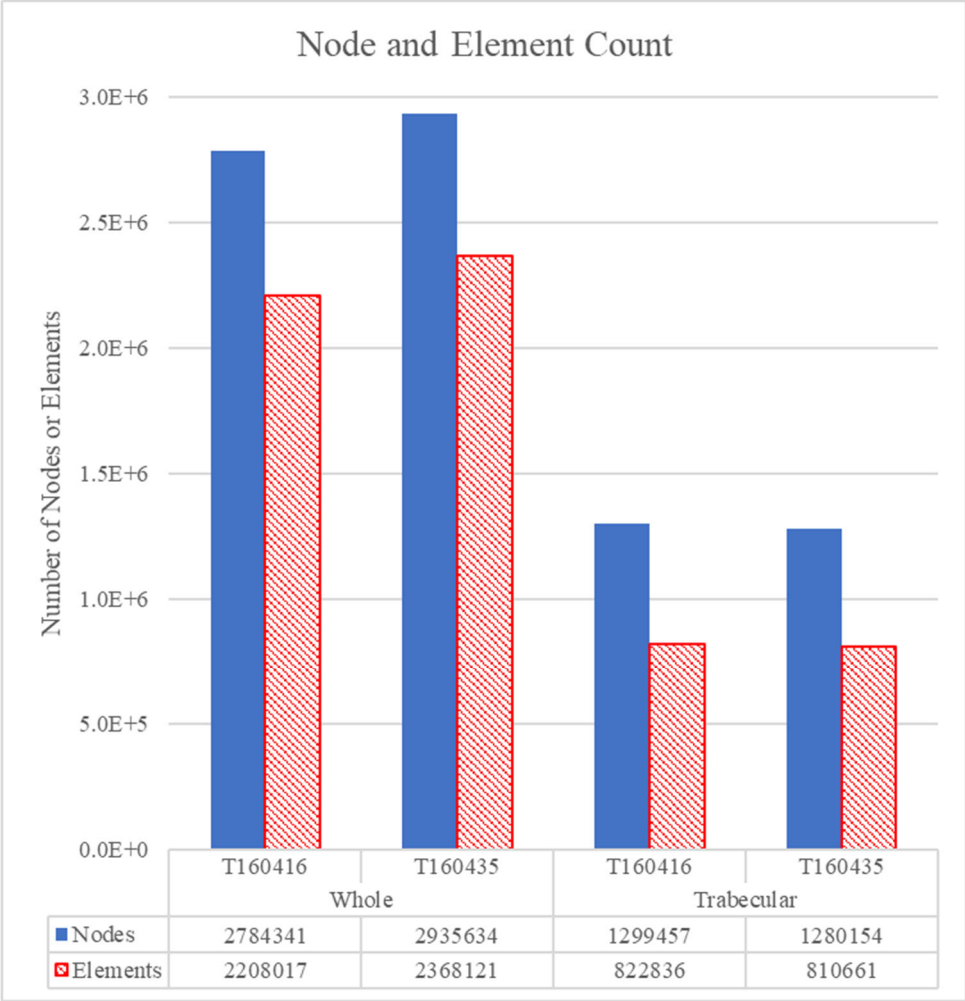


Figure 32. Node and Element Counts in Each Volume
Whole is the volume with the cortex, Trabecular is after the shell is removed.

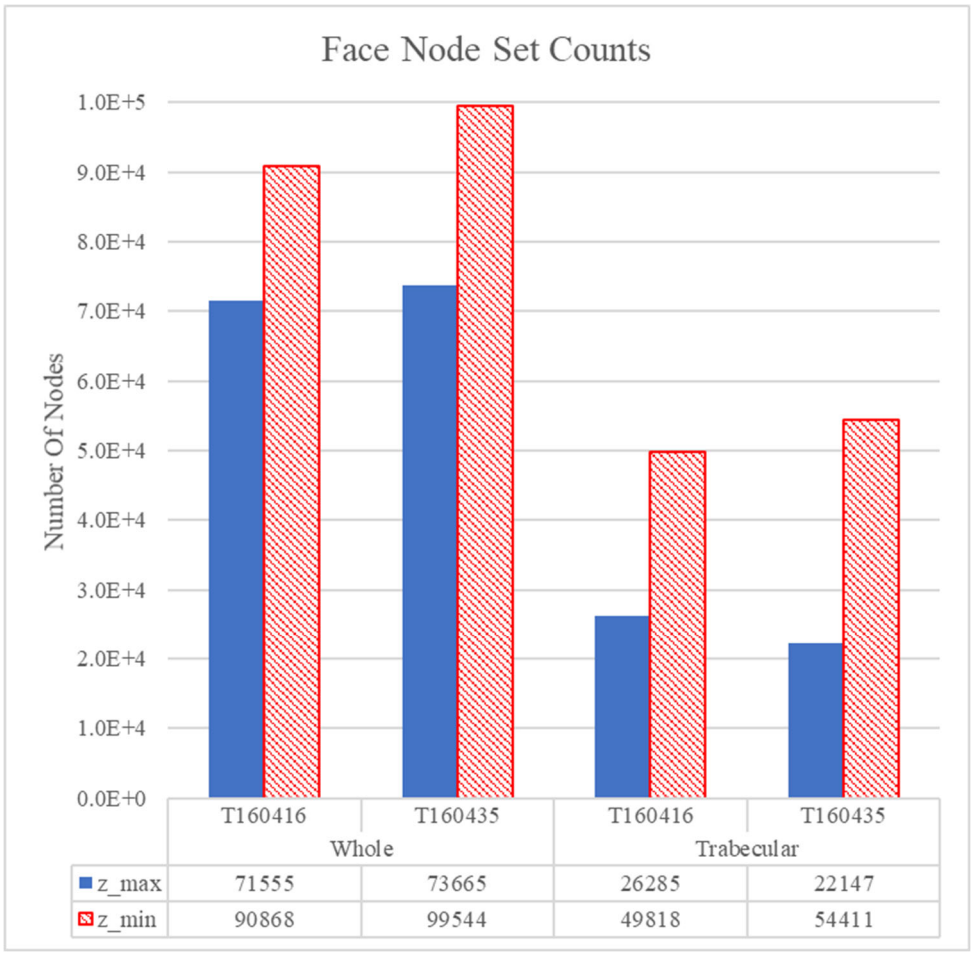


Figure 33. Top and Bottom Face Node Set Counts
z_max refers to the most proximal surface, z_min the most distal. Whole is the volume with the cortex, Trabecular is after the shell is removed.

T160435 has more nodes in both the top and bottom node sets in the whole volume, though the difference is greater on the bottom, more distal face (Figure 33). The trabecular volumes show more contrast between the specimen, with T160435 containing more nodes on the bottom face and less on the top face than T160416. This is an indicator of the comparative "hollowness" of the structure approaching the more proximal end of the test section.

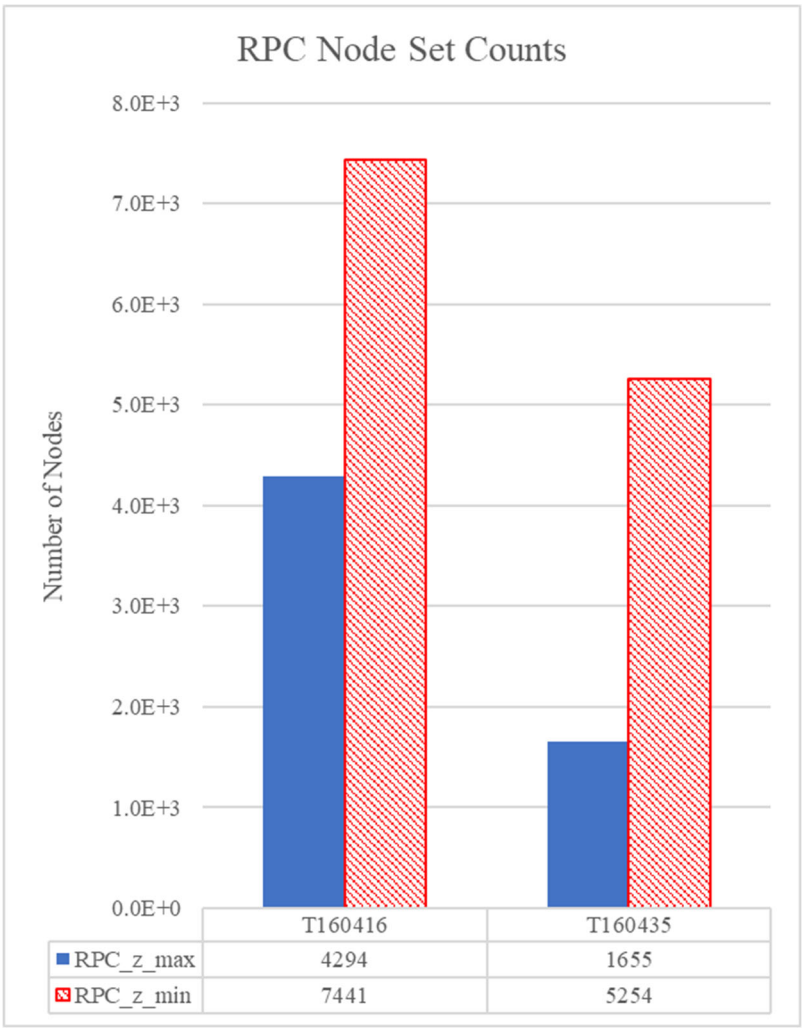


Figure 34. RPC Contact Area Node Set Counts
Contact area for the platen used in reduced platen compression (RPC) simulations (Figure 7, Figure 25)
RPC_z_max refers to the most proximal surface, RPC_z_min the most distal.

Figure 34 shows that T160416 contains substantially more nodes inside the platen contact region of the simulated RPC test, both on the top and bottom faces.

4.3 Reaction Forces

This section presents the reaction force (RF) results from the FEA simulations. The results for both whole specimen compression (WSC) and reduced platen compression (RPC) simulations are presented for three different volumes. With Cortex refers to the simulation results for the entire scanned specimen. Trabecular Only refers to the simulation results for the trabecular volume generated during image processing. The Reduced Trabecular results are the element values from the With Cortex simulation, but they have been reduced in post-processing to include only those elements which are present in the Trabecular Only volume. This allows for strain distributions for different loading cases to be compared on volumes with the same number of elements. The different volumes presented in the RF results are illustrated in Figure 35.

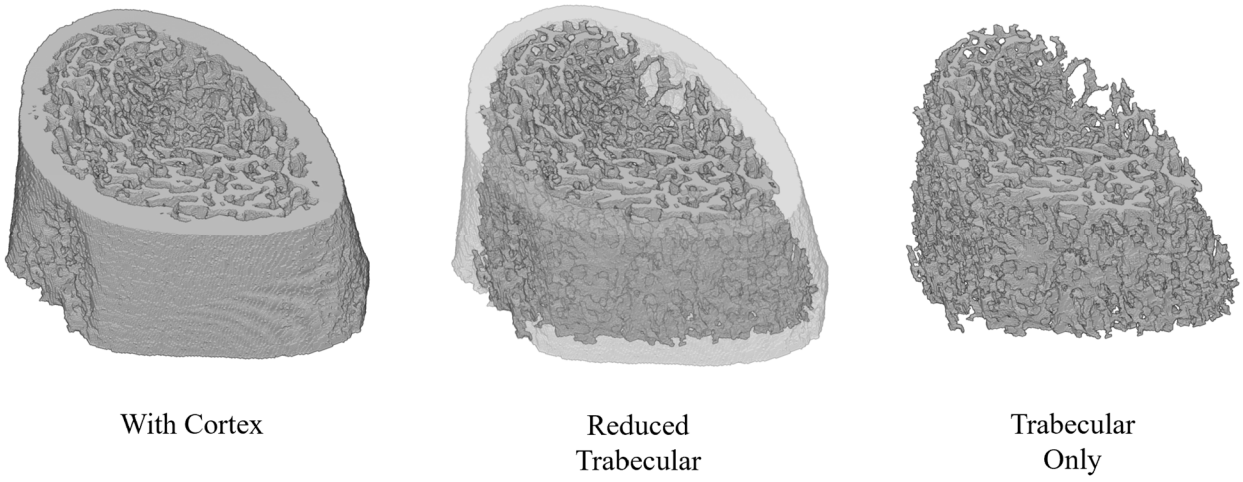


Figure 35. Visualization of Volumes Presented in Reaction Force Results
Reduced Trabecular (RT) results are the element values from the With Cortex (WC) simulations, but only for the elements which are present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations. TO volume is generated using an image opening process to remove the cortex (Figure 20). Volume shown is a 2mm thick section of the rat distal femur metaphysis, the top surface being the most proximal.

The total reaction forces from each simulation are presented in Table 2. The rough simulations produced both transverse (x and y) and normal (z) reaction forces. As there were no boundary conditions (BC) imposed on the x and y displacement in the smooth simulations, only the normal reaction force is present. The simulations produced equal and opposite pairs of one positive and one negative force value on either the top or bottom face. In order to compare forces between specimens, all values are presented as positive in Table 2. The full original force results are available in Appendix A. More detailed comparisons of the results in Table 2 are presented in figures throughout the rest of this section. With the linear analysis in the present study, these reaction forces are estimates of yield forces, not maximum loads.

Table 2. Simulated Reaction Force Results (N).

T160416	BC	Top X	Bottom X	Top Y	Bottom Y	Top Z	Bottom Z
With Cortex	WR	8.61E+01	8.61E+01	1.33E+02	1.33E+02	3.59E+03	3.59E+03
	WS	0.00E+00	4.30E-04	0.00E+00	1.89E-04	3.15E+03	3.15E+03
	RR	4.02E+00	4.02E+00	6.34E+00	6.34E+00	1.02E+02	1.02E+02
	RS	0.00E+00	6.74E-08	0.00E+00	4.42E-08	8.95E+01	8.95E+01
Reduced Trab	WR	2.41E+01	2.90E+01	3.40E+01	4.56E+01	6.16E+02	1.13E+03
	WS	0.00E+00	4.30E-04	0.00E+00	1.89E-04	5.12E+02	1.20E+03
	RR	4.02E+00	4.02E+00	6.34E+00	6.34E+00	1.02E+02	1.02E+02
	RS	0.00E+00	6.74E-08	0.00E+00	4.42E-08	8.95E+01	8.95E+01
Trabecular Only	WR	1.30E+01	1.30E+01	3.25E+01	3.25E+01	5.35E+02	5.35E+02
	WS	0.00E+00	1.03E-05	0.00E+00	2.23E-05	4.67E+02	4.67E+02
	RR	3.64E+00	3.64E+00	5.60E+00	5.60E+00	9.28E+01	9.28E+01
	RS	0.00E+00	2.26E-06	0.00E+00	1.11E-06	8.33E+01	8.33E+01
T160435	**BC**	**Top X**	**Bottom X**	**Top Y**	**Bottom Y**	**Top Z**	**Bottom Z**
With Cortex	WR	7.65E+01	7.65E+01	3.54E+02	3.54E+02	3.83E+03	3.83E+03
	WS	0.00E+00	2.03E-04	0.00E+00	1.54E-04	3.18E+03	3.18E+03
	RR	4.82E+00	4.82E+00	1.35E+00	1.35E+00	4.89E+01	4.89E+01
	RS	0.00E+00	6.00E-09	0.00E+00	1.69E-08	3.97E+01	3.97E+01
Reduced Trab	WR	1.58E+01	3.65E+00	5.64E+01	1.40E+02	4.33E+02	1.18E+03
	WS	0.00E+00	2.03E-04	0.00E+00	1.54E-04	3.13E+02	1.21E+03
	RR	4.82E+00	4.82E+00	1.35E+00	1.35E+00	4.89E+01	4.89E+01
	RS	0.00E+00	6.00E-09	0.00E+00	1.69E-08	3.97E+01	3.97E+01
Trabecular Only	WR	1.88E+01	1.88E+01	6.68E+01	6.68E+01	4.90E+02	4.90E+02
	WS	0.00E+00	5.23E-06	0.00E+00	1.13E-06	3.95E+02	3.95E+02
	RR	3.99E+00	3.99E+00	1.38E+00	1.38E+00	4.46E+01	4.46E+01
	RS	0.00E+00	2.48E-09	0.00E+00	1.13E-09	3.75E+01	3.75E+01

Units of presented results are in N. For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth, WR = WSC Rough, WS = WSC Smooth. Each of these is simulated separately for every volume. Rough means the displacement is fixed in the x and y, with applied displacement in the z direction. Smooth means applied displacement in the z direction, and free to move in the x and y. Top refers to the most proximal surface, bottom refers to the most distal surface. Z direction is along major axis of femur and is the applied loading direction. X and Y direction are transverse to the cross section. Reduced Trabecular (RT) results are the values from the With Cortex (WC) simulations, but only for the elements which are also present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations.

One observation from Table 2 is the relative magnitude of the normal and transverse reaction forces. When present, the transverse forces are a full order of magnitude lower than any normal reactions.

4.3.1 Transverse Reaction Forces in Rough Compression Simulations

The magnitudes of the transverse reaction forces in the rough simulations are shown in Figure 36 and Figure 37. There were no transverse loads in the smooth simulations. Since the specimens do not share common axes, there is no value in comparing x and y or positive and negative reactions; therefore, the magnitudes of the total transverse reaction force are presented here.

As would be expected for the simulation results (i.e., excluding Reduced Trabecular), the reaction forces on the top and bottom surface are in equilibrium (Figure 36). The Reduced Trabecular results are not in equilibrium, with the bottom trabecular face having a higher reaction force than the top trabecular face for both specimens. This is because the top (more proximal) end has a higher percentage of cortical tissue, which is removed in the case of the Reduced Trabecular results. In general, T160435 has a larger transverse reaction force, indicating the specimen is less symmetric or has a structural orientation not well aligned with the axis of loading.

In Figure 37, T160416 has the larger transverse reaction force in the simulated RPC test, differing from the WSC test before. The Reduced Trabecular results are identical to the With Cortex since the loaded region in the RPC simulations are entirely within the trabecular structure. The Trabecular Only models have a 10-15% lower transverse reaction force, suggesting some minor influence from the cortical tissue.

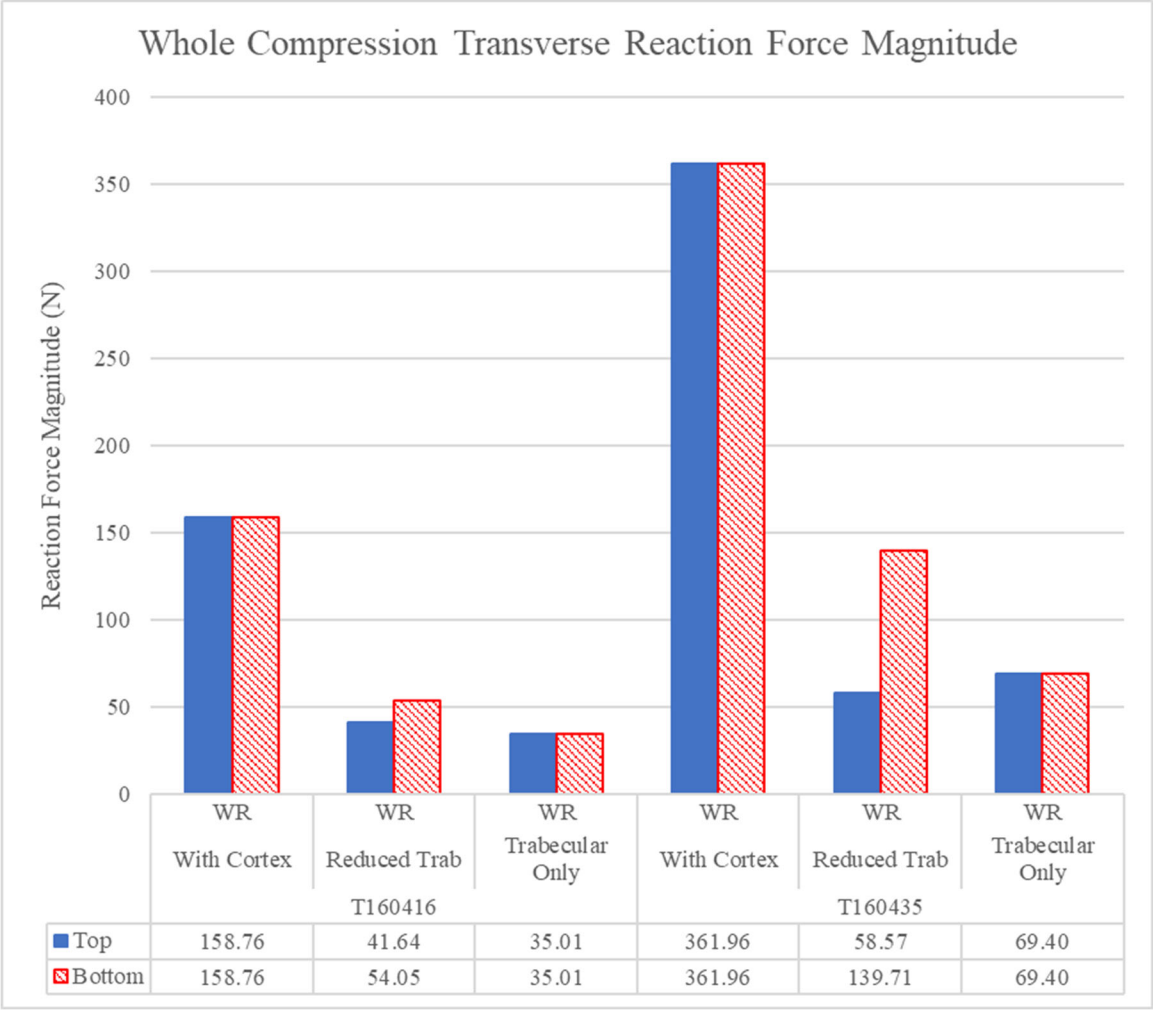


Figure 36. Transverse Reaction Force (RF) Magnitude in Whole Specimen Compression (WSC) Simulations. WR = WSC Rough. Top refers to the most proximal surface, bottom refers to the most distal surface. Transverse RF is defined as the magnitude of the total force in the x and y direction. Reduced Trabecular (RT) results are the values from the With Cortex (WC) simulations, but only for the elements which are also present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations.

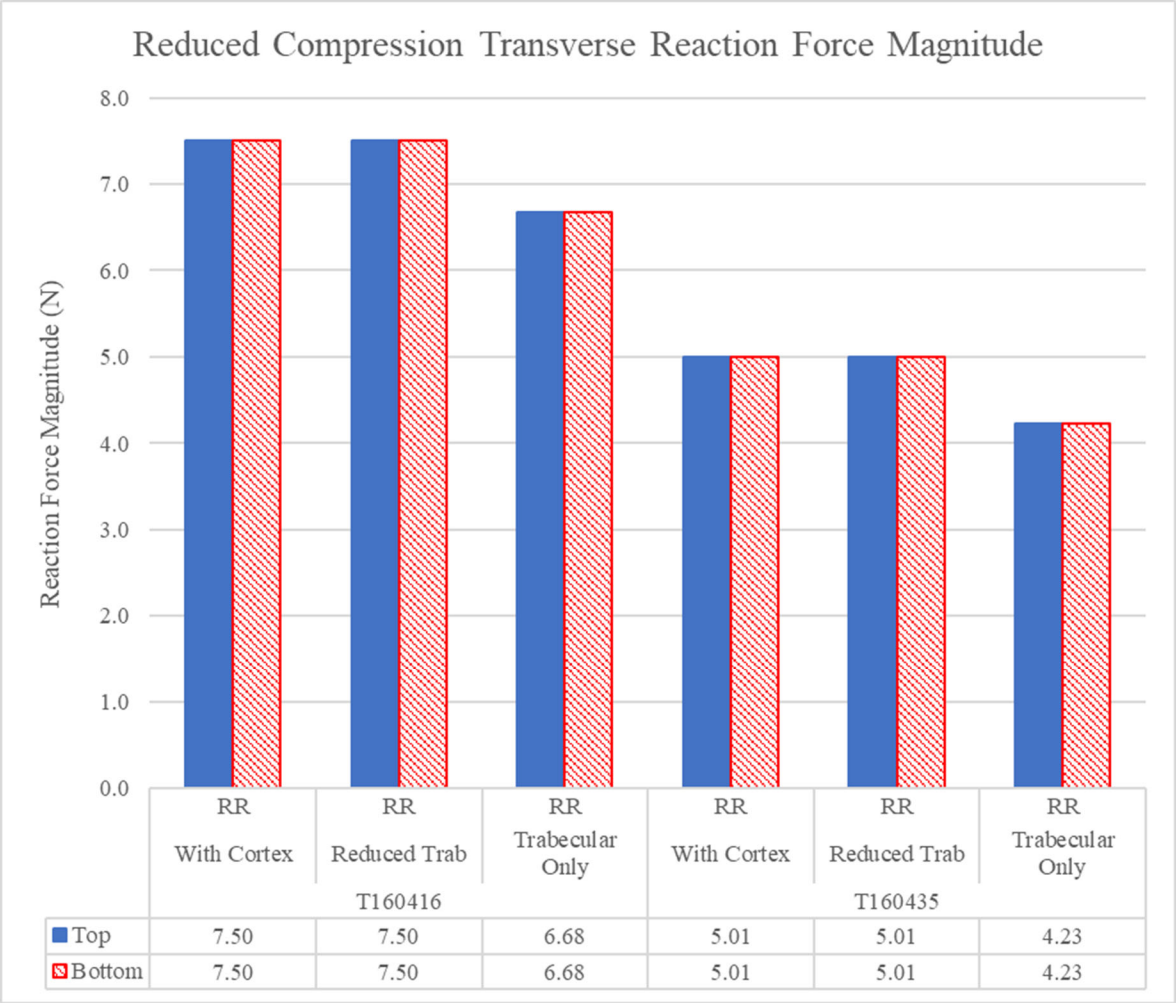


Figure 37. Transverse Reaction Force (RF) Magnitude in Reduced Platen Compression (RPC) Simulations
RR = RPC Rough. Top refers to the most proximal surface, bottom refers to the most distal surface. Transverse RF is defined as the magnitude of the total force in the x and y direction. Reduced Trabecular (RT) results are the values from the With Cortex (WC) simulations, but only for the elements which are also present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations.

4.3.2 Normal Reaction Forces

The normal reaction forces in the With Cortex and Trabecular Only simulations are shown in Figure 38. This does not show a large difference between specimens, but the loading methods can be compared. The amount of force required to produce 5% strain in the WSC simulation with the cortex is an order of magnitude greater than with only the trabecular structure, which is in turn an order of magnitude greater than the RPC simulations. More detailed comparisons are made in the following figures.

The results of the WSC simulations are presented in Figure 39. The reaction force for T160416 is found to be 6% less than for T160435 in the WR With Cortex simulations, with no appreciable difference in the WS version. For the Trabecular Only, the roles are reversed with T160416 being 10-15% larger than T160435 in both the WR and WS simulations. From the Reduced Trabecular results, the Top Z values for T160435 are smaller despite being larger in the With Cortex simulations, suggesting that the cortical tissue is playing a stronger role in load bearing for T160435 than T160416. The same trend is not seen in the Bottom Z (more distal) faces.

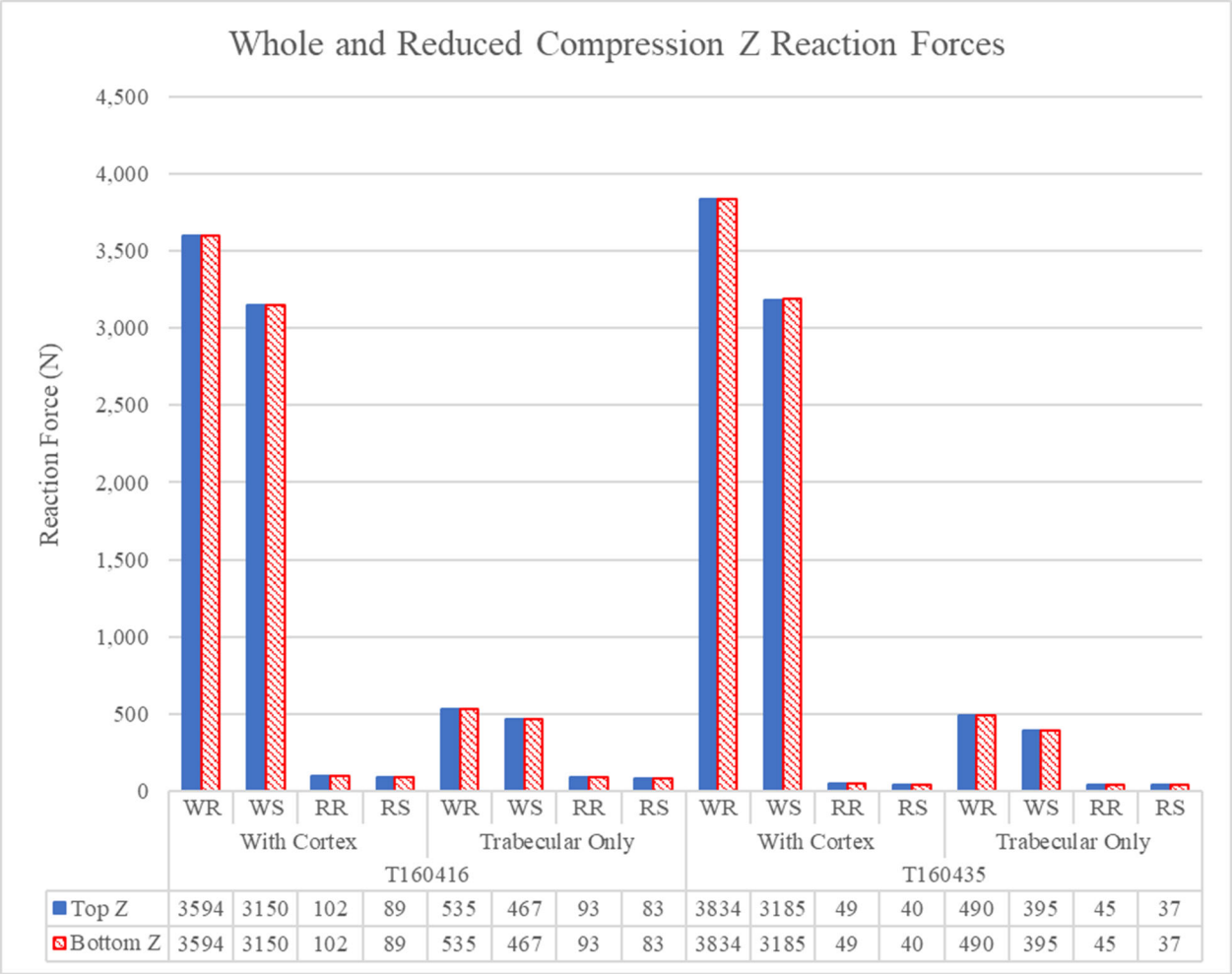


Figure 38. Normal Reaction Force in Whole and Reduced Compression Simulations

For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth, WR = WSC Rough, WS = WSC Smooth. Top refers to the most proximal surface, bottom refers to the most distal surface. Z direction is along major axis of femur and is the applied loading direction.

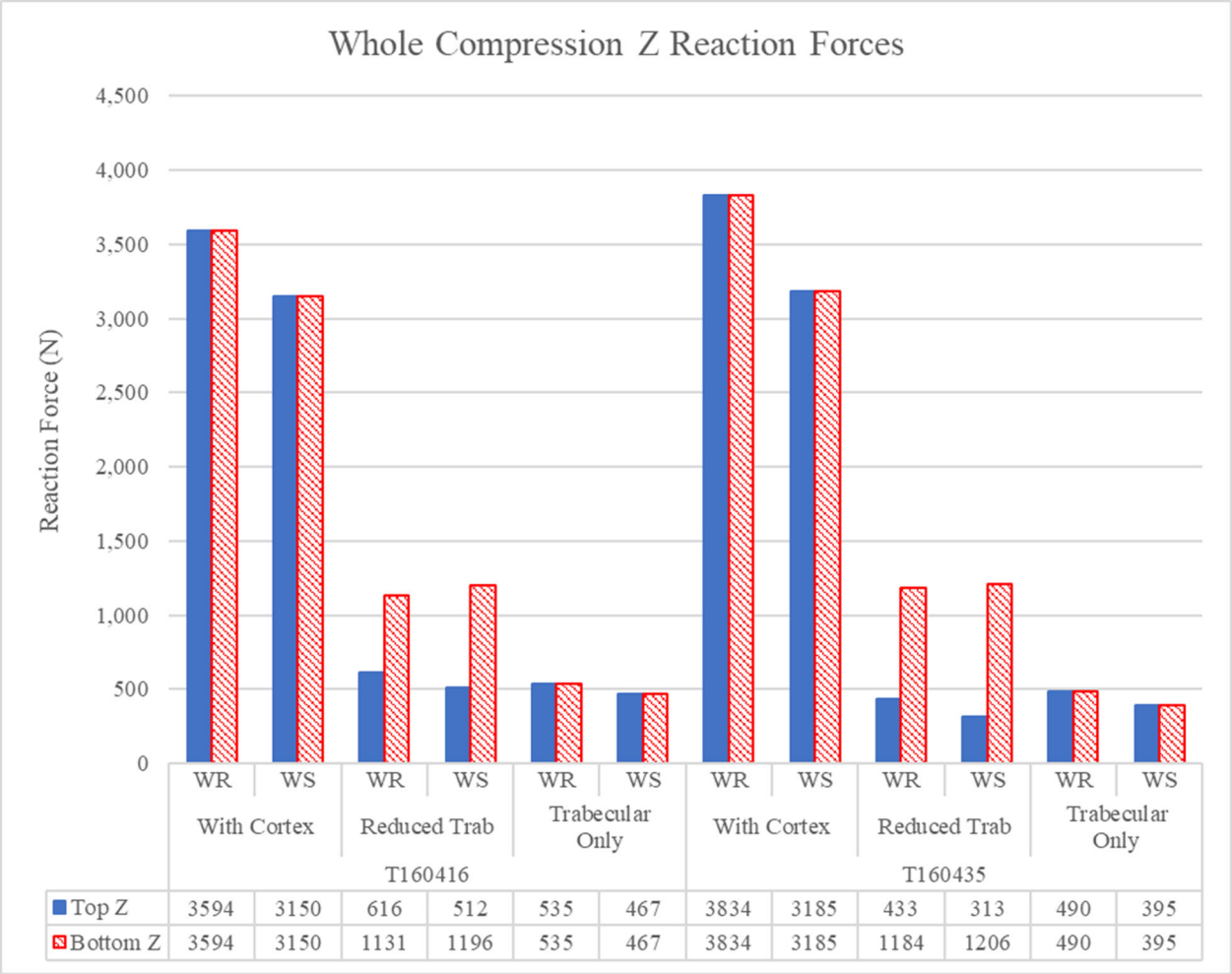
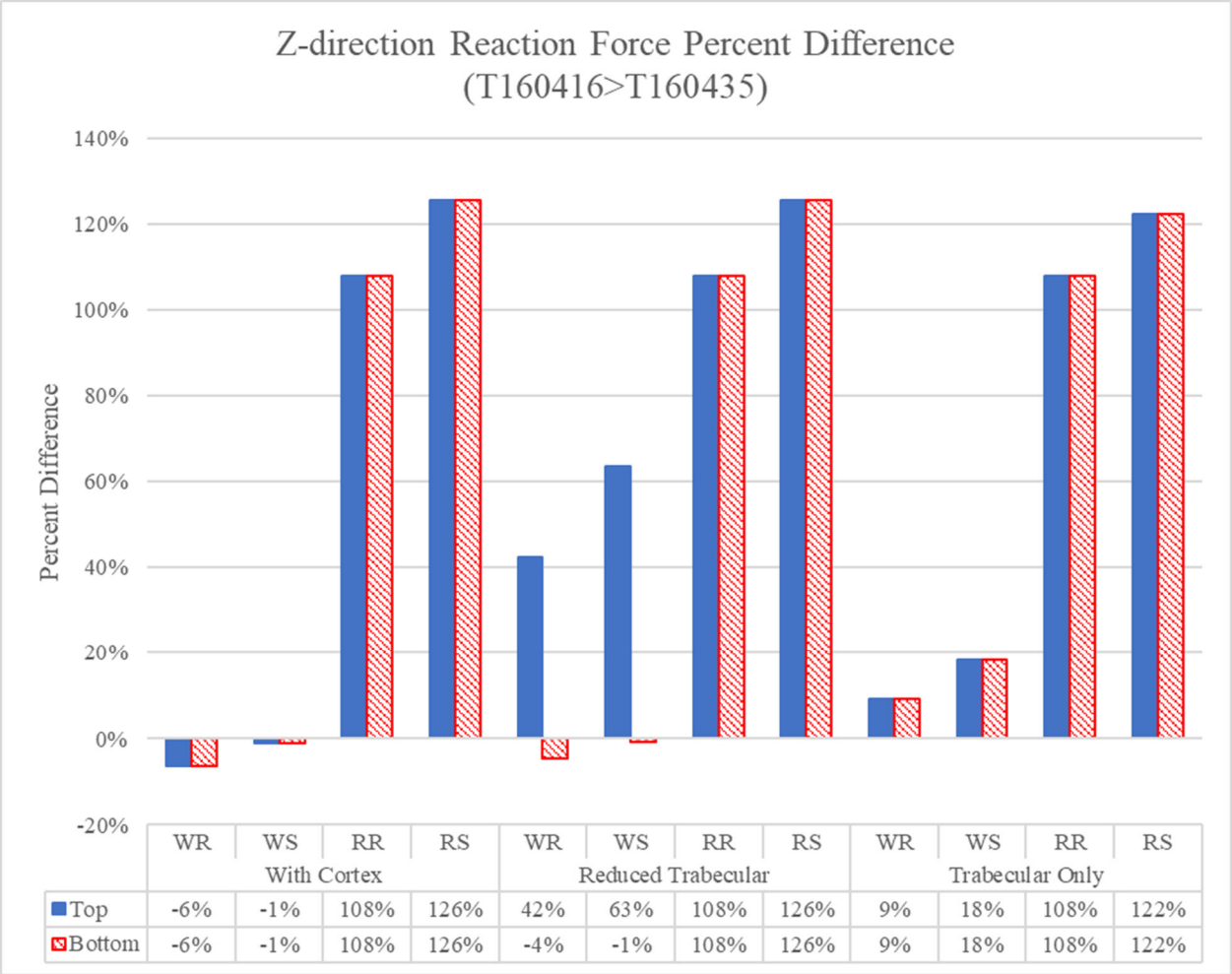


Figure 39. Comparison of Normal Reaction Force in Whole Specimen Compression Simulations.
For whole specimen compression (WSC) the applied boundary conditions (BC) are: WR = WSC Rough, WS = WSC Smooth. Top refers to the most proximal surface, bottom refers to the most distal surface. Z direction is along major axis of femur and is the applied loading direction. Reduced Trabecular (RT) results are the values from the With Cortex (WC) simulations, but only for the elements which are also present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations.

Figure 40. Comparison of Normal Reaction Force (RF) in Reduced Platen Compression (RPC) Simulations. For reduced platen compression (RPC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth. Top refers to the most proximal surface, bottom refers to the most distal surface. Z direction is along major axis of femur and is the applied loading direction.

The normal reaction force results for the RPC simulations are shown in Figure 40. The Reduced Trabecular results are not included as they are identical to the With Cortex values (similar to Figure 37). Across the board, the RS simulations produce less force than their RR counterparts. The rough simulation increases the stiffness of the specimen in the z-direction by having the x and y displacements constrained, even though only 6 of the 101 node layers have applied BCs. For each simulation, T160435 bears approximately half the load of T160416.



	WR	WS	RR	RS	WR	WS	RR	RS	WR	WS	RR	RS
	With Cortex				Reduced Trabecular				Trabecular Only			
Top	-6%	-1%	108%	126%	42%	63%	108%	126%	9%	18%	108%	122%
Bottom	-6%	-1%	108%	126%	-4%	-1%	108%	126%	9%	18%	108%	122%

Figure 41. Normal Reaction Force Percent Difference Between Specimens.
For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth, WR = WSC Rough, WS = WSC Smooth. Top refers to the most proximal surface, bottom refers to the most distal surface. Z direction is along major axis of femur and is the applied loading direction. Reduced Trabecular (RT) results are the values from the With Cortex (WC) simulations, but only for the elements which are also present in the Trabecular Only (TO) structure. RT volume is not an actual simulation, just a reduction of results in post processing. WC and TO are the volumes used in simulations.

Figure 41 provides a summary for all the simulations, comparing the two specimens using the percent difference in reaction force. The WSC results either show little difference or minimally favor T160416 in the case of Trabecular Only. The RPC simulations for all volumes show the reaction forces for T160416 to be more than 100% greater than those for T160435.

4.3.3 Past Mechanical Testing Results

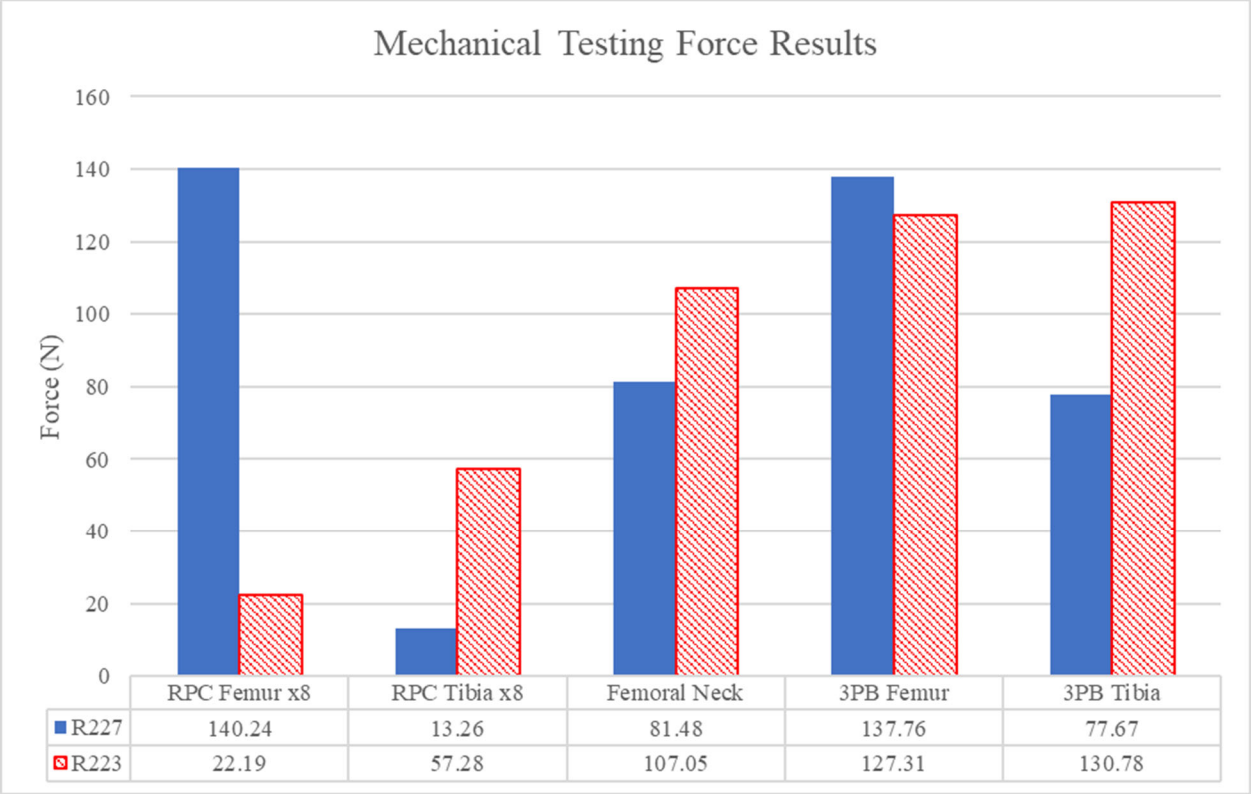


Figure 42. Mechanical Testing Force Results.
Mechanical testing results for the two animals used in this study for reduced platen compression (RPC), femoral neck, and three-point bending (3PB). All results are yield force, except femoral neck which is maximum load. See section 2.3 in the background for information on these tests. RPC test results are multiplied by a factor of 8 for visual comparison with the other mechanical tests. Mechanical data is from past theses [31-33].

For comparison with the FEA simulation reaction forces, the past mechanical testing results for the two animals in this study are presented in Figure 42. All the results show R223 being stronger, at least in the sense of generating a larger force, except for the three-point bending femur results, which do not show a significant difference. The RPC femur test trends the opposite way, with R227 sustaining far more load than R223.

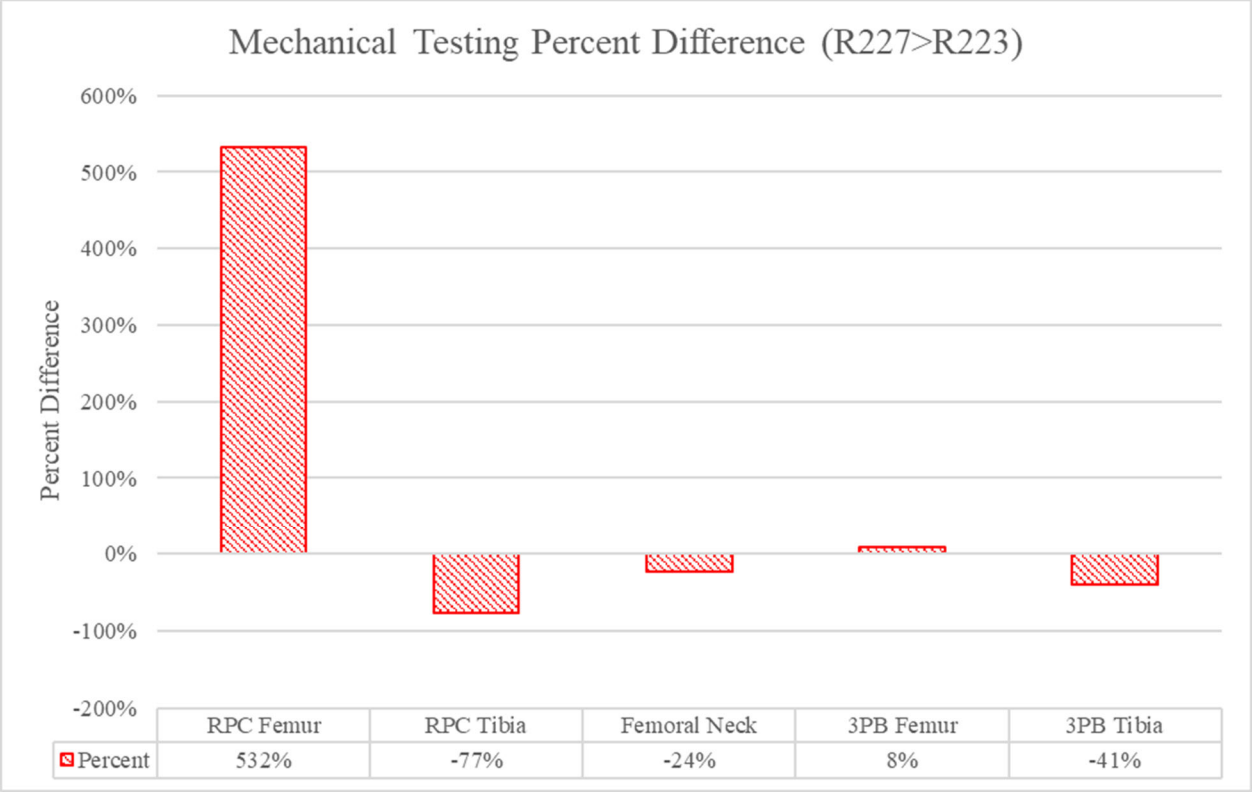


Figure 43. Mechanical Testing Percent Difference Between Specimens
Mechanical testing results for reduced platen compression (RPC), femoral neck, and three-point bending (3PB). See section 2.3 in the background for information on these tests.

Figure 43 shows the mechanical testing results summarized as percent differences, allowing for direct comparison with the FEA simulation results in Figure 41.

For complete comparison between the actual RPC mechanical tests and FEA simulations, the discrepancies in platen size need to be considered. Figure 44 reports these platen values; the circles shown are to scale. R227 used a slightly large platen than was computed in the FEA pre-processing, bringing in the possibility that some cortical tissue was inadvertently loaded during mechanical testing. R223 used an appreciably smaller platen than the corresponding simulation. For the comparatively "hollow" specimen in the case of R223/T160435 (Figure 31), the even smaller actual platen would exacerbate the issue, loading less tissue than the FEA simulation.

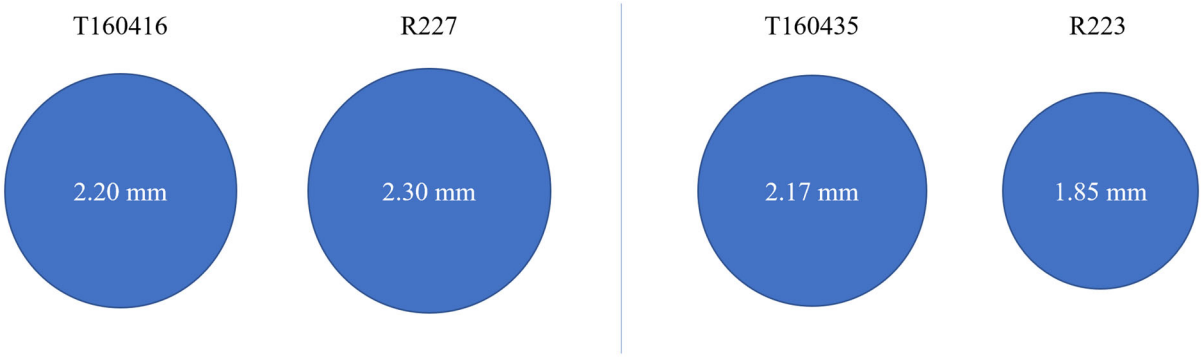


Figure 44. Comparison of Simulated and Real Platen Sizing.
T160416 and T160435 are the simulated platen sizes as determined by the image processing techniques for FEA boundary conditions. R227 and R223 are the corresponding real platen sizes used in mechanical reduced platen compression (RPC) testing. Circles are to scale.

4.4 Contour Plots

While the reaction force results from the FEA allow comparison with mechanical testing data, a more detailed diagnostic review of the results is presented in this section using the values of strain in each element. The following contour plots visually illustrate the regions in the specimens where high values of strain occur, indicating potential zones of failure. The contour plots were generated using fixed values for the contour levels, with values greater than the applied strain of +5% (tension zones) being highlighted in red, values between +5% and -5% in green, and values beyond -5% (compression zones) in blue. Occasionally the simulations produced a strain valued in an element exceeding 100%, which was arbitrarily chosen as the upper and lower limits for the contours. These elements are rendered as gray or black, but there are so few elements with these extreme values that they cannot be found with visual inspection. The legends next to each contour plot indicate whether elements in the image exceed the applied bounds. All contour plots herein are on the undeformed volume. To aid in the viewing of internal elements, contour plots with free body cuts are presented; these illustrate only the rear half of the specimen so internal elements can be viewed. For aid in visualizing the free body cut, Figure 45 is provided aligning the full and cut volume.

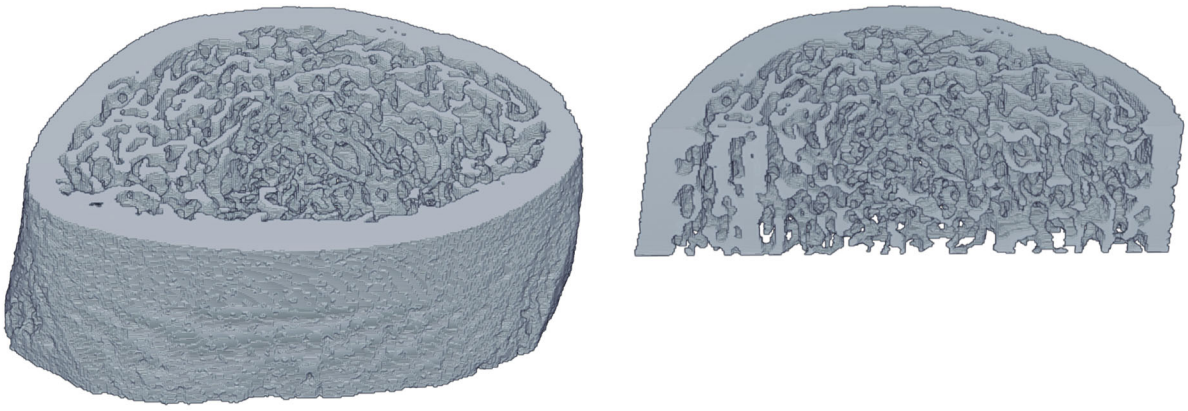


Figure 45. Illustration of Volumes for Contour Plots.
Left is the full volume from the simulations. Right is the rear half of the original volume that has been exposed in order to view internal elements.

Figure 46 shows the free body cut contour plots for each of the WSC simulations performed on T160416. The relative loading in the cortex of the specimen compared to trabecular volume is evident in the large regions of tissue in compression (blue). Once far enough away from the cortex, the internal trabecular structure behaves similarly in both volumes in that the regions of failure are largely the same. The WS simulations produce more tension than WR, and the most tension is present in the Trabecular WS.

Figure 47 shows the free body cut contour plots for each of the RPC simulations performed for T160416. The strain zones are mostly the same in all scenarios. Slightly more tension is observed in the RS Whole than the RR Whole, but the difference is much less noticeable than in the WS and WR load cases. If the same areas in the WSC and RPC simulations are compared, specifically those loaded during the RPC simulations, the strain zones are quite similar.

The free body cut contours cannot be used to compare between specimens since the cut planes do not occur in the same place. The whole specimen contours, however, do allow for limited visual comparison. Figure 48 shows the contour plots for the RS Whole loading scenario of both specimens. The large difference in the amount of tissue inside the RPC contact area is highlighted by the amount of tissue experiencing strain.

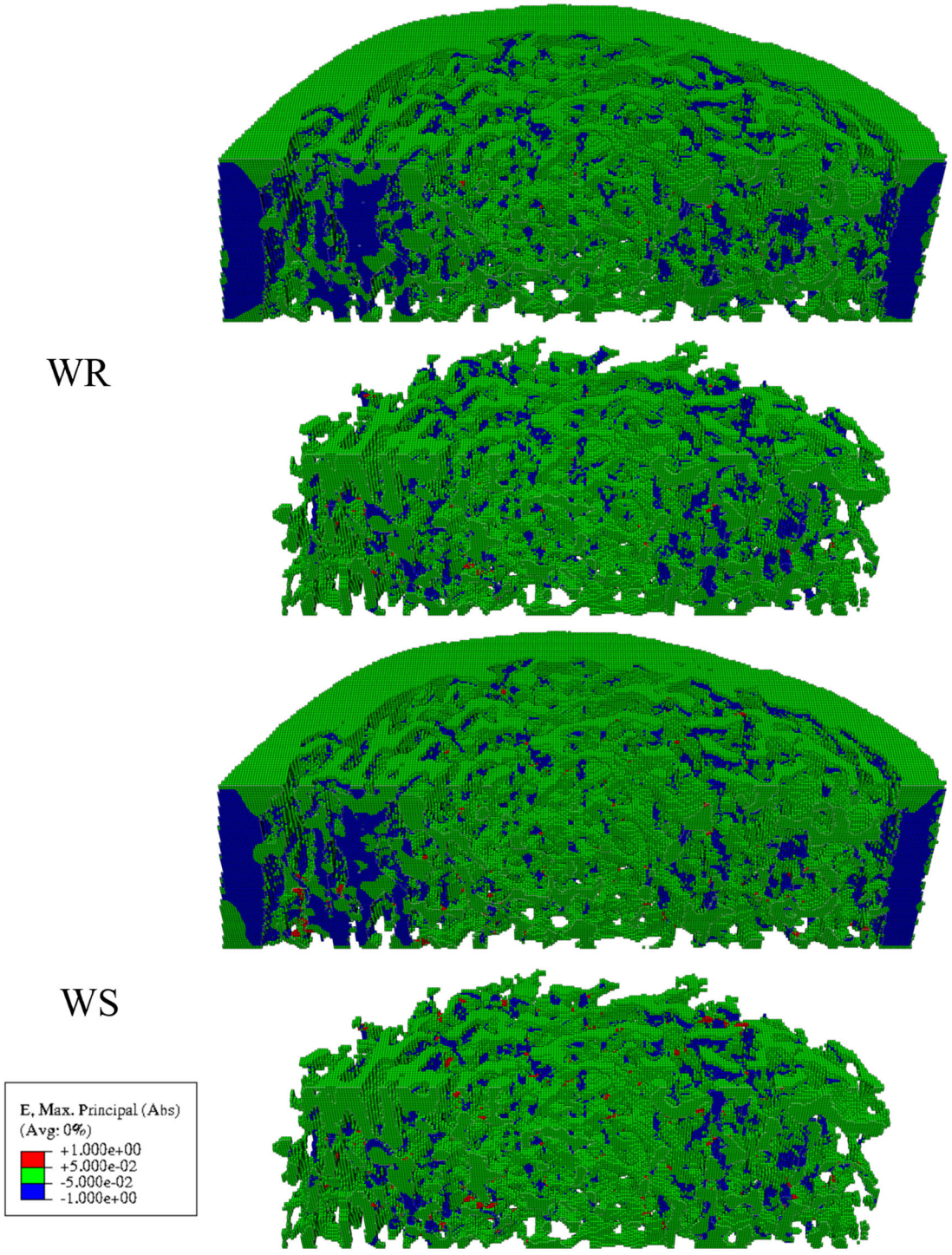


Figure 46. Free Body Cut Contour Plots for T160416 WR and WS

For whole specimen compression (WSC) the applied boundary conditions (BC) are: WR = WSC Rough, WS = WSC Smooth. Whole is the body with the cortex, Trabecular is without. Only half of the simulated volume is shown to expose internal elements. Red elements are >5% strain in tension, blue elements are >5% in compression, and green are anything between the two (5% is the applied strain). Plots are generated using the absolute maximum principal strain as calculated in ABAQUS.

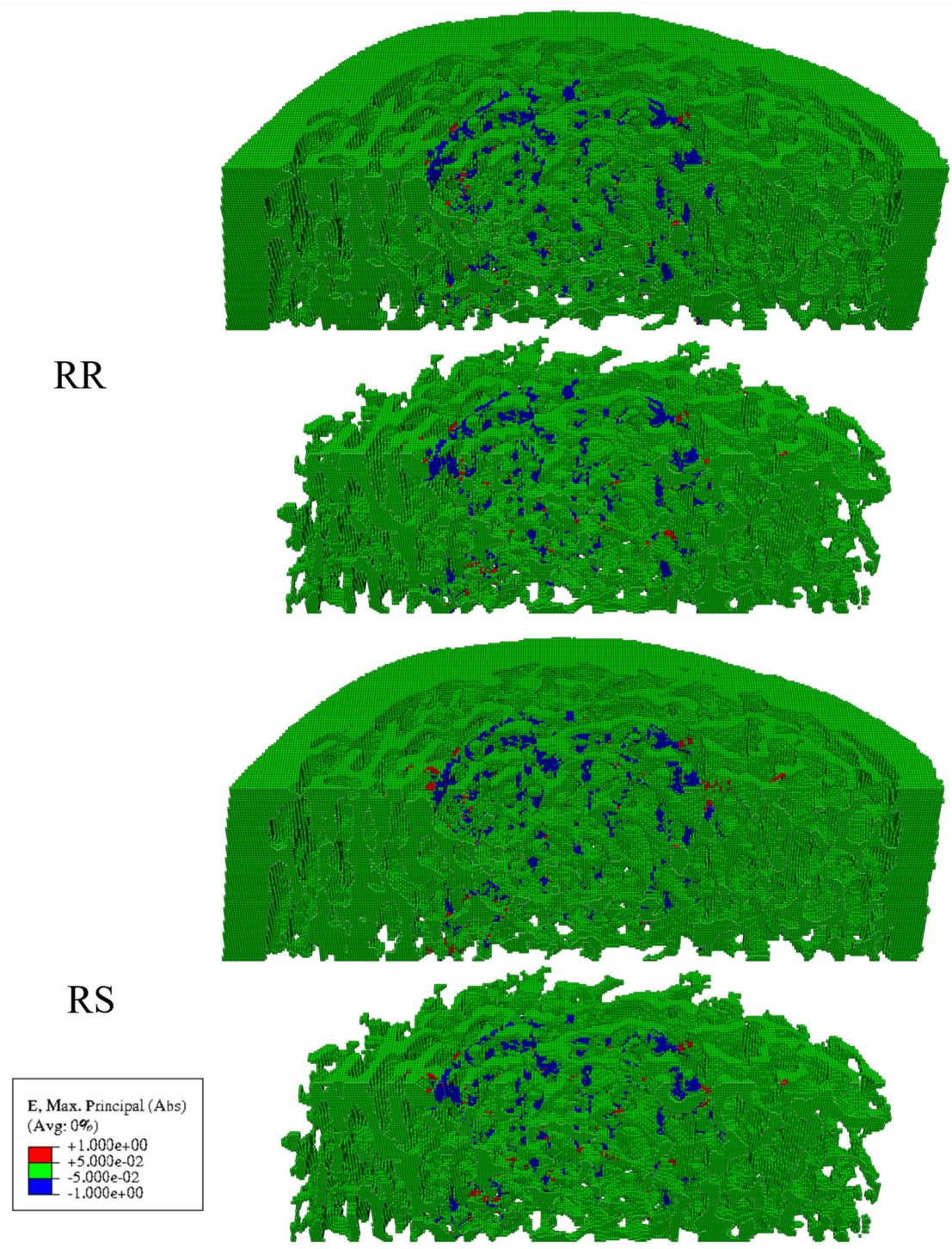


Figure 47. Free Body Cut Contour Plots for T160416 RR and RS

For reduced platen compression (RPC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth. Whole is the body with the cortex, Trabecular is without. Only half of the simulated volume is shown to expose internal elements. Red elements are >5% strain in tension, blue elements are >5% in compression, and green are anything between the two (5% is the applied strain). Plots are generated using the absolute maximum principal strain as calculated in ABAQUS.

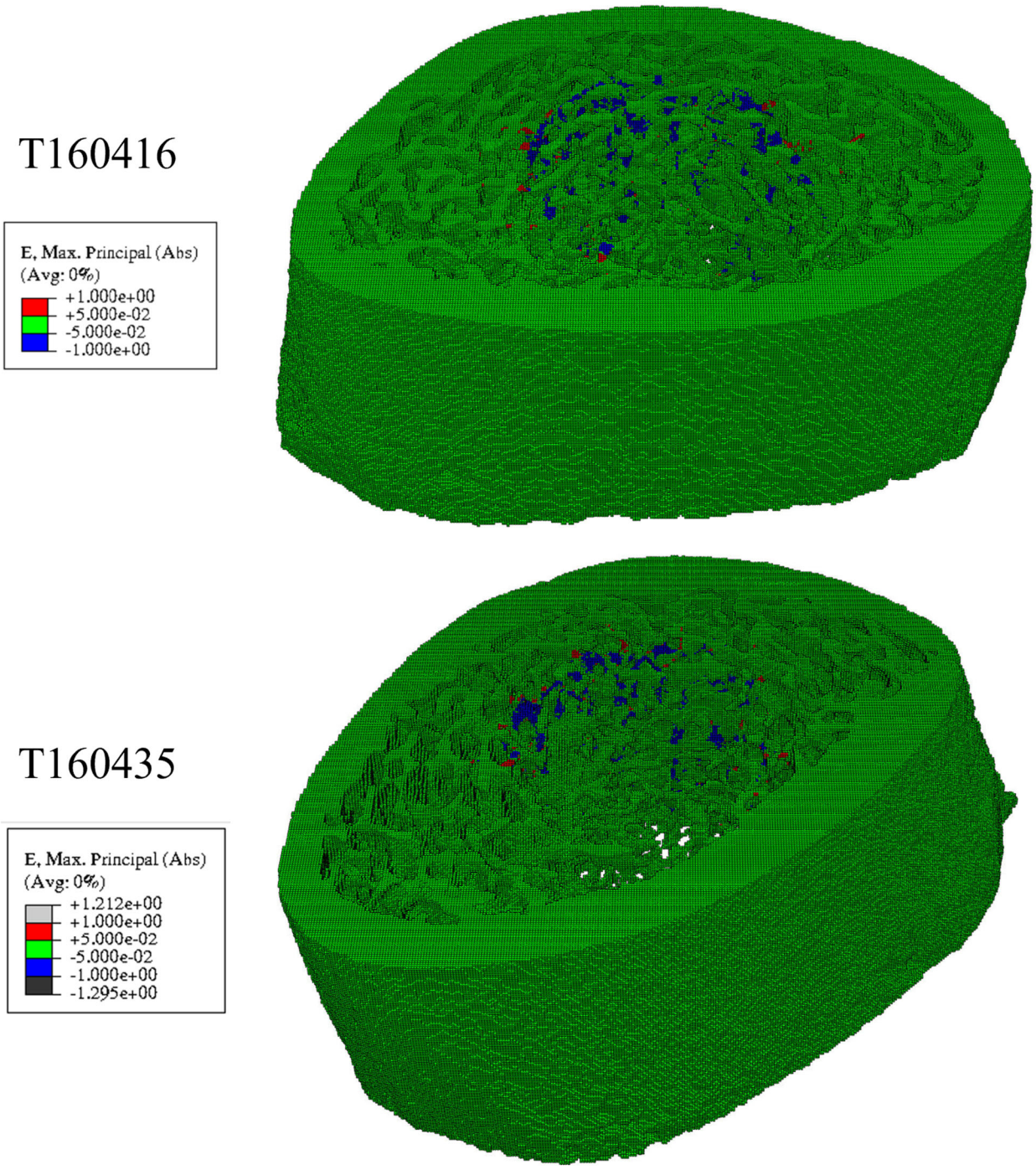


Figure 48. Contour Plots of T160416 and T160435 RS

For reduced platen compression (RPC) the applied boundary conditions (BC) is RS = RPC Smooth. Red elements are >5% strain in tension, blue elements are >5% in compression, and green are anything between the two (5% is the applied strain). Plots are generated using the absolute maximum principal strain as calculated in ABAQUS. Volume is a 2mm thick section of a rat distal femur metaphysis with the top surface being the most proximal.

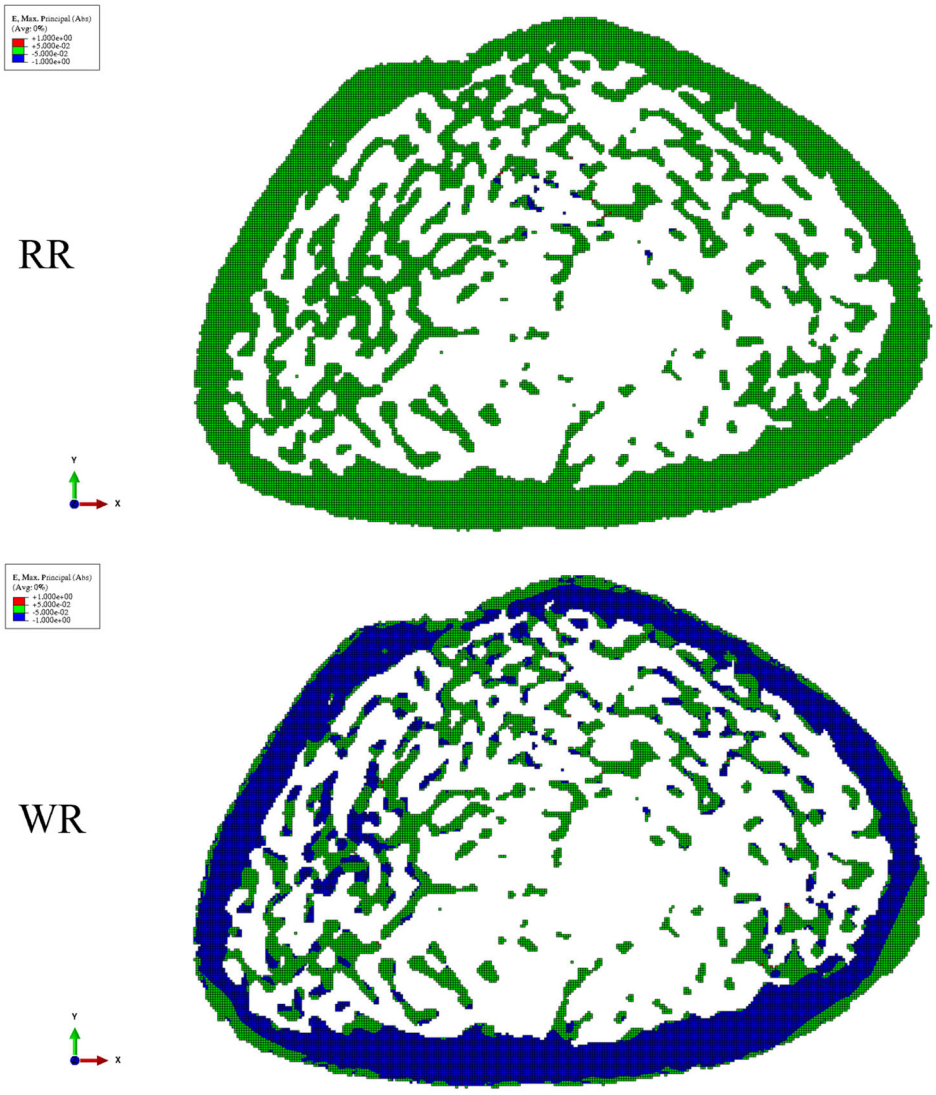


Figure 49. Z Cross Section Slice Contour Plots for T160416 RR and WR

For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, WR = WSC Rough. Red elements are >5% strain in tension, blue elements are >5% in compression, and green are anything between the two (5% is the applied strain). Image shown is a cross section of a rat distal femur metaphysis. Plots are generated using the absolute maximum principal strain as calculated in ABAQUS.

Figure 49 shows some of the cross-section contour plots generated for T160416. The large amount of compressive strain in the cortex and outer cancellous regions of the WR simulation is shown contrasted with the minimal trabecular loading of the RR simulation.

4.5 Effective Strain Distributions

The figures in this section present the effective strain calculated during post-processing. These distributions are illustrated using both histograms and CDFs. The strain distributions allow for comparison between different loading cases within a specimen as well as the behavior of the different specimens. The With Cortex (Whole) and Trabecular Only (Trabecular) simulation results, as well as the Reduced Trabecular (Reduced) values taken from the With Cortex simulation (Figure 35), are shown on the same plot for each load case. Note that the x-axis for the WR and WS plots range from 0 to 0.1 strain, and those on the RR and RS plots range from 0 to 0.05 strain. This was done so that differences in the RR and RS plots were more easily observed.

There are large peaks in the histograms for the Whole models of the WR and WS load scenarios, centered near the applied strain of 0.05. This is due to the large percentage of the elements in the cortex which experience that level of strain. The Trabecular models disperse the applied strain so that only a small percentage experience the 0.05 level. The CDFs in Figure 50 and Figure 51 show that 95% of the elements are at or below the 0.05 applied strain level for the Trabecular models. Looking at that same metric for the Whole models, ~60% and ~80% of the elements experience at or below 0.05 strain for the WR and WS load cases, respectively. The Reduced values are between that of the Whole and Trabecular distributions, displaying behavior similar the Trabecular structure but having a higher induced strain in some elements due to the influence of the cortical tissue. Also of note in Figure 50 and Figure 51 is the non-zero intercept of the CDFs for the WR case, where 5% of the elements are at zero strain. This is from the applied boundary condition (BC), where the top and bottom three layers of nodes have fixed displacements in all directions; therefore, the elements containing those nodes experience no

strain. This is not the case for WS, in which the BC nodes are free to move in the x and y-direction, allowing for plane strain. There is a large observable difference between the WR and WS load cases in the same specimen, but they do not show significant differences between two animals.

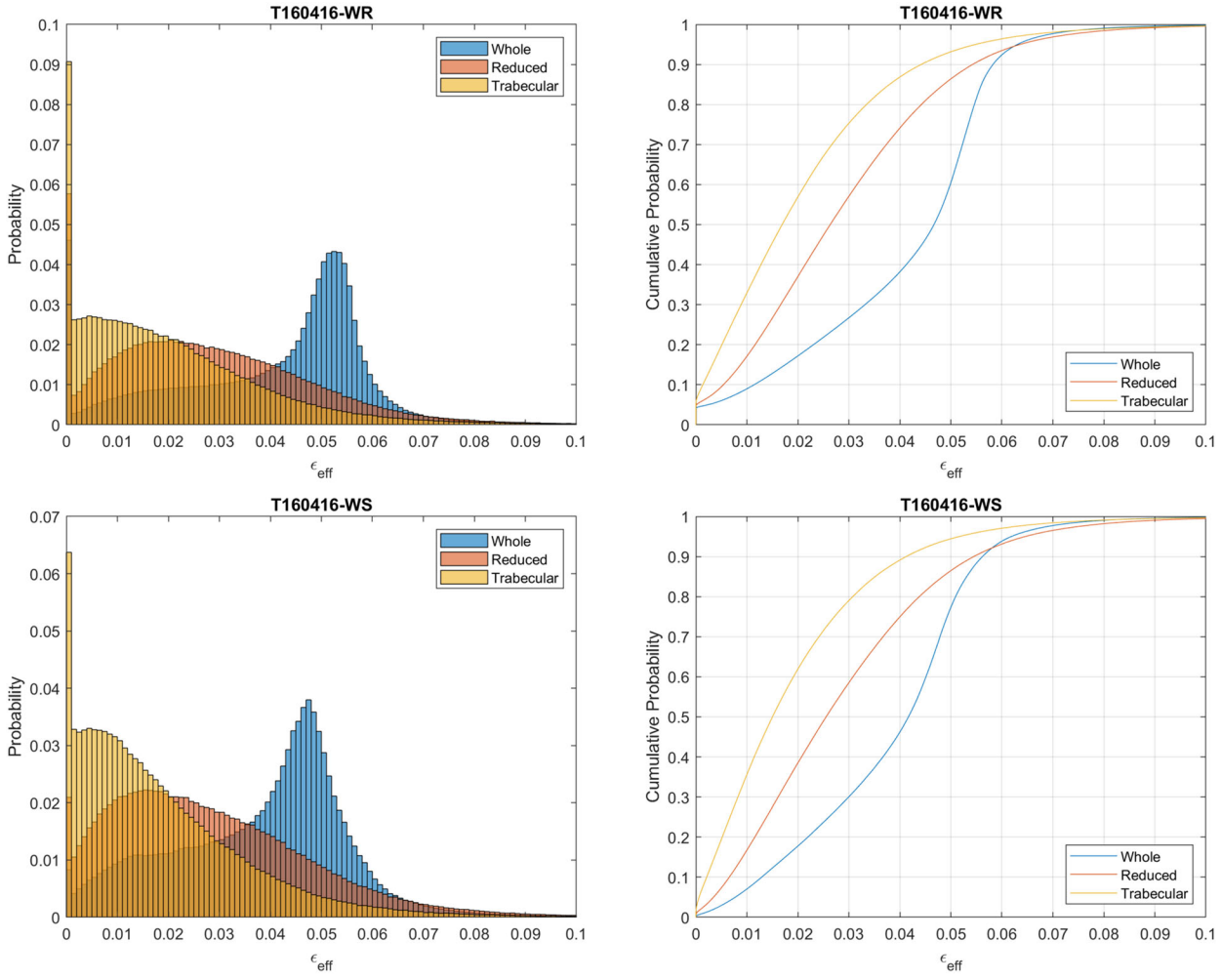


Figure 50. Histograms and CDFs for T160416 WR and WS

Histograms and cumulative density functions (CDFs) are generated using the effective strain calculated in MATLAB. For whole specimen compression (WSC) the applied boundary conditions (BC) are: WR = WSC Rough, WS = WSC Smooth. Reduced Trabecular (Reduced) results are the values from the With Cortex (Whole) simulations, but only for the elements which are also present in the Trabecular Only (Trabecular) structure. Reduced volume is not an actual simulation, just a reduction of results in post processing. Whole and Trabecular are the volumes used in simulations.

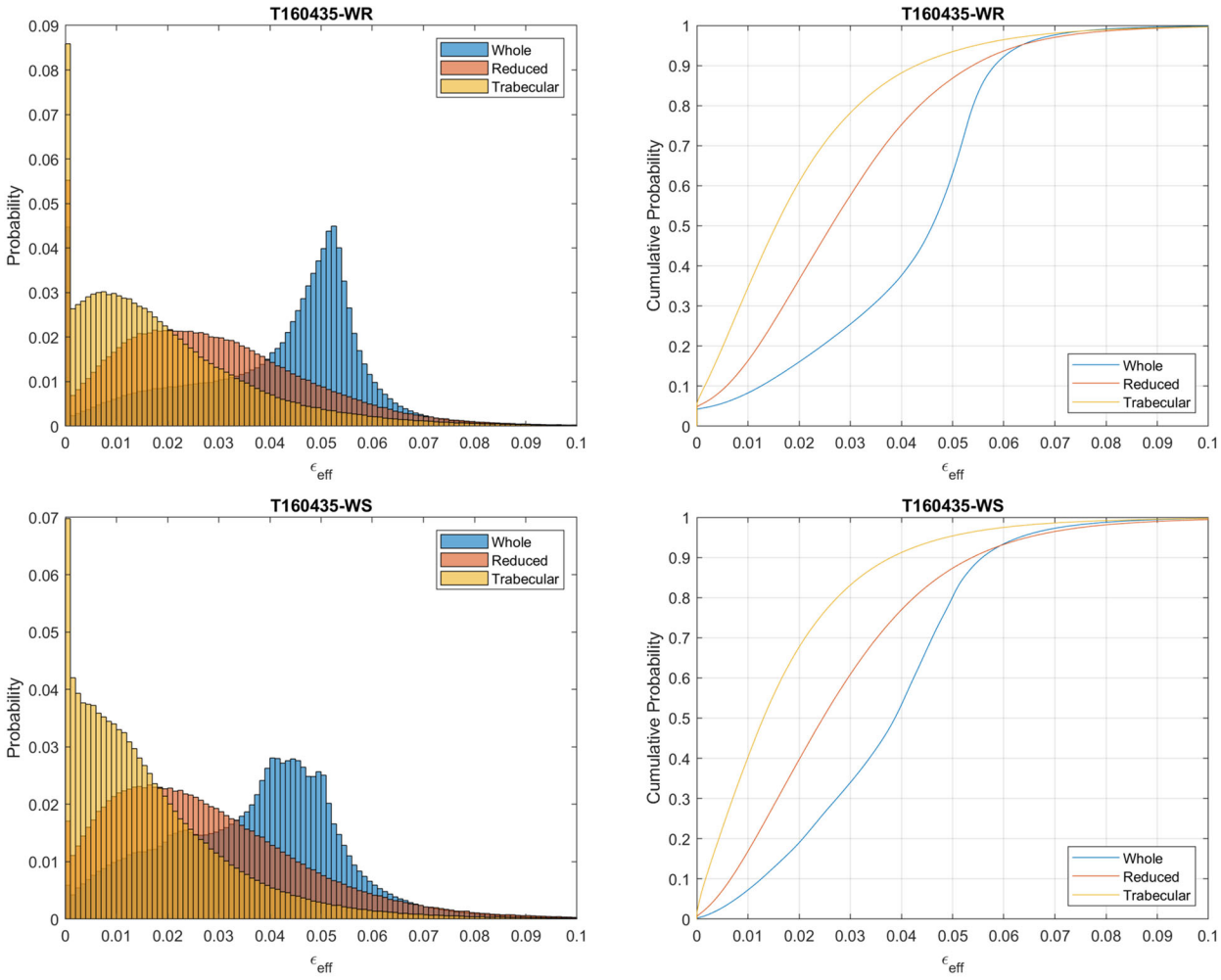


Figure 51. Histograms and CDFs for T160435 WR and WS

Histograms and cumulative density functions (CDFs) are generated using the effective strain calculated in MATLAB. For whole specimen compression (WSC) the applied boundary conditions (BC) are: WR = WSC Rough, WS = WSC Smooth. Reduced Trabecular (Reduced) results are the values from the With Cortex (Whole) simulations, but only for the elements which are also present in the Trabecular Only (Trabecular) structure. Reduced volume is not an actual simulation, just a reduction of results in post processing. Whole and Trabecular are the volumes used in simulations.

The histograms and CDFs are shaped much differently for the RR and RS load cases, since now only a small portion of the model is loaded by the BCs. In Figure 52 and Figure 53, all three distributions (Whole, Reduced, Trabecular) are of the same general shape. The Whole curve on each of the CDFs is skewed upward since it includes elements in the cortical tissue which remain unloaded in the RPC simulations. T160416 shows an observable but insignificant difference in the distribution for the Reduced and Trabecular models and T160435 has no observable difference between them. The RR and RS plots show little difference within the same specimen but show a significant difference between the two animals (in direct opposition to the earlier WR and WS distributions). For instance, T160416 has 85% at or below 0.01 strain for both RR and RS. The same 85% threshold would be 0.005 strain in T160435.

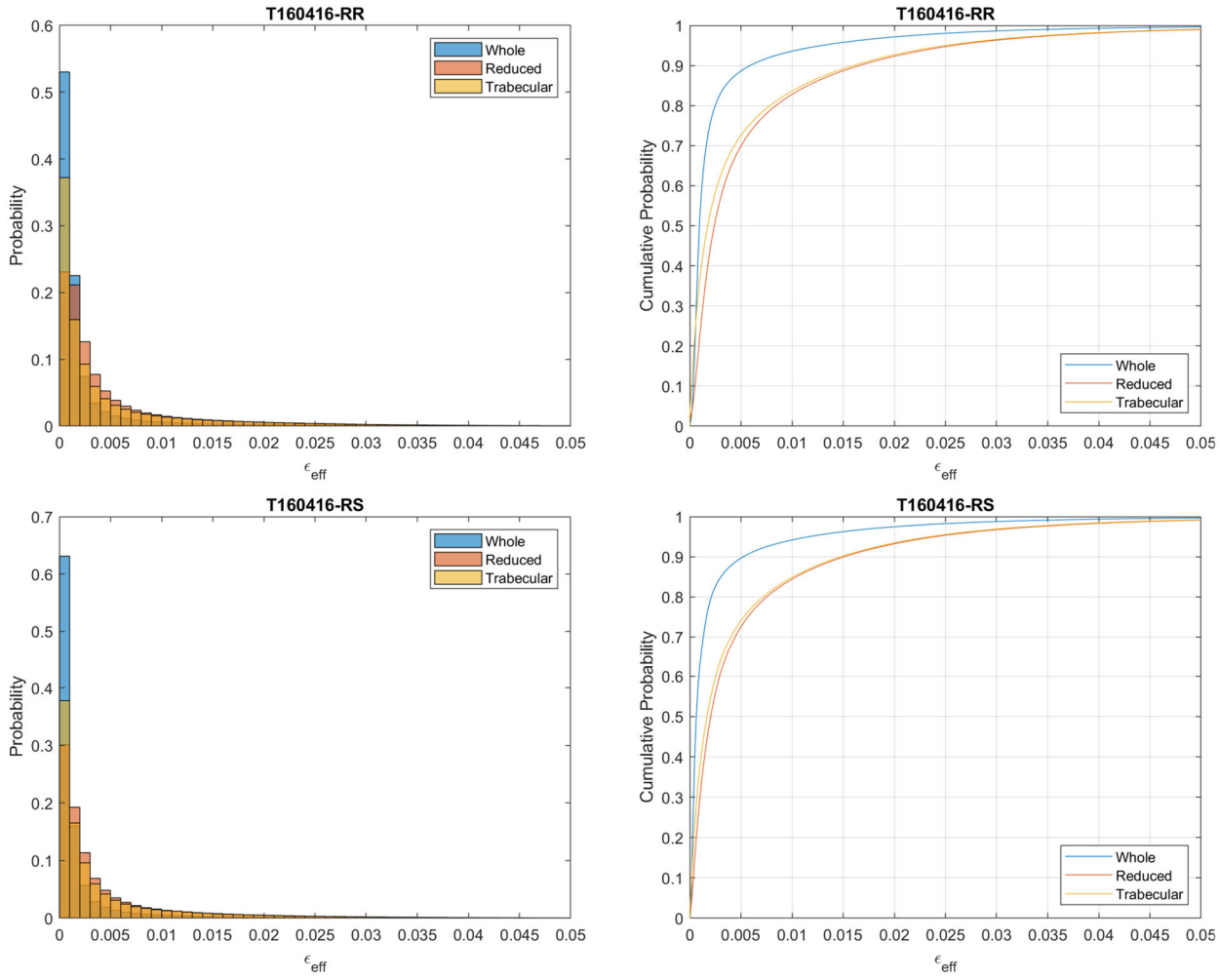


Figure 52. Histograms and CDFs for T160416 RR and RS.

Histograms and cumulative density functions (CDFs) are generated using the effective strain calculated in MATLAB. For reduced platen compression (RPC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth. Reduced Trabecular (Reduced) results are the values from the With Cortex (Whole) simulations, but only for the elements which are also present in the Trabecular Only (Trabecular) structure. Reduced volume is not an actual simulation, just a reduction of results in post processing. Whole and Trabecular are the volumes used in simulations.

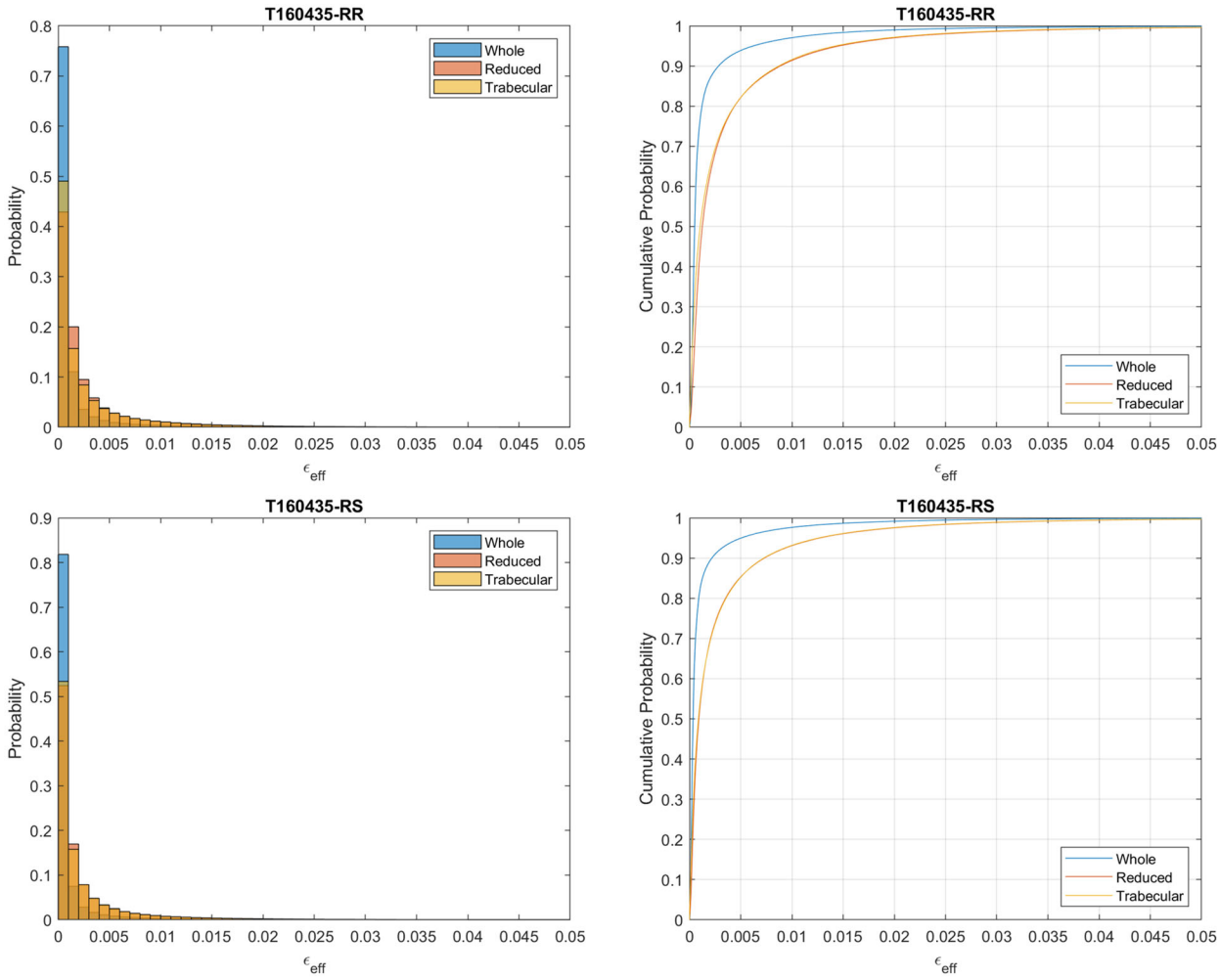


Figure 53. Histograms and CDFs for T160435 RR and RS.

Histograms and cumulative density functions (CDFs) are generated using the effective strain calculated in MATLAB. For reduced platen compression (RPC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth. Reduced Trabecular (Reduced) results are the values from the With Cortex (Whole) simulations, but only for the elements which are also present in the Trabecular Only (Trabecular) structure. Reduced volume is not an actual simulation, just a reduction of results in post processing. Whole and Trabecular are the volumes used in simulations.

4.6 Computational Requirements

This section contains the computational data for processing the simulations. The length of time to run, the amount of memory used, and the sizes of the files generated during the processing step are summarized in the following figures.

Figure 54 shows the amount of CPU time the simulations required to complete. Most of the Whole simulations required approximately 1 hour, except for the WR load cases which took ~2.5 hours. The Trabecular models were much quicker, taking around 20 minutes, save for the WR load cases which required 40-60 minutes. The significant increase in time for the WR load case is likely due to the methodology of the iterative solver in ABAQUS. After processing the input data, the solver preconditions the matrices generated to aid in the performance of the solver. The matrices in the WR load case are likely ill-conditioned for this preconditioning treatment since a large portion of the degrees of freedom are specified to be zero.

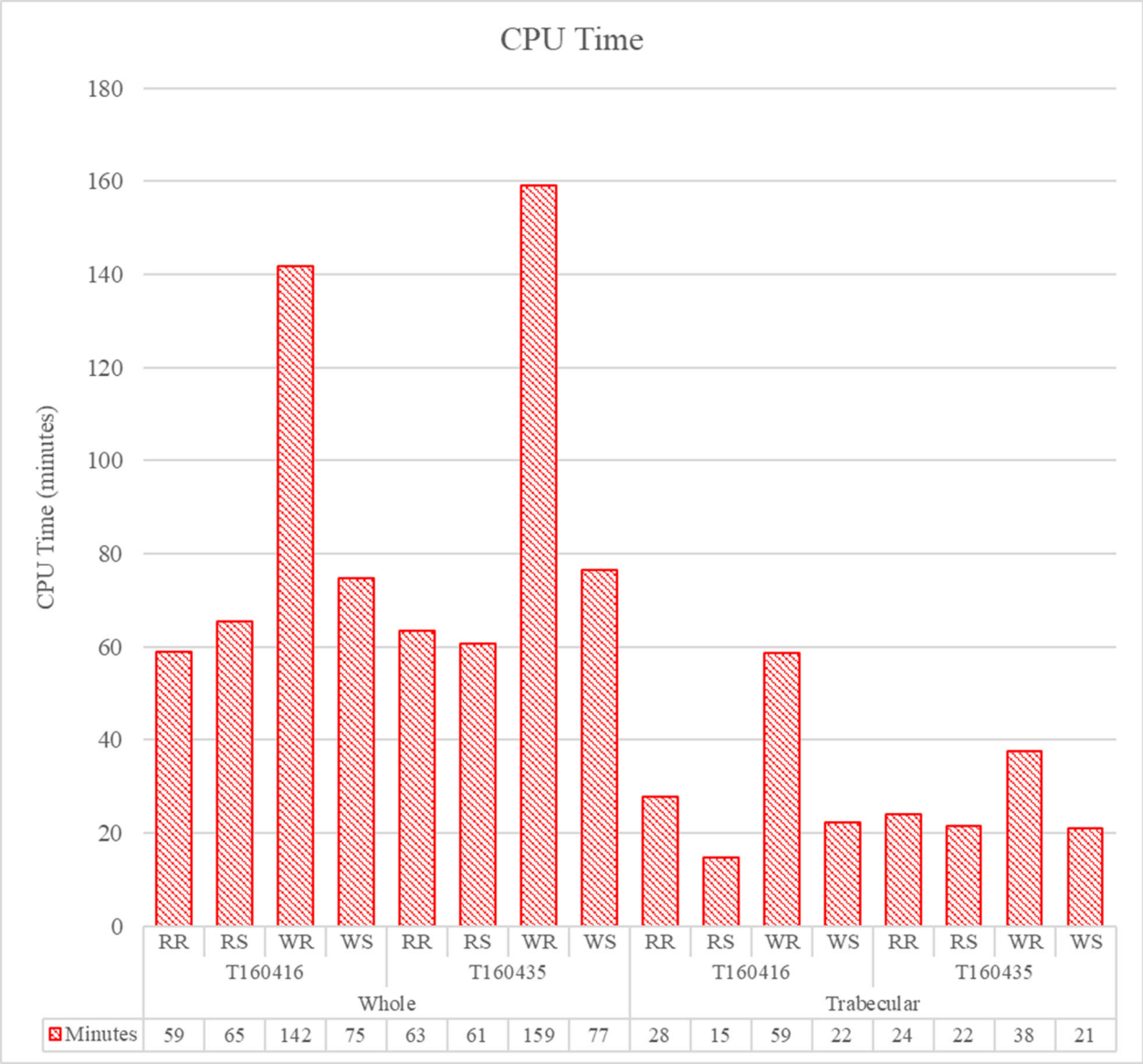


Figure 54. Run Time for Each Simulation.

CPU time is the amount of time the simulation took to run on the Ada supercomputer at TAMU HPRC, as reported in the job summary after completion. Whole volumes had 2+ million elements, Trabecular had ~800,000. Only one CPU was employed for these analyses. For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth, WR = WSC Rough, WS = WSC Smooth; each of these is simulated separately for every volume.

Figure 55. Memory Used for Each Simulation
Memory reported as the maximum memory used while processing on the Ada supercomputer at TAMU HPRC, as reported in the job summary after completion. Whole volumes had 2+ million elements, Trabecular had ~800,000. Only one CPU was employed for these analyses. WR = WSC Rough. The other boundary conditions cases were nearly identical in memory required to the WR case.

Figure 55 shows the maximum amount of memory used in each of the simulations. Only the WR load case is shown, since the other load cases required nearly identical amounts of memory (±0.1%). The Trabecular models required substantially less memory to complete the simulations, stemming from the fewer elements in the model which generate smaller matrices and fewer equations that need to be solved.

Figure 56. Sizes of Files Generated by ABAQUS Simulations.
Files included are ABAQUS input file (INP), output database (ODB), part file (PRT), and the comma separated value (CSV) reports generated in post-processing. For reduced platen compression (RPC) and whole specimen compression (WSC) the applied boundary conditions (BC) are: RR = RPC Rough, RS = RPC Smooth, WR = WSC Rough, WS = WSC Smooth; each of these is simulated separately for every volume.

Figure 56 shows the sizes of some of the files generated through the FEA process. Other files generated during processing and post-processing are not included, but do not significantly affect the results shown – they total approximately 10 MB. The input files (INP) were 7%, ODBs 43-45%, PRTs 4%, Element CSVs 27-29%, and Node CSVs 15-18% of the total memory cost of the simulations. In total the Whole models generate 16-18 GB of data, and the Trabecular models generate ~6.5 GB. For each bone, that means 20-25 GB of storage would be needed for the results using all the simulations in the current method.

5. DISCUSSION

5.1 Image Processing

MATLAB proved to be an effective tool for generating FE meshes from CT data, particularly because of its image processing capabilities. The ability to directly read DICOM files and to manipulate 2D and 3D images both in a graphical user interface (GUI) and with scripts make it well suited for process diagnostics and capable of extension to a large number of samples.

The algorithm generated and employed here for removing the cortex has benefits over a manual process. When isolating trabecular volumes by hand, a gap is often included between the inner cortex and cancellous region to avoid unintentional inclusion of cortical tissue. Bone sites like the distal femur metaphysis (DFM) used here are in the transition between dense cancellous tissue and the hollow midshaft; thus, they have far fewer trabeculae in the center of the specimen then nearer the cortex. This means that including the gap makes the "hollow" center section a greater percentage of the cancellous volume used in analysis. Densitometric quantities calculated using this volume will be underestimated compared to more traditional histomorphometry methods. The automatic cortex removal algorithm used here follows the cortex much more tightly. For some volumes, however, this automatic method includes tissue that might visually be considered cortex. For highly porous cortical bone, or areas where the cortex is not considerably thicker than the trabecular architecture, some cortical tissue may be included in the final trabecular volume. This would overestimate the trabecular strength and stiffness for whole specimen compression (WSC) tests but would not significantly change the reduced platen compression (RPC) results.

The threshold value used during the binarization process can have a large effect on the generated volume, particularly after applying a connectivity filter (Figure 28). Thus, the force results from FEA can be influenced by changing this threshold value. It is unlikely the high strain zones would be largely affected, but the strain distribution results would be skewed for a similar applied strain level. Visual verification was used in this study, comparing the final volume to the original μCT images to ensure a that it was representative recreation. A more robust method for ensuring the final volumes are accurate would likely need to be employed for a wider study.

5.2 Comparing the Geometry of Each Specimen

T160435/R223 was the larger of the two bones. This is evident in the greater amount of elements in the model (Figure 32), the larger simulated force for the WR and WS With Cortex simulations (Figure 39), and is consistent with the higher body weight of R223 throughout the experiment (Appendix B, Figure 58). T160416/R227 may be the smaller of the pair, but it has a higher ratio of cancellous to cortical tissue. In the Trabecular Only models T160416/R227 has more elements than T160435/R223 despite having fewer elements overall. This means T160435 had more cortical tissue, which was removed to generate the trabecular volumes. This manifests in the higher forces seen for the WR and WS Trabecular Only simulations.

Looking at the face node sets of the Trabecular volumes, T160435/R223 has more nodes on the bottom, but fewer on the top face. This is an indicator of the hollower geometry of T160435/R223, as illustrated in Figure 31. The simulated platen sizes are nearly the same for the two specimens, but T160416/R227 has 2.5 times as many nodes in the top RPC contact node set.

The transverse reaction forces seen in the rough simulations are highly dependent on the geometry of the specimen. T160135 has double the transverse reaction force in both the

simulations with and without the cortex. This indicates the structure of the simulated test specimen is aligned along an axis slightly skew to the z-axis along which the load was applied. This could be due to in vivo loading conditions, differences in the overall shape of the bones from different animals, or to the bone not being exactly straight during the CT scan. The RPC simulations show T160416/R227 as having a slightly higher transverse reaction, but with the difference in amount of loaded tissue between the specimens, no conclusions about the geometry can be made.

5.3 Comparing Simulation Results to Past Mechanical Testing

The force results from the RPC simulation are about one order of magnitude larger than typical force measurements from RPC tests with similar strain values. This is likely due to overestimation of the tissue mechanical properties. A Young's Modulus of 10 GPa was assigned, so if 1 GPa was used instead the results would be a closer representation. Values reported in literature vary from 1-30 GPa [22]. In the simulations done here, the Young's Modulus was specified in Pa and the units of the mesh supplied in mm. This gave force results in μN. Using MPa and mm would yield results in terms of N.

With T160435/R223 being the larger animal, it would be expected that its bones would hold more force in the different mechanical tests assuming the tissue properties were the same. Past mechanical testing results at multiple anatomical sites support this conclusion. However, in the femur RPC mechanical testing and simulations, T160416/R227 is 500% and 120% greater, respectively. This result shows the sensitivity of the RPC test to the internal "hollowness" of the specimen. As evidenced by the contact node sets and illustrated by the size of the contact areas in Figure 48, more tissue gets loaded in T160416/R227 since T160435/R223 is comparatively hollower. The platen sizing discrepancy (Figure 44) lends a potential explanation to why the

mechanical testing results are more drastically different than the simulations. T160416/R227 used a slightly larger (2.3 mm v. 2.20 mm) platen than the image processing determined, and thus likely loaded some the cortical tissue during RPC mechanical testing. T160435/R223 used a much smaller platen (1.85 mm v. 2.17 mm) for the mechanical test than the simulation. For the already hollower specimen, this means even less tissue would be loaded than in the simulation.

5.4 Comparison of Boundary Conditions

Mechanical testing of cancellous bone has been done with both glued and lubricated specimen [43]. Thus, the rough and smooth simulations have mechanical analogs. However, neither perfectly recreate *in situ* loading. The rough simulations fix the top and bottom surfaces, and the smooth simulations leave them unconstrained. *In situ*, those surfaces would be compliant, but the bone tissue above and below the test section would resist motion. It is therefore likely that the rough simulations overestimate and smooth simulations underestimate the actual structural stiffness. The simulations support that assertion as the reaction force for each volume was higher in the rough simulations.

The differences seen in the histograms and cumulative density functions (CDF) of the effective strain distributions are corroborated by the contour plots. The thick cortex induces higher strains in nearby parts of the cancellous region. This explains why the Reduced Trabecular distributions have a higher percentage of high strain elements than the Trabecular only, and why the Reduced curve on the CDFs lies between the Trabecular and Whole curves. A larger difference is visible between the contour plots of the rough and smooth WSC models than those of the RPC models. This is the same case for the CDFs. The contour plots are limited in their application for comparison between the specimen, as the models do not have the same axes. The CDFs do allow some comparison, but difference in specimen geometry cannot be ignored.

A much higher percentage of the elements in T160416 are at elevated strain levels than in T160435 in the RPC simulations. However, since more elements in T160416 are loaded by the contact BCs, the elevated strain levels do not indicate a weaker specimen as they would suggest. Using strain related failure criterion, RPC methods would need to compare distributions for a representative loaded volume, rather than the entire volume used in the simulation.

Many studies report estimated strength using a force back-calculated from the effective strain distribution [42]. The difference in the WR and WS distribution is significant up to and slightly above the applied strain, but there is little noticeable difference at very high levels of effective strain. Typical simulated strength measures would therefore be similar between the two boundary conditions, if a high enough threshold strain was chosen. Based on the similarity of the distributions for the two specimens in this study, the estimated failure force would show the same level of difference seen in the RF results.

A limitation of the effective strain used in the estimated failure criterion is that it is calculated using the strain energy density – a scalar value blind to tension or compression. Cortical bone tissue is known to have different material behaviors in tension and compression [44], and other finite element modelling studies suggest trabecular tissue shows a similar behavior [45]. Defining a material model for FEA that would treat failure differently in tension and compression would show a difference in the BCs of this study. WS simulations consistently showed more tension than did the WR simulations on the same body, meaning the estimated failure load would differ. Thus, tissue asymmetry would show different behavior between smooth and rough simulations, while the effective strain failure models would not.

5.5 Sensitivity of the RPC Test

The area loaded during RPC testing is considerably smaller than that of WSC, and it is free of the dense cortical shell. Thus, the RPC test produces reaction forces that are orders of magnitude lower than WSC tests. This means that smaller, high resolution testing equipment must be used.

In larger studies the RPC mechanical test has shown a more substantial difference between treatment groups than WSC tests [13]. Biologically, trabecular bone has a higher turnover rate than cortical bone. Therefore, any drug treatment or unloading-induced effects would be expected to have a higher impact on the cancellous region than the cortical shell. The RPC test is a more direct measurement of the cancellous bone properties, which explains the increased sensitivity compared to WSC tests. This direct cancellous measurement is demonstrated in the reaction force results on the simulated volumes with and without the cortical tissue. There is only a 5-10% decrease in the reaction force, i.e. stiffness of the specimen, when the cortex is removed; suggesting the cortical tissue is having less influence on the RPC results than the WSC simulations. The RPC method also creates a more realistic *in situ* loading condition than typical trabecular structure mechanical tests that remove cancellous bone completely from its surrounding tissue. By maintaining the tissue on the sides, only the end effects on the top and bottom are deviant from *in situ* loading.

The present study, however, shows that the RPC test is not without its downsides. T160435 was predicted to be the weaker of the two bones, despite its donor R223 having higher strength indices at most other test sites. This was due to the comparatively "hollow" center where an unrepresentative amount of tissue was successfully loaded. One of the benefits of the FEA performed here is the muted response to the ill-conditioned geometry. The mechanical

testing showed a 500% difference in the bones, whereas the difference was only 120% for the simulated reaction forces. This highlights an area where the simulation method outperforms the mechanical testing: consistency. The computational analysis does not rely on human subjectivity to assign platens or to align the specimen in a load cell. Larger studies employing the FEA techniques developed herein are expected to have the same increased cancellous sensitivity present in the mechanical RPC test results but would have less scatter in the data; therefore, the findings would be more statistically significant.

The specimens used in this study were picked because of their unusual mechanical testing results. R227 represented an outlier on the upper end, and R223 on the lower end of yield force in the RPC femur mechanical test. Most specimen used in a wider study would be expected to fall somewhere between these animals. Of particular interest would be to see how the past manual platen selections compare to computer assigned values.

5.6 Computational Requirements of μFEA

In the models generated from μCT data, the large number of elements complicates the analysis process. Workstations with sufficient resources are required, and the large amount of data generated presents file transfer and storage issues.

Processing the images in MATLAB using only the scripts written for this study does not strain computer resources, but any higher functions such as using the Volume Viewer do. RAM in excess of 16GB was consistently used, and while the process might still be achievable on 8GB machines, the user experience will be slower.

The methodology of the algorithms used in generating the meshes and writing the input files prioritizes speed over memory. In the case of the element connectivity array (ECA) and nodal coordinate array (NCA), the entire matrices were generated and stored in local memory

prior to being written to the input file. Alternatively, each row of the matrix could be written to the input file as it is calculated and then discarded. This would alleviate memory but would take considerably longer as the low-level file I/O functions are called millions of times rather than once.

The large file sizes generated during and after FEA processing present issues as well. From a storage perspective, modern portable hard drives (4TB+) are increasingly affordable and would be able to store the results for hundreds of bones. However, transferring the files in this study was challenging, even with only two animals being investigated. With the FEA being processed on Texas A&M High Performance Research Computing (HPRC) supercomputer resources, the analysis files are stored on their systems. Access via the OnDemand Portal is not an efficient means of file transfer. There are other programs available that simplify the file transfer, allowing transfer overnight without user monitoring. Additionally, only the principal stresses and strains were used in further calculations, so only those quantities need to be exported in the CSV reports. This would alleviate the cost of memory storage and file transfer times.

MATLAB is well suited for handling tables and matrices, so post-processing the tabular results is not computationally complex. Generating the contour plots, however, requires a suitable graphic workstation to reasonably be able to work with the ODBs. Using the OnDemand Portal and launching ABAQUS in a virtual window was sufficient if the internet speed was fast enough, usually >90 Mbps. Most wired connections at Texas A&M meet this need; WiFi may not achieve those speeds. Using a local copy of ABAQUS on a machine would solve the issue as well. To generate the images in this document, it was necessary to use a monitor with greater than 1080p resolution.

In general, the processing time required for the simulations to complete is not a concern. Multiple simulations can be running at the same time on the supercomputer, and batch files could be written to have sequential flows run overnight. The analyses herein used only one CPU, so parallelization could be investigated to bring down the computation time (this would not change the computational cost, as more CPUs running shorter times would be billed the same).

The rough simulations took longer to complete than their smooth counterparts, with the WR load case taking the longest of any. The Trabecular Only simulations also ran much faster than the With Cortex versions, solely due to the much lower number of elements in the model. While the WR and WS simulations showed differences for the same specimen, comparison between specimens was equally effective using either. For this reason, completing both is not necessary. Running only the smooth BCs for both the WSC and RPC simulations would substantially reduce the run time and memory cost of this study's FEA.

6. LIMITATIONS

The present study is limited due to its sample size. With the development of the mesh generation and FEA process being the primary goal, only two specimens were fully analyzed. Other samples are expected to have results between those of the bones in this study, since both specimens were outliers in past mechanical testing results. However, the ability of the process to show significant differences between groups in a larger study remains unconfirmed.

Linear FEA was performed in this study, so the results cannot be extended to true material failure. Linear analysis is limited to describing material behavior up to the yield point. Further, isotropic properties were assumed, and bone tissue is well known to be anisotropic and orientation dependent.

The only mesh investigated was that of direct voxel conversion, though the element size in these analyses were sufficiently small to provide 4+ elements through the thickness of an average trabecula. Image processing could be used to refine the element size. This would not change the overall structure, as the original volume would still be limited by the CT scan resolution. However, large gradients in stress and strain would be more accurate in the refined volume. Computationally, the cost to refine the mesh would be high.

An inherent limitation to any CT data is that only bone mineral is captured in the images. This produces a recreation of the bone geometry but does not include surrounding soft tissues like marrow and muscle. Therefore, the FEA performed herein captures the response of the mineralized structure to loading but does not account for interactions with surrounding tissue.

Finally, the transfer function used in mapping the CT scan bone density values to TIFF pixel intensities when exporting images is not known. Consequently, the commonly reported binarization threshold value in $mgHA/cm^3$ cannot be determined.

7. FUTURE WORK

The methods for CT-based mesh generation and FEA developed in this thesis can be extended to larger groups to investigate differences between drug treatments and disuse-induced bone loss. The reaction force and stiffness results in a wider study are expected to mirror the data from RPC mechanical testing but would be subject to less scatter. This would make the results more statistically significant, even if less dramatic.

This thesis compares the strain distributions in the specimens with the cortical shell and the generated trabecular volumes by reducing the whole specimen data to include only the elements also present in the trabecular only structure. This allows for comparison of equal volumes. This technique could be extended by reducing both volumes to include only the elements in the cylindrical core that is loaded by the RPC platens. This would allow for a much more detailed comparison of the strain distributions generated during the RPC simulations.

The reaction forces calculated here can be scaled by the number of elements above a threshold level of effective strain. This would provide estimated failure loads commonly reported in other CT FEA studies [42]. The meshes generated could be extended to non-linear FEA. This would allow for more realistic mechanical testing beyond the point of yield. A truly robust FEA could be achieved through inclusion of a material failure model that treats tension and compression differently, as observed in mechanical testing of bone tissue.

Computationally, the algorithms developed herein have room for improvement. Parallelization could be employed during the processing step of the ABAQUS simulations, potentially improving the speed of the analysis but not changing the memory cost. The pre- and post-processing in MATLAB has been optimized for speed, but alternative numeric types could be employed to reduce the memory requirements of processing and storage.

8. CONCLUSIONS

Finite element analysis (FEA) of bone tissue is growing more popular as the capabilities for patient specific models using CT data grow. This study sought to develop a method for pre-existing CT images to be converted to finite element (FE) meshes, and to investigate the capabilities of such models to simulate mechanical tests. MATLAB was used to process the images from their raw 2D format into usable 3D volumes. An element and node numbering scheme was then used to convert the voxels in the binary 3D images directly to hexahedral elements in a FE mesh. This mesh was imported to ABAQUS for processing. Different boundary conditions (BC) were imposed on the models to simulate multiple mechanicals tests. Both rough and smooth simulated platens were used in modelling a Whole Specimen Compression (WSC) and Reduced Platen Compression (RPC) test. After completion, the models were post-processes in ABAQUS for visual results, and in MATLAB for quantitative results. Strain distributions were compared to results commonly reported in modern bone FEA literature.

This study expands upon past 2D and 3D continuum modeling of the RPC test by introducing discrete trabecular architecture. The results herein indicate that the RPC test mimics loading only the cancellous tissue, without the added processing step of physically removing the cortical tissue. The reaction forces produced in RPC are substantially smaller than in WSC. This changes the demand from an apparatus and sensitivity standpoint. Two specimens were investigated for comparison, and the FEA results mirror past mechanical testing data but are less dramatic in their difference. If this FEA procedure was extended to a wider sample of specimens, the results would likely be similar to the mechanical testing data but with less scatter. This would mean any differences between groups would be more statistically significant.

The rough and smooth boundary conditions (BC) simulated different styles of mechanical tests. The results indicated their differences, with the rough simulations being stiffer across the board. In the strain distributions, the differences were more pronounced for the WSC than the RPC simulations. Even then, the different BCs showed the same amount of difference between the two animals. For this reason, future studies need only pick either the rough or smooth BC to compare groups.

Another objective of this study was to examine the computational requirements of the discrete trabecular models. The meshes generated are quite large, exceeding 2 million elements. Consequently, the memory demands for completing the analysis are complex. Supercomputer resources were necessary for processing, due to the large amount of RAM needing during analysis. The result files generated are large as well, so file transfer and storage become non-trivial. For a wider study, this would likely be the most cumbersome issue. The computation time is of minimal concern. The simulations completed in approximately and hour, and multiple simulations can run simultaneously.

Overall, the mesh generation technique developed in this study is quite powerful. This allows users to circumvent expensive commercial software packages and offers more control in mesh and analysis parameters. While only linear analysis was performed in this work, the mesh generated can easily be extended to more robust analyses. The mesh specifies the geometry, but the rest of the model parameters can be determined by the user. Nonlinear FEA and more realistic material models can be applied without needing to change the mesh algorithm. The versatility of this study's mesh generation scheme is that it is not limited to bone. Any 3D binary image or matrix, even those created artificially or by manually manipulating matrices, could be converted to a finite element mesh.

REFERENCES

[1] A. Qaseem, M.A. Forciea, R.M. McLean, T.D. Denberg, f.t.C.G.C.o.t.A.C.o. Physicians, Treatment of Low Bone Density or Osteoporosis to Prevent Fractures in Men and Women: A Clinical Practice Guideline Update From the American College of PhysiciansTreatment of Low Bone Density or Osteoporosis to Prevent Fractures in Men and Women, Annals of Internal Medicine 166(11) (2017) 818-839.

[2] A. LeBlanc, V. Schneider, L. Shackelford, S. West, V. Oganov, A. Bakulin, L. Voronin, Bone mineral and lean tissue loss after long duration space flight, J Musculoskelet Neuronal Interact 1(2) (2000) 157-60.

[3] A. LeBlanc, T. Matsumoto, J. Jones, J. Shapiro, T. Lang, L. Shackelford, S.M. Smith, H. Evans, E. Spector, R. Ploutz-Snyder, J. Sibonga, J. Keyak, T. Nakamura, K. Kohri, H. Ohshima, Bisphosphonates as a supplement to exercise to protect bone during long-duration spaceflight, Osteoporosis International 24(7) (2013) 2105-2114.

[4] T. Lang, J.J.W.A. Van Loon, S. Bloomfield, L. Vico, A. Chopard, J. Rittweger, A. Kyparos, D. Blottner, I. Vuori, R. Gerzer, P.R. Cavanagh, Towards human exploration of space: the THESEUS review series on muscle and bone research priorities, npj Microgravity 3(1) (2017) 8.

[5] R. Huiskes, E.Y. Chao, A survey of finite element analysis in orthopedic biomechanics: the first decade, J Biomech 16(6) (1983) 385-409.

[6] M.R. Betker, J.S. Fernando, S.P. Whalen, The history of the microprocessor, Bell Labs Technical Journal 2(4) (1997) 29-56.

[7] B. van Rietbergen, H. Weinans, R. Huiskes, A. Odgaard, A new method to determine trabecular bone elastic properties and loading using micromechanical finite-element models, Journal of biomechanics 28(1) (1995) 69-81.

[8] A.L. Adams, H. Fischer, D.L. Kopperdahl, D.C. Lee, D.M. Black, M.L. Bouxsein, S. Fatemi, S. Khosla, E.S. Orwoll, E.S. Siris, T.M. Keaveny, Osteoporosis and Hip Fracture Risk From Routine Computed Tomography Scans: The Fracture, Osteoporosis, and CT Utilization Study (FOCUS), Journal of Bone and Mineral Research 33(7) (2018) 1291-1301.

[9] T.P. Harrigan, M. Jasty, R.W. Mann, W.H. Harris, Limitations of the continuum assumption in cancellous bone, Journal of Biomechanics 21(4) (1988) 269-275.

[10] P.K. Zysset, X. Edward Guo, C. Edward Hoffler, K.E. Moore, S.A. Goldstein, Elastic modulus and hardness of cortical and trabecular bone lamellae measured by nanoindentation in the human femur, Journal of Biomechanics 32(10) (1999) 1005-1012.

[11] E. Takai, K.D. Costa, A. Shaheen, C.T. Hung, X.E. Guo, Osteoblast elastic modulus measured by atomic force microscopy is substrate dependent, Annals of biomedical engineering 33(7) (2005) 963-971.

[12] X.E. Guo, Mechanical properties of cortical bone and cancellous bone tissue, Bone mechanics handbook 10 (2001) 1-23.

[13] H.A. Hogan, S.P. Ruhmann, H.W. Sampson, The mechanical properties of cancellous bone in the proximal tibia of ovariectomized rats, Journal of Bone and Mineral Research 15(2) (2000) 284-292.

[14] W. Rogers, Methods and Modeling for The Reduced Platen Compression of Cancellous Bone in Rodent Proximal Tibia, MS Thesis, Texas A&M University, 2002.

[15] J.S. Nyman, S. Uppuganti, A.J. Makowski, B.J. Rowland, A.R. Merkel, J.A. Sterling, T.L. Bredbenner, D.S. Perrien, Predicting mouse vertebra strength with micro-computed tomography-derived finite element analysis, Bonekey Rep 4 (2015) 664-664.

[16] Y. Chen, E. Dall′Ara, E. Sales, K. Manda, R. Wallace, P. Pankaj, M. Viceconti, Micro-CT based finite element models of cancellous bone predict accurately displacement once the boundary condition is well replicated: A validation study, Journal of the Mechanical Behavior of Biomedical Materials 65 (2017) 644-651.

[17] O. Vestrum, M. Langseth, T. Børvik, Finite element modeling of porous polymer pipeline coating using X-ray micro computed tomography, Composites Part B: Engineering 172 (2019) 406-415.

[18] R.B. Martin, D.B. Burr, N.A. Sharkey, D.P. Fyhrie, Skeletal tissue mechanics, Springer1998.

[19] D.B. Burr, M.R. Allen, Editors, Basic and Applied Bone Biology, Academic Press 2014.

[20] Anatomy and Physiology. https://opentextbc.ca/anatomyandphysiology/chapter/6-3-bone-structure/.

[21] H. Roesler, The history of some fundamental concepts in bone biomechanics, Journal of Biomechanics 20(11) (1987) 1025-1034.

[22] S.C. Cowin, Bone mechanics handbook, CRC press2001.

[23] E. Fukada, I. Yasuda, On the piezoelectric effect of bone, Journal of the physical society of Japan 12(10) (1957) 1158-1162.

[24] H.M. Frost, Changing concepts in skeletal physiology: Wolff's Law, the Mechanostat, and the “Utah Paradigm”, American Journal of Human Biology: The Official Journal of the Human Biology Association 10(5) (1998) 599-605.

[25] H.M. Frost, Bone's mechanostat: a 2003 update, The Anatomical Record Part A: Discoveries in Molecular, Cellular, and Evolutionary Biology: An Official Publication of the American Association of Anatomists 275(2) (2003) 1081-1101.

[26] J.B. Tyrovola, X. Odont, The “mechanostat theory” of frost and the OPG/Rankl/RANK system, Journal of cellular biochemistry 116(12) (2015) 2724-2729.

[27] Y.-c. Fung, Biomechanics: mechanical properties of living tissues, Springer Science & Business Media2013.

[28] W. George, D. Vashishth, Damage mechanisms and failure modes of cortical bone under components of physiological loading, Journal of orthopaedic research 23(5) (2005) 1047-1053.

[29] C.H. Turner, D.B. Burr, Basic biomechanical measurements of bone: a tutorial, Bone 14(4) (1993) 595-608.

[30] T.M. Keaveny, X.E. Guo, E.F. Wachtel, T.A. McMahon, W.C. Hayes, Trabecular bone exhibits fully linear elastic behavior and yields at low strains, Journal of biomechanics 27(9) (1994) 1127-1136.

[31] J. Black, Comparison of the Persisting Effects of Bisphosphonate Treatments Prior to Hindlimb Unloading on Mechanical and Densitometric Properties in the Tibia of Adult Male Rats, MS Thesis, Texas A&M University, 2016.

[32] J. Kosniewski, The Efficacy of Bisphosphonate Pre-Treatment in Preventing Losses in Densitometric and Mechanical Properties During Hindlimb Unloading and Throughout Reambulation in the Distal Femur Metaphusis of Adult Male Rats, MS Thesis, Texas A&M University, 2017.

[33] J. Brezicha, Skeletal Responses to Simulated Microgravity in the Adult Rat Model After Sclerostin Antibody Pretreatment, PhD Dissertation, Texas A&M University, 2019.

[34] J. Hsieh, Computed Tomography: Principles, Design, Artifacts, and Recent Advances, (2015).

[35] Z. Poniznik, V. Salit, M. Basista, D. Gross, Effective elastic properties of interpenetrating phase composites, Computational Materials Science 44(2) (2008) 813-820.

[36] J.N. Reddy, Introduction to the Finite Element Method, 4th edition. ed., McGraw-Hill Education, New York, 2019.

[37] D. Ulrich, B. van Rietbergen, H. Weinans, P. Rüegsegger, Finite element analysis of trabecular bone structure: a comparison of image-based meshing techniques, Journal of Biomechanics 31(12) (1998) 1187-1192.

[38] T. Wronski, E. Morey-Holton, Skeletal response to simulated weightlessness: a comparison of suspension techniques, Aviation, space, and environmental medicine 58(1) (1987) 63-68.

[39] H.R. Buie, G.M. Campbell, R.J. Klinck, J.A. MacNeil, S.K. Boyd, Automatic segmentation of cortical and trabecular compartments based on a dual threshold technique for in vivo micro-CT bone analysis, Bone 41(4) (2007) 505-515.

[40] L.T. Wilkerson, Finite element analysis of cancellous bone, MS Thesis, University of Kentucky, 2012.

[41] R. Boudreaux, Sequential High-Impact, Free-Fall Loading and Zoledronic Acid as a Novel Pre-Treatment for Disuse-Induced Bone Loss, PhD Dissertation, Texas A&M University, 2014.

[42] W. Pistoia, B. van Rietbergen, E.M. Lochmüller, C.A. Lill, F. Eckstein, P. Rüegsegger, Estimation of distal radius failure load with micro-finite element analysis models based on three-dimensional peripheral quantitative computed tomography images, Bone 30(6) (2002) 842-848.

[43] F. Linde, I. Hvid, The effect of constraint on the mechanical behaviour of trabecular bone specimens, Journal of Biomechanics 22(5) (1989) 485-490.

[44] D.T. Reilly, A.H. Burstein, The elastic and ultimate properties of compact bone tissue, Journal of Biomechanics 8(6) (1975) 393-405.

[45] G.L. Niebur, M.J. Feldstein, J.C. Yuen, T.J. Chen, T.M. Keaveny, High-resolution finite element models with tissue strength asymmetry accurately predict failure of trabecular bone, Journal of Biomechanics 33(12) (2000) 1575-1583.

APPENDIX A ADDITIONAL TABLES

Table 3. Uncorrected Reaction Force Results from Simulations (μN)

T160435	BC	Top X	Bottom X	Top Y	Bottom Y	Top Z	Bottom Z
With Cortex	WR	7.647E+07	-7.647E+07	3.538E+08	-3.538E+08	-3.834E+09	3.834E+09
	WS	0.000E+00	-2.031E+02	0.000E+00	-1.535E+02	-3.185E+09	3.185E+09
	RR	4.819E+06	-4.819E+06	-1.352E+06	1.352E+06	-4.888E+07	4.888E+07
	RS	0.000E+00	6.003E-03	0.000E+00	1.693E-02	-3.968E+07	3.968E+07
Reduced Trab	WR	1.577E+07	-3.651E+06	5.640E+07	-1.397E+08	-4.329E+08	1.184E+09
	WS	0.000E+00	-2.031E+02	0.000E+00	-1.535E+02	-3.132E+08	1.206E+09
	RR	4.819E+06	-4.819E+06	-1.352E+06	1.352E+06	-4.888E+07	4.888E+07
	RS	0.000E+00	6.003E-03	0.000E+00	1.693E-02	-3.968E+07	3.968E+07
Trabecular Only	WR	1.879E+07	-1.879E+07	6.681E+07	-6.681E+07	-4.898E+08	4.898E+08
	WS	0.000E+00	5.233E+00	0.000E+00	1.130E+00	-3.951E+08	3.951E+08
	RR	3.994E+06	-3.994E+06	-1.385E+06	1.385E+06	-4.464E+07	4.464E+07
	RS	0.000E+00	2.483E-03	0.000E+00	-1.133E-03	-3.749E+07	3.749E+07
T160416	**BC**	**Top X**	**Bottom X**	**Top Y**	**Bottom Y**	**Top Z**	**Bottom Z**
With Cortex	WR	-8.614E+07	8.614E+07	-1.334E+08	1.334E+08	-3.594E+09	3.594E+09
	WS	0.000E+00	-4.297E+02	0.000E+00	-1.890E+02	-3.150E+09	3.150E+09
	RR	4.019E+06	-4.019E+06	-6.336E+06	6.336E+06	-1.017E+08	1.017E+08
	RS	0.000E+00	-6.739E-02	0.000E+00	4.420E-02	-8.948E+07	8.948E+07
Reduced Trab	WR	-2.409E+07	2.902E+07	-3.396E+07	4.559E+07	-6.163E+08	1.131E+09
	WS	0.000E+00	-4.297E+02	0.000E+00	-1.890E+02	-5.120E+08	1.196E+09
	RR	4.019E+06	-4.019E+06	-6.336E+06	6.336E+06	-1.017E+08	1.017E+08
	RS	0.000E+00	-6.739E-02	0.000E+00	4.420E-02	-8.948E+07	8.948E+07
Trabecular Only	WR	-1.299E+07	1.299E+07	-3.251E+07	3.251E+07	-5.350E+08	5.350E+08
	WS	0.000E+00	-1.033E+01	0.000E+00	-2.230E+01	-4.675E+08	4.675E+08
	RR	3.639E+06	-3.639E+06	-5.597E+06	5.597E+06	-9.280E+07	9.280E+07
	RS	0.000E+00	-2.263E+00	0.000E+00	-1.108E+00	-8.332E+07	8.332E+07

WR = Whole Rough, WS = Whole Smooth, RR = RPC Rough, RS = RPC Smooth. Top refers to most proximal surface of distal femur metaphysis section used in simulations; Bottom is the most distal surface. See Figure 35 for illustration of the With Cortex, Reduced Trab, and Trabecular Only volumes. Z is the loading direction. Abaqus input parameters were specified in mm and Pa, so force results are in μN; see Discussion section 5.3 for more information.

Table 4. µCT Results for Both Animals

µCT	R223	R227
BV/TV	0.2681	0.2991
Conn-Dens.	69.7435	77.1044
SMI	1.2522	1.2637
Tb.N	3.6954	4.1161
Tb.Th	0.0913	0.0993
Tb.Sp	0.2626	0.2274
vBMD	965.051	982.483
DA	1.5946	1.5448
Dim-Z	148	148
T-Number	T16-0435	T16-0416
Sigma	0.8	0.8
Supp	1	1
Thresh	295	295
tp	d56	d112
Ctrlf-Name	Rat Femur 19mm	Rat Femur 19mm

The femurs in this study were previously scanned using a Scanco µCT 50 (Scanco Medical AG, Switzerland) using a 20 µm nominal voxel size (55 kVp, 145 µA, 400 ms, 0.5 mm Al filter). For definition of structural parameters see original publication [32].

Table 5. pQCT Results for Both Animals.

pQCT - DFM	R223	R223	R223	R223	R227	R227	R227	R227
Total BMC (mg/mm)	14.09	13.50	12.79	12.24	14.28	13.59	13.31	12.75
Total BMD (mg/cm3)	637.2	640.4	642.3	649.7	610.9	612.8	624.2	630.6
Trab BMC (mg/mm)	6.14	5.42	4.41	3.49	8.00	7.08	6.18	5.19
Trab BMD (mg/cm3)	431.1	404.2	363.1	318.1	490.3	461.1	437.3	403.6
Total Bone Area (mm2)	22.11	21.08	19.91	18.84	23.37	22.18	21.32	20.22
Trab Area (mm2)	14.23	13.40	12.14	10.96	16.32	15.35	14.13	12.87
Cort BMC (mg/mm)	7.95	8.08	8.38	8.75	6.28	6.51	7.13	7.56
Cort BMD (mg/cm3)	1009.5	1052.6	1078.0	1110.4	890.0	953.5	991.5	1027.8
Cort Area (mm2)	7.88	7.68	7.78	7.88	7.05	6.83	7.19	7.35
Cort Thickness (mm)	0.563		0.598	0.620	0.448	0.478		
Polar Area MOI (mm4)	91.328	83.320	75.352	68.257	103.110	93.010	86.499	78.394
Calc Cort Thickness (mm)	0.524617	0.525088	0.551671	0.581069	0.448221	0.446645	0.484283	0.512954

See previous publication for pQCT scan parameters [32].

Table 6. Reduced Platen Compression (RPC) Mechanical Testing Results for Femur and Tibia of Both Animals.

RPC	Femur		Tibia	
Animal	**R223**	**R227**	**R223**	**R227**
Slope	19.90136	161.94079	64.44822	35.90596
Max Force	3.09396	17.6987	8.80871	2.80063
Yield Force	2.77426	17.52981	7.16045	1.65709
Platen Size	1.90	2.30	1.95	2.15
Actual Platen Size	1.85	2.3	1.925	2.16
Specimen Thickness	1.97	2.00	2.05	1.9
Displacement at Yield	0.14148	0.11054	0.1132	0.05009
Displacement at Max Force	0.17375	0.10627	0.1864	0.18382
Energy to Max Force	0.78899	1.05847	2.05629	0.37308

See previous publication for details on RPC process and specimen preparation. Femur [32], Tibia [31].

Table 7. Femoral Neck Mechanical Testing Results for Both Animals.

Femoral Neck Test	R223	R227
Max Compressive load (N)	107.05169	81.47681
Comp Extension at Max Load (mm)	0.5632	0.52506
Y-intecept	5.99912	1.25401
Xo	-0.03623	-0.00851
Stiffness	165.5863	147.43159
Max Force	107.05169	81.47681
Yield Force	0.99966	30.41093
Yield Force (95%)	0.99966	-1.22675
Yield Force (90%)	0.99966	-1.22675
Fracture Force	107.05169	81.33155
Displacement at Yield	0.60396	0.20958
Displacement at Yield (95%)	0.60396	0.54024
Displacement at Yield (90%)	0.60396	0.54024
Displacement at Max Force	0.59943	0.53357
Displacement at Fracture	0.59943	0.53144
Energy to Yield	31.30988	3.17985
Energy to Yield (95%)	31.30988	21.61904
Energy to Yield (90%)	31.30988	21.61904
Energy to Max Force	31.02155	21.25545
Energy to Fracture	31.02155	21.08221
Energy from Yield to Max Force	-0.28833	18.0756
Energy from Yield to Fracture	-0.28833	17.90236
Energy from Max Force to Fracture	0	-0.17324
Test comments		

The yield results do not make physical sense, so the max compressive load was used for comparison. See previous publication for details on femoral neck mechanical testing parameters and specimen preparation details [33].

Table 8. Three-Point Bend Mechanical Test Results for Femur and Tibia of Both Animals.

Three-Point Bend	Femur		Tibia	
Animal	**R223**	**R227**	**R223**	**R227**
Stiffness (N/mm)	578.0899	577.5969	386.5718	392.2169
Max Force (N)	255.3505	292.2402	145.7552	100.4291
Yield Force (N)	127.3088	137.7618	130.7758	77.67468
Displacement at Yield (mm)	0.22585	0.24071	0.34855	0.20019
Energy to Yield (mJ)	15.90938	18.92429	23.42027	7.82472
Displacement at Fracture (mm)	0.91598	0.89604	0.64734	0.27979
Energy to Fracture (mJ)	158.9281	173.8517	64.92221	14.93378
Polar Area MOI (mm^4)	30.916	29.877	11.434	10.97
AP Dimension (mm)	4.05	3.95	3.75	3.8
Span (mm)	15	15	18	18
CSMI (mm^4)	15.458	14.9385	5.717	5.485
Modulus (GPa)	2.629509	2.718632	8.215579	8.688123
Yield Stress (MPa)	62.5405	68.29991	193.0069	121.079
Ultimate Stress (MPA)	125.4411	144.8877	215.1145	156.5485
Pre-Yield Toughness (mJ/mm^3)	0.844073	0.988273	2.880423	1.029981
Post Yield Displacement (mm)	0.69013	0.65533	0.29879	0.0796
Notes				

See previous publication for details of three-point bend test parameters and specimen preparation [31].

Table 9. Body Weight by Week for Both Animals.

Week of	R223	R227
28-Jul	462	451
4-Aug	491	469
	480	467
11-Aug	490	476
	492	476
18-Aug	490	474
	493	478
25-Aug	495	471
	485	472
1-Sep		
	461	442
8-Sep	470	443
	471	447
15-Sep	484	452
	482	450
22-Sep	488	457
	471	443
29-Sep		446
6-Oct		
13-Oct		
20-Oct		433
27-Oct		450
3-Nov		
10-Nov		499

Weights are in grams. Hindlimb Unloading occurred for 4 weeks beginning September 1. See previous publications for details of animal care and experiment protocol [31-33].

APPENDIX B ADDITIONAL FIGURES

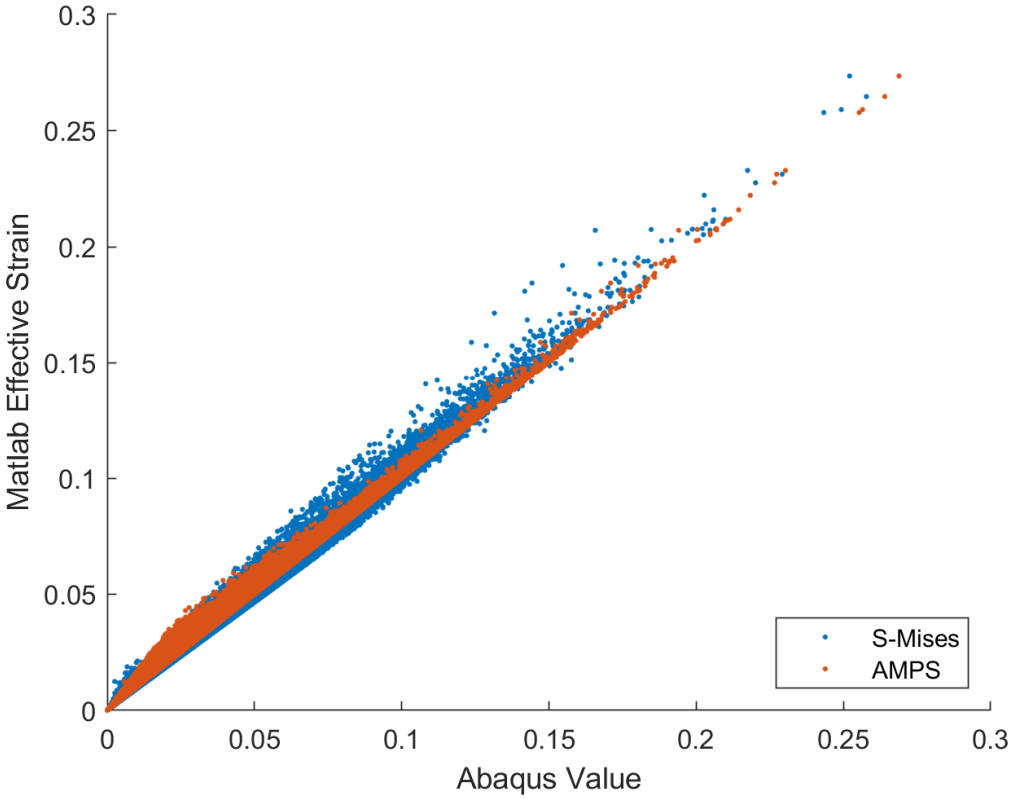


Figure 57. Effective Strain v. Absolute Maximum Principal Strain (AMPS) and Mises Stress.
Abaqus Value is the exported value for the element, Matlab Effective Strain is the value for the element calculated in post-processing using the strain energy density. The AMPS correlates with effective strain and has less scatter than the Mises stress.

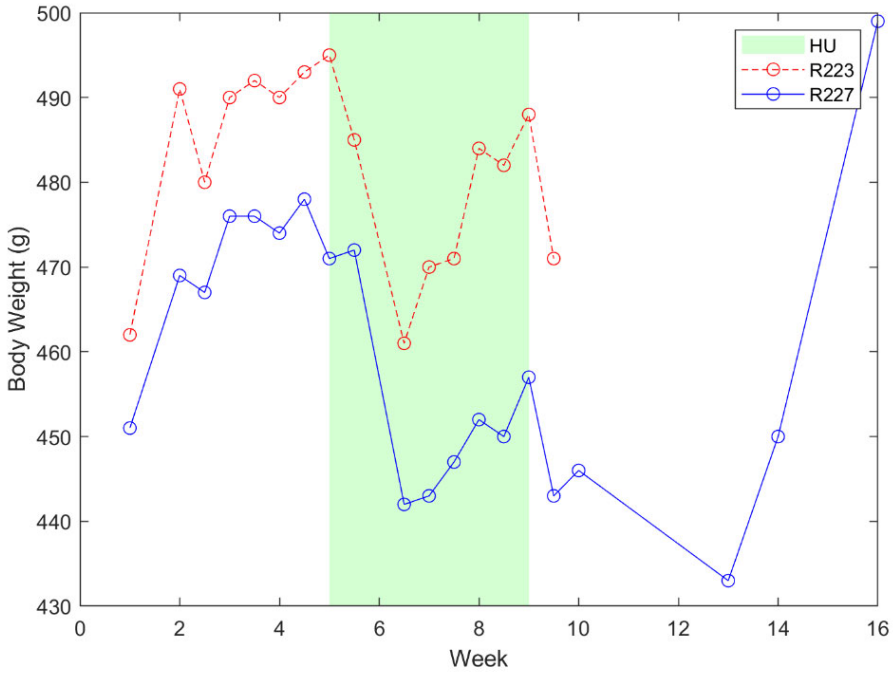


Figure 58. Body Weights Over Time of Each Animal.
HU is 4-week hindlimb unloading protocol.

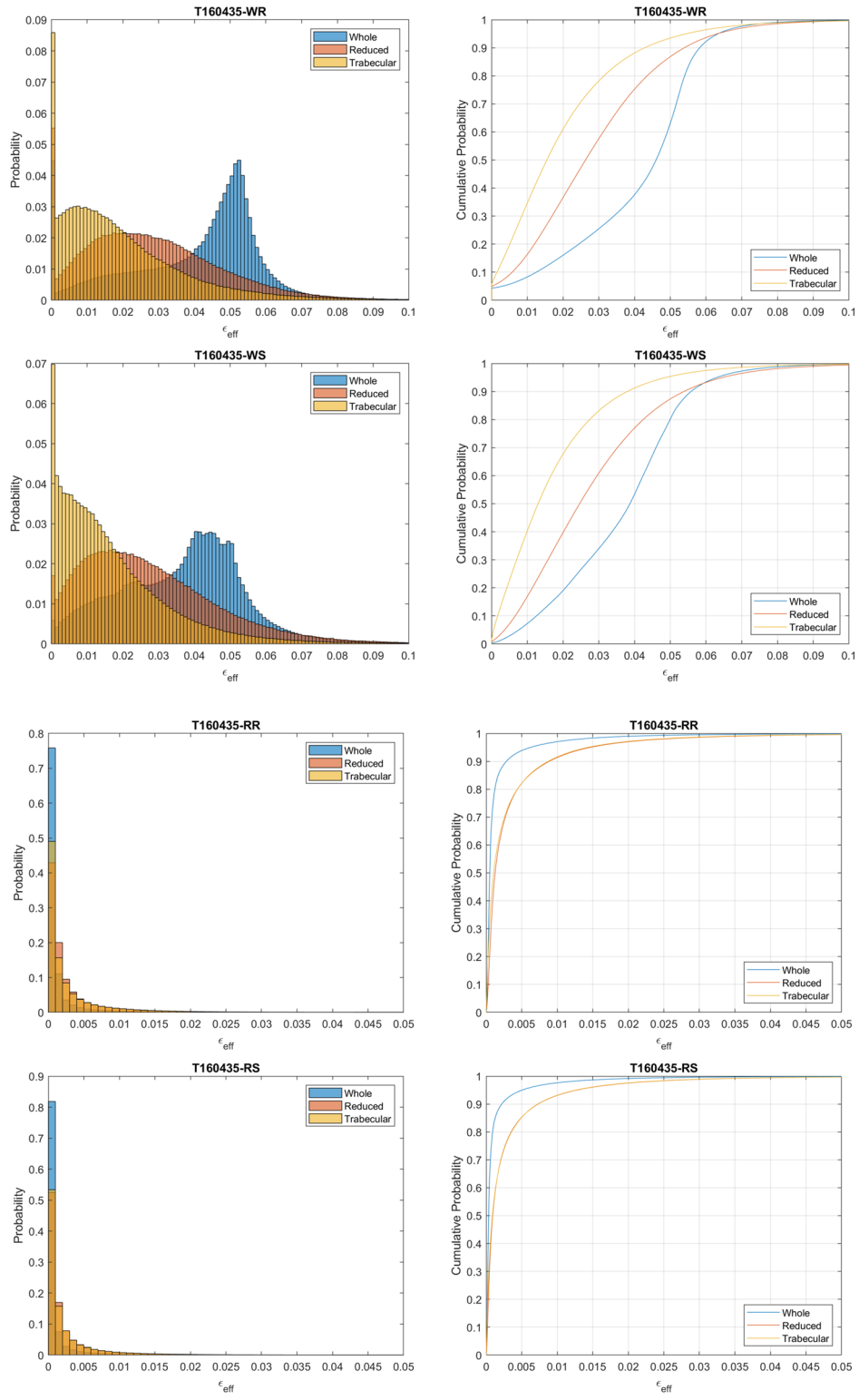


Figure 59. Histograms and Cumulative Density Functions (CDF) for T160435.
WR = Whole Rough, WS = Whole Smooth, RR = RPC Rough, RS = RPC Smooth.

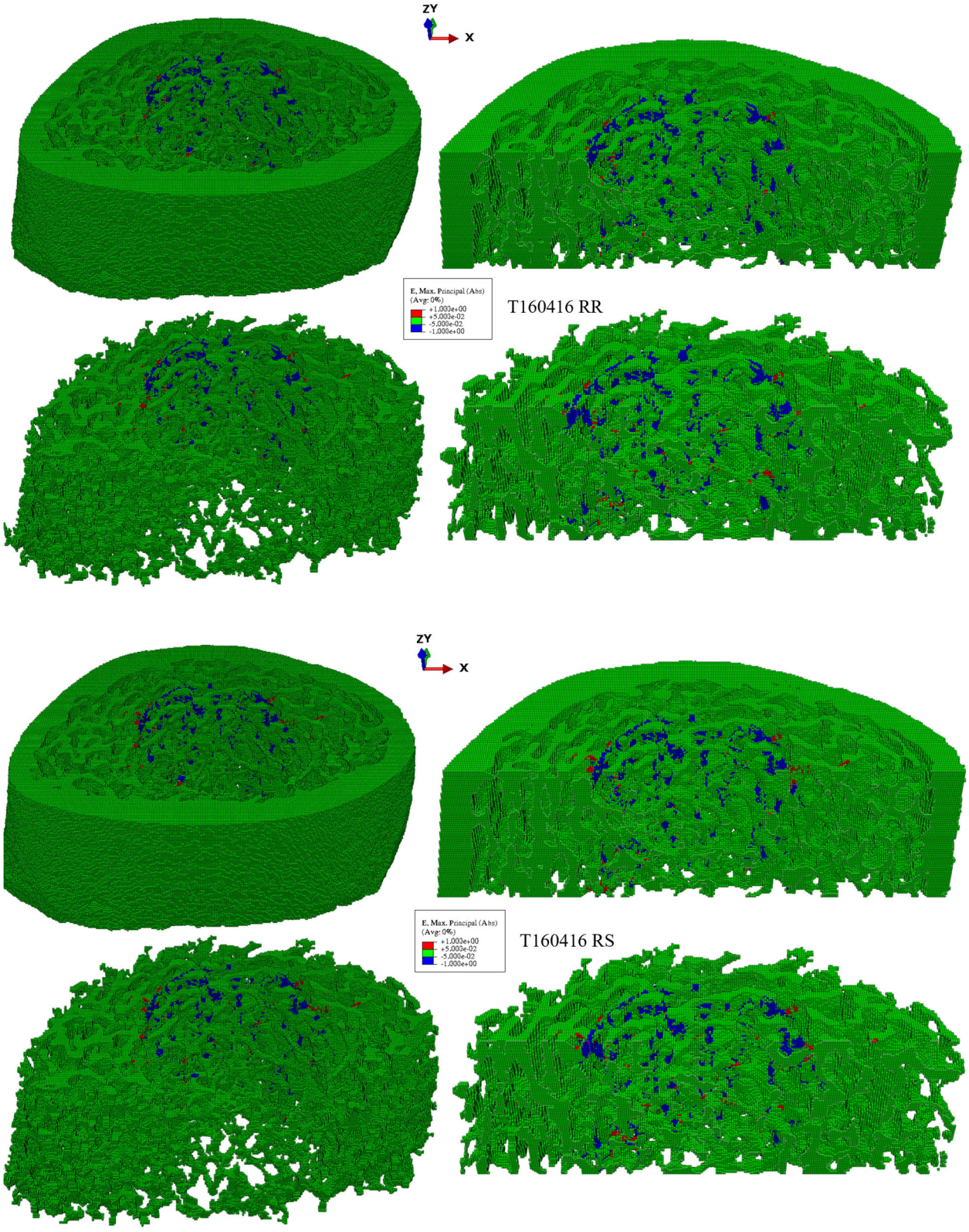


Figure 60. All Contour Plots for T160416 Reduced Platen Compression (RPC) Simulations.
RR = RPC Rough, RS = RPC Smooth.

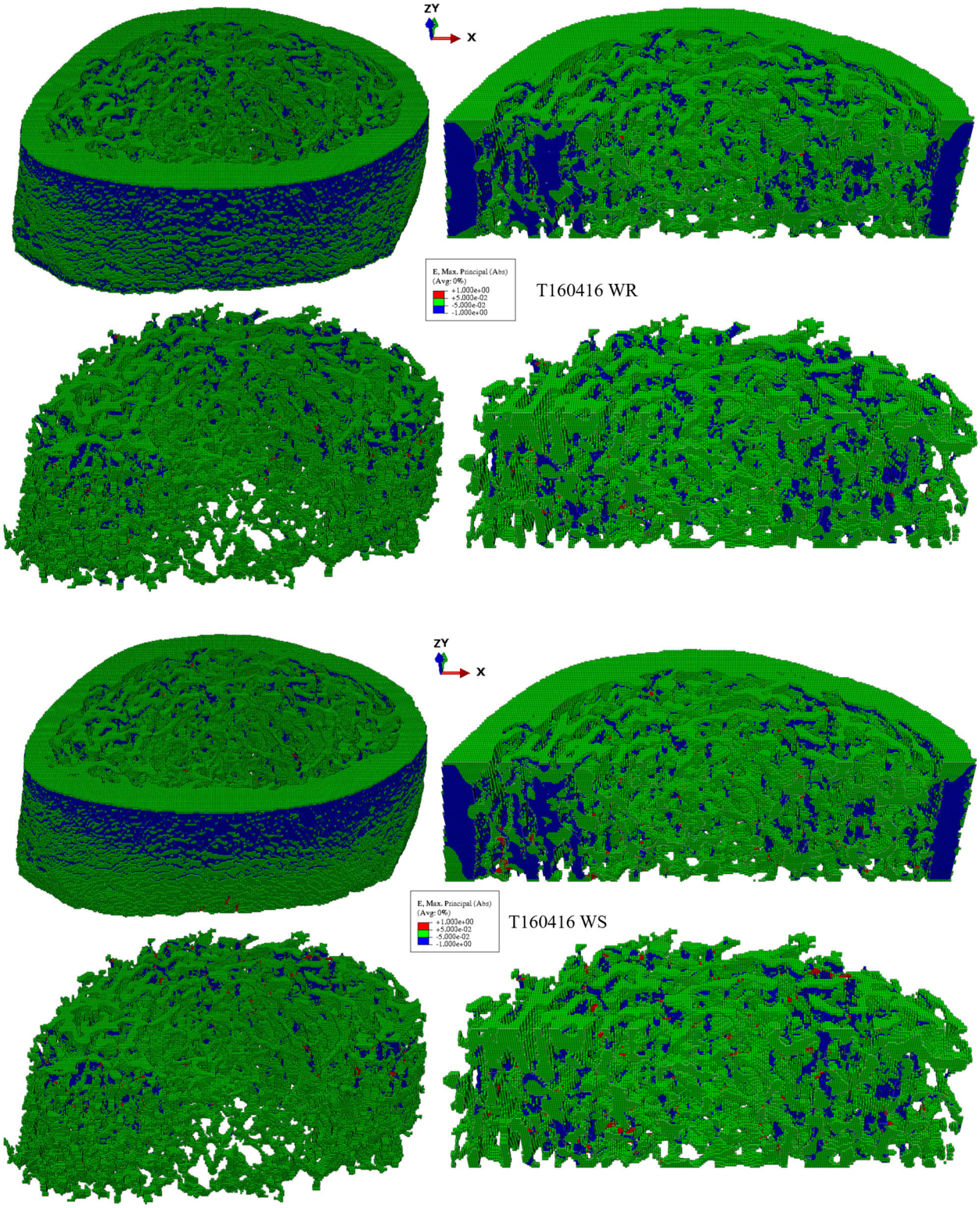


Figure 61. All Contour Plots for T160416 Whole Specimen Compression (WSC).
WR = Whole Rough, WS = Whole Smooth

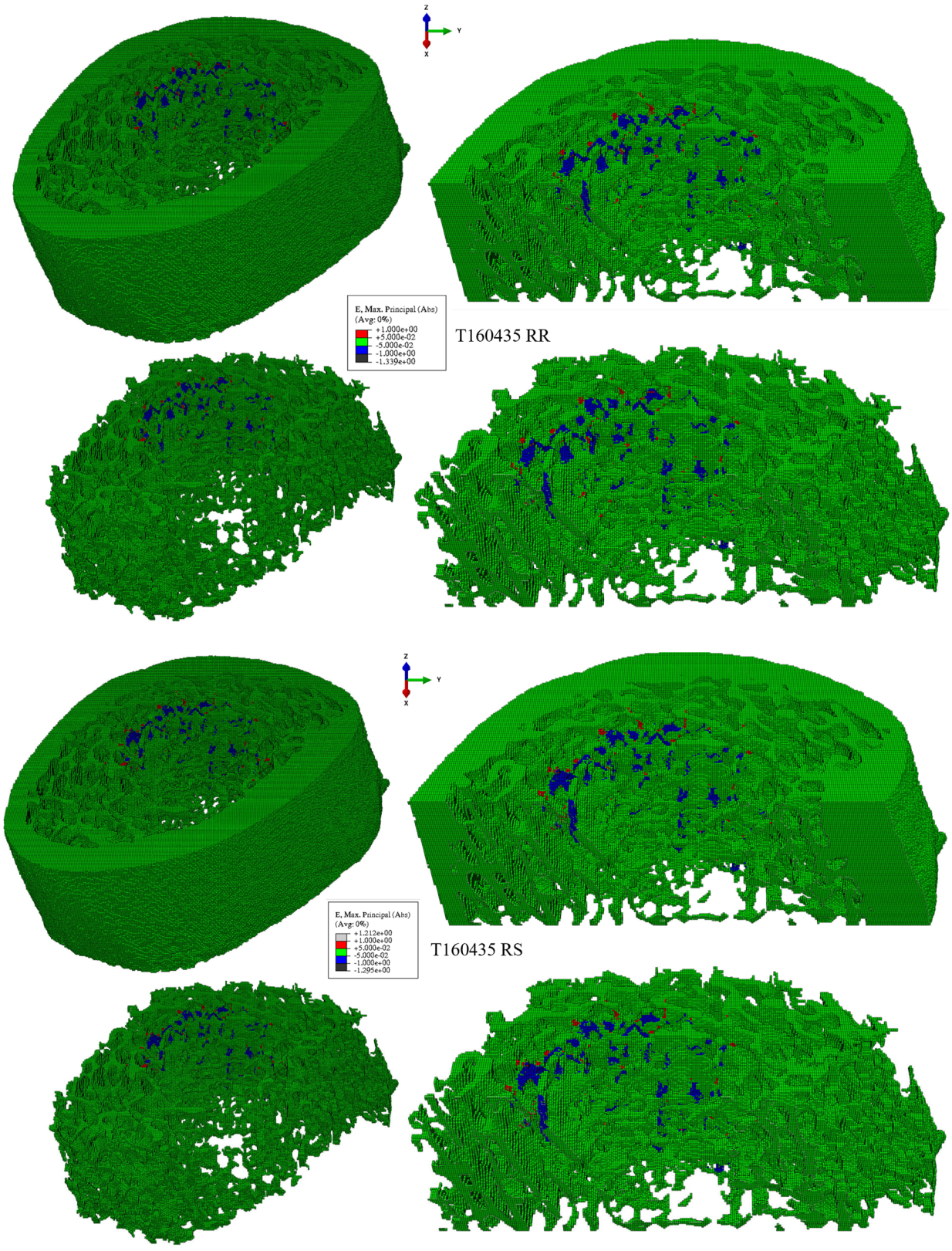


Figure 62. All Contour Plots for T160435 Reduced Platen Compression (RPC).
RR = RPC Rough, RS = RPC Smooth.

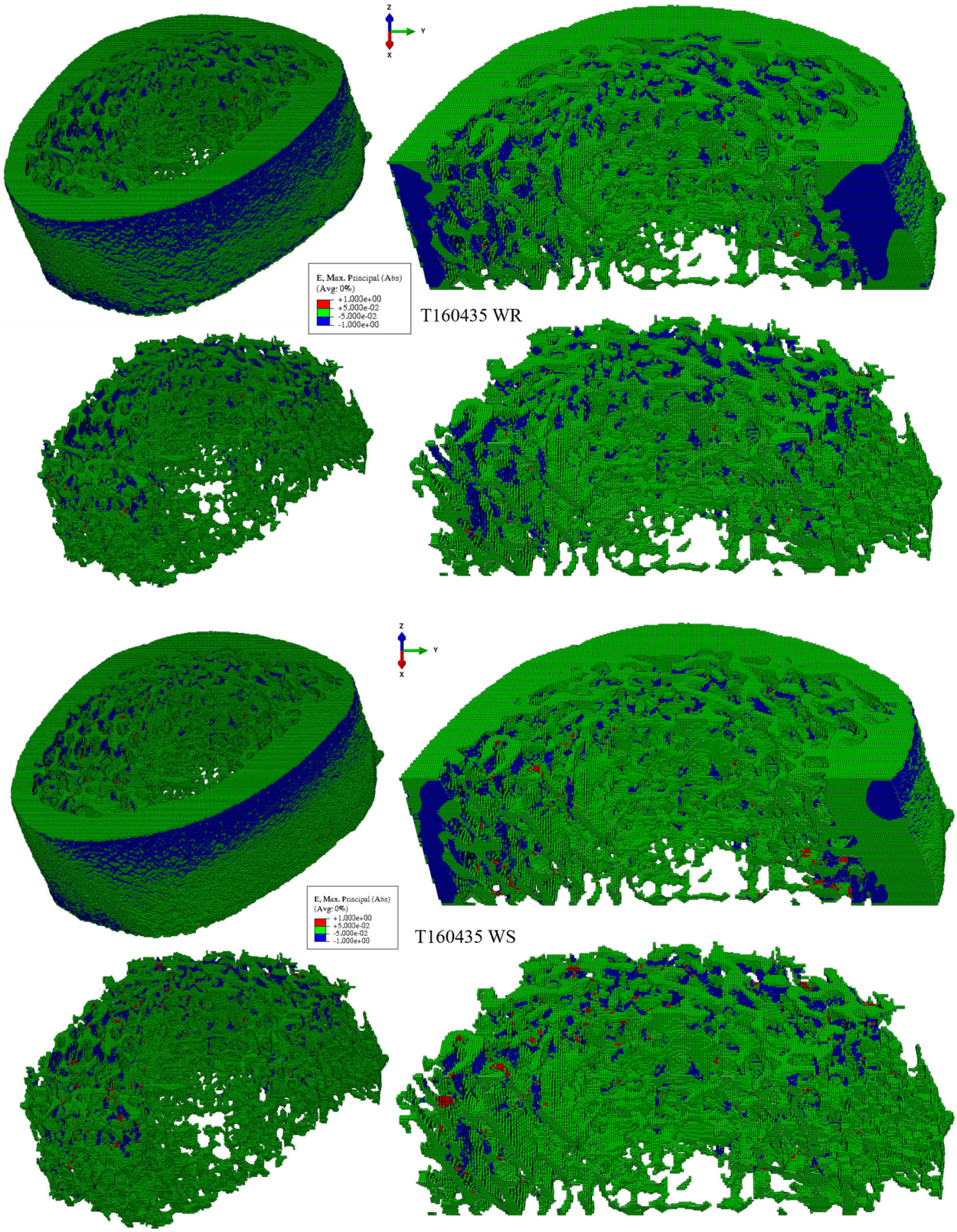


Figure 63. All Contour Plots for T160435 Whole Specimen Compression (WSC).
WR = Whole Rough, WS = Whole Smooth

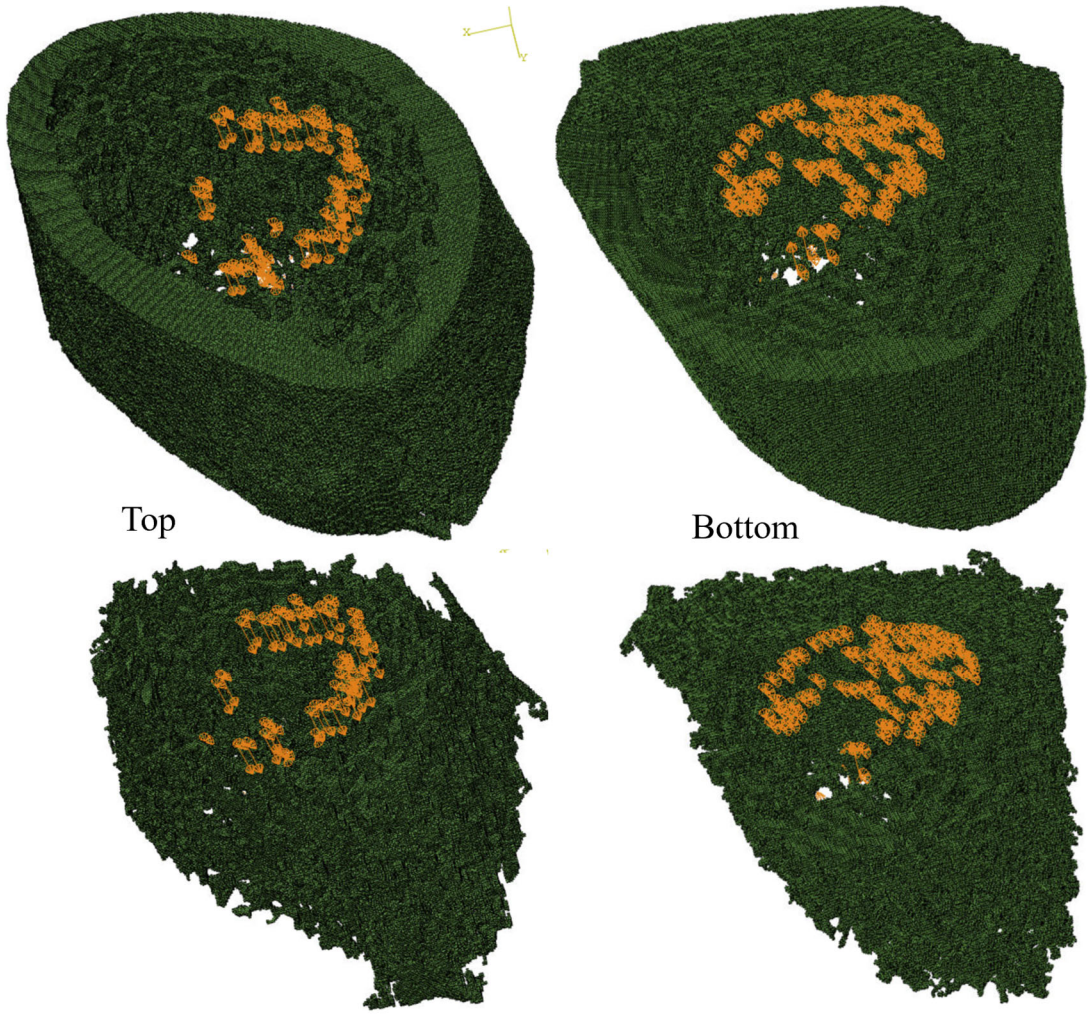


Figure 64. Boundary Conditions for RPC Illustrated on Sample Bone.
Distal femur metaphysis from adult male rat is shown, top represents the most proximal transverse surface. Orange cones represent a fixed degree of freedom.

CT_To_Volume

```
% CT_To_Volume:
%
% This script reads a series of grayscale images and stacks them, binarizes
% the volume, and applies a connectivity filter to isolate a single
% connected body.
%
% Output V_conn is a 3D binary image.

%--------------------------------------------------------------------------
% User Input Parameters:

% Specify folder of images:
Folder_Path = 'D:\00000610\TIFF\';
Image_Leader = 'C0000628_';  %Leading characters in file from CT export
First_Image = 0140;
Last_Image = 0239;

% Cropping dimensions:  specify extents of cropping - these can be
% identified by probing values in the image viewer or using imshow().
xlow = 161;
xhigh = 513;
ylow = 569;
yhigh = 846;

%--------------------------------------------------------------------------
% Crop and Stack Images:

filepath = append(Folder_Path,Image_Leader);

count = 0;
Image_Number = First_Image;
Layer = 1;

A = zeros(yhigh-ylow,xhigh-xlow+1,'uint8');
[m,n] = size(A);
V = zeros(m,n,(Last_Image-First_Image+1),'uint8');

for i = First_Image:Last_Image

    %change filename to open each image
    filename = append(filepath,num2str(Image_Number, '%04.4i'));
    Z = imread(filename,'tif');

    %select only the desire pixels:
    for j = 1:(yhigh - ylow)
        A(j,:) = Z(ylow + (j-1), xlow:xhigh);
```

```
    end

    %stack each cropped image into a 3D volume:
    V(:,:,Layer) = A;
    Image_Number = Image_Number + 1;
    Layer = Layer + 1;

end

%-------------------------------------------------------------------------
% Binarization:

V_seg = imbinarize(V); % Otsu's Method

%-------------------------------------------------------------------------
% Keep only the largest section of connected voxels

V_conn = Conn_Filt(V_seg);
```

Published with MATLAB® R2019b

Conn_Filt

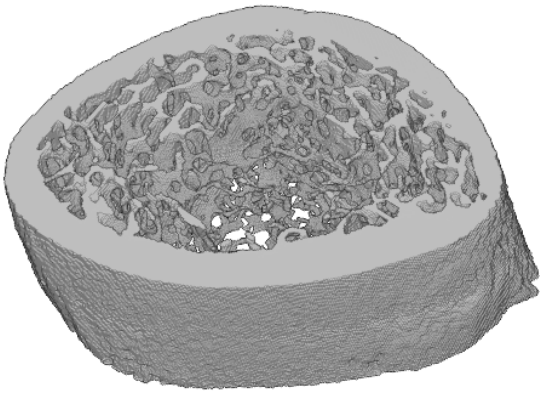
```
function Conn = Conn_Filt(V)
% Conn_Filt takes a 3D binary image and imposes face connectivity to
% isolate the single largest connected body.
%
%   Conn = Conn_Filt(V) returns the single largest connected component in
%   the 3D binary image V.
%
%   Conn is a 3D binary image the same size as V.

CC = bwconncomp(V, 6);

numPixels = cellfun(@numel, CC.PixelIdxList); %find number of pixels in
                                              %connected component.
[~, idx] = max(numPixels);  %index of largest connected component.

Conn = zeros(size(V), 'logical');

Conn(CC.PixelIdxList{idx}) = 1; %new image isolating largest connected body

end
```

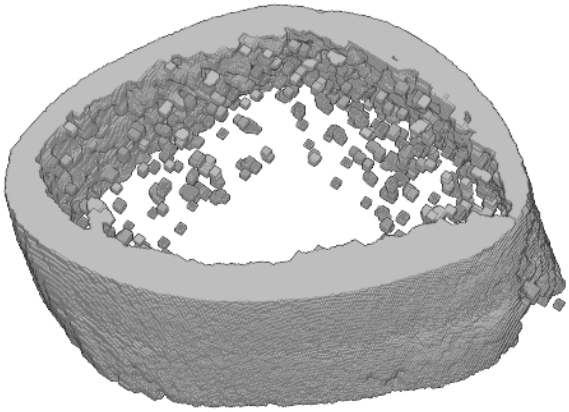
Published with MATLAB® R2019b

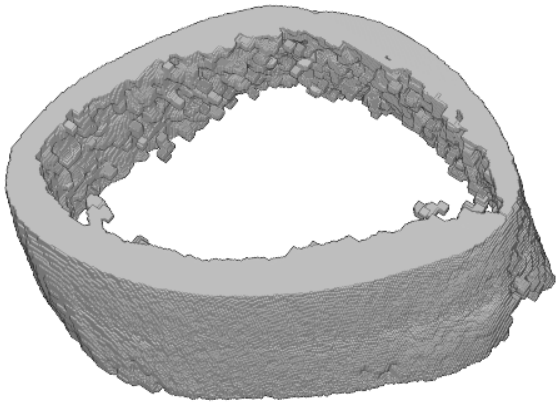
Isolate_Trab

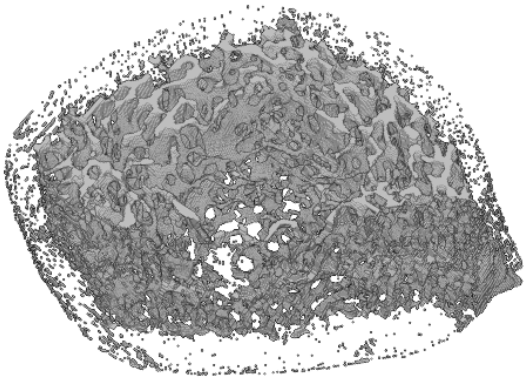
```
%Isolate_Trab:
% This script takes a connected volume of a bone and applies image opening
% functions to isolate the trabecular structure.

E = V_conn; %Connected volume

se = strel('cube',5);  % 5x5x5 cubic structural element.
E_open = imopen(E,se);

E_open_conn = Conn_Filt(E_open);  %Isolated cortical shell.

G = E-E_open_conn; %subtract cortical shell from original image to isolate
    % trabecular structure.

Trab = Conn_Filt(G); %apply connectivity filter again to ensure single
    % body for use in FEA.

%--------------------------------------------------------------------------
%display for pubilcation:
config

figure
E_ = volshow(E,config);
figure
E_open_ = volshow(E_open,config);
figure
E_open_conn_ = volshow(E_open_conn,config);
figure
G_ = volshow(G,config);
figure
Trab_ = volshow(Trab,config);
```

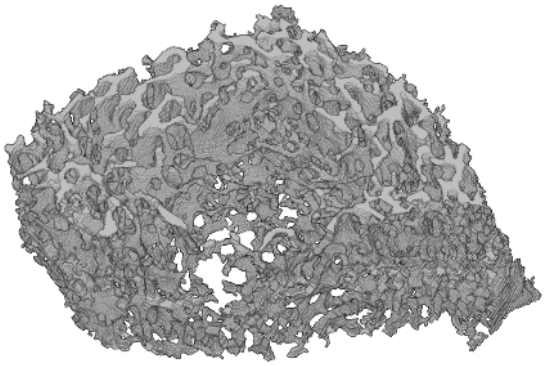
```
config =

  struct with fields:

     CameraPosition: [1.4699 0.9159 3.1322]
     CameraUpVector: [-0.4824 -0.3667 0.7955]
       CameraTarget: [0 0 0]
    CameraViewAngle: 15
    BackgroundColor: [1 1 1]
           Renderer: 'VolumeRendering'
           Colormap: [256×3 double]
           Alphamap: [256×1 double]
           Lighting: 0
    IsosurfaceColor: [1 0 0]
           Isovalue: 0.4900
```









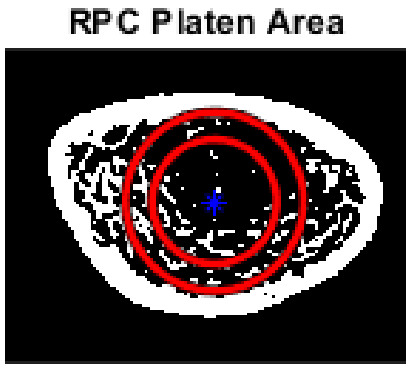


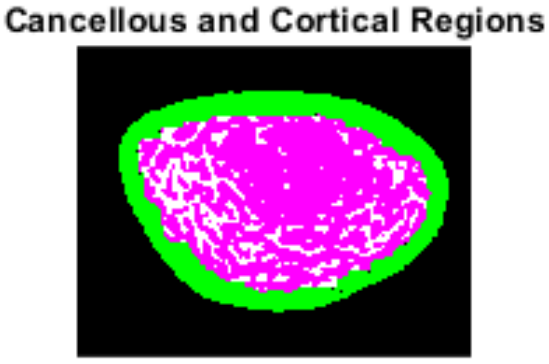
Published with MATLAB® R2019b

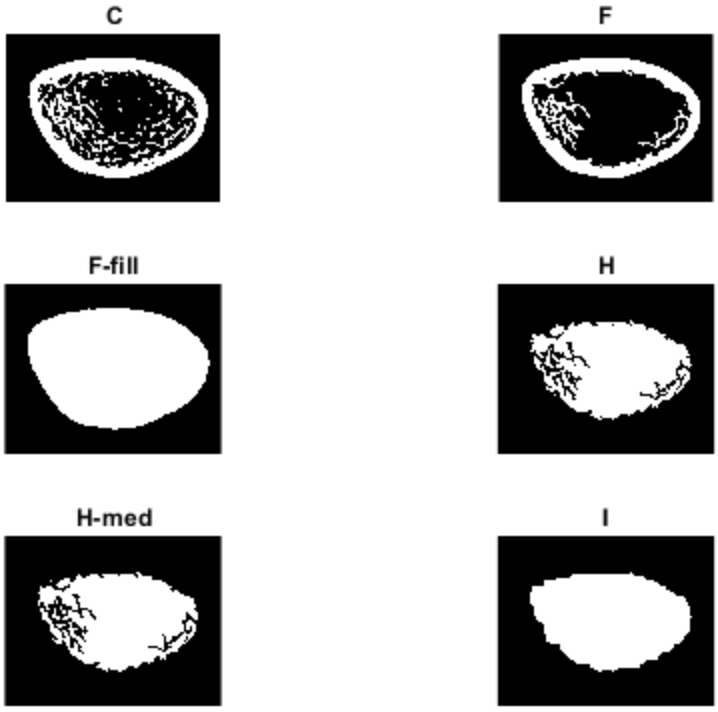
Find_RPC_Area

```
%Find_RPC_Area:
%   This script takes a binary image of the top of the RPC specimen and
%   determines the RPC contact area size and location.  The algorithm finds
%   the centroid of the cancellous region and minor diameter, which
%   approximates the diameter of the endocortical cirlce. 70% of this
%   diameter is used for the platen sizing. Figures aregenerated to show
%   the areas.

CC = bwconncomp(C, 4); % C is the binary image of the top slice. Imposes
                       % connectivity.

numPixels = cellfun(@numel, CC.PixelIdxList);
[biggest, idx] = max(numPixels);

F = zeros(size(C));

F(CC.PixelIdxList{idx) = 1; % largest connected component, rough cortical
                            % shell.

se = strel('square',15);  % a 15x15 square strcutural element.
F_fill = imfill(F,'holes'); % solid filled total area.
H = F_fill-F; % rough cancellous area.
H_med = medfilt2(H); % filtering to smooth cancellous area.
I = imclose(H_med,se); % closing to fill in shape of cancellous area.

s = regionprops(I,'centroid', 'MajorAxisLength', 'MinorAxisLength');
centroids = cat(1,s.Centroid); % centroid of cancellous areas.

figure % shows the endocortical circle and RPC area
h = imshow(C);
hold on
axis = h.Parent;
plot(centroids(:,1),centroids(:,2),'b*')
viscircles(s.Centroid, (s.MinorAxisLength)/2*1); %Endocortical circle
viscircles(s.Centroid, (s.MinorAxisLength)/2*0.7); %Platen Area
hold off
title('RPC Platen Area')

figure % highlights cancellous and cortical regions.
imshowpair(C,I)
title('Cancellous and Cortical Regions')

figure %illustrate images throughout process
subplot(3,2,1)
imshow(C)
title('C')
subplot(3,2,2)
imshow(F)
title('F')
subplot(3,2,3)
```

```
imshow(F_fill)
title('F-fill')
subplot(3,2,4)
imshow(H)
title('H')
subplot(3,2,5)
imshow(H_med)
title('H-med')
subplot(3,2,6)
imshow(I)
title('I')
```







Published with MATLAB® R2019b

Matrix_Generation

```
% Matrix_Generation:
% This script takes a 3D binary image and generates the Nodal Coordinate
% Array and Element Connectivity Array. Node set arrays for use in applying
% Boundary Conditions are generated as well.
% The desired volume to be used should already be loaded in the workspace.
% This volume is reassinged to V_w for use in this script. The user needs
% to change the specimen number and RPC location information.

%--------------------------------------------------------------------------
% User Input Parameters:

V_w = Trab; % Volume to be written to input file.
pixel_width = 0.02; % width of the voxels in [mm].

Specimen_Number = 'T160416_Trab';
    % Use the T-number from the CT scan for the naming convention of the
    % input files.  Add '_Trab' if the volume has been remade without
    % the cortex.

Layers = 3; % Number of layers in node sets on top and bottom
            % for generating BCs.

RPC_centroid = 0.02*[184.9 135.4];  %centroid in x and y coordinates of
                                    %the RPC platen contact area.
RPC_radius = 0.02*54.9;  %radius of the RPC contact area.

%--------------------------------------------------------------------------
% Generate NC, EC

NC = Node_Coord(V_w, pixel_width);
EC = Elem_Conn(V_w);

% Node Set matrices

[z_min, z_max] = Layer_Nodes(NC, Layers); %top and bottom layers

[RPC_z_min, Bottom_Centroid_Node] = ...
    RPC_Nodes(z_min, RPC_centroid, RPC_radius); %bottom RPC contact area

[RPC_z_max, Top_Centroid_Node] = ...
    RPC_Nodes(z_max, RPC_centroid, RPC_radius); %top RPC contact area
```

Published with MATLAB® R2019b

Node_Coord

```
function [Sorted] = Node_Coord(V, pixel_width)
% This function writes the nodal coordinate array, where the first column
% is the node number, followed by the x, y, and z coordinates.
%   V is the assembled image (logical matrix)
%   The sum of the values for the elements the node touches is calculated.
%       If this sum is any number other than zero, this means the node is
%       in an active element(s), so the node is written to the input file.

%-------------------------------------------------------------------------
% Initial Variables:

[m,n,p] = size(V);
M = m+1;
N = n+1;
P = p+1;

pw = pixel_width;

col = 1; %initialize first column count
%-------------------------------------------------------------------------
% Corners
%Check the corner nodes for element contact
%Corner nodes will contact 1 element
%8 corners need to be checked individually.


% (0,0,0)
I = 1;
J = 1;
K = 1;

if V(I,J,K) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end

%(xmax,0,0)
I = 1;
J = N;
K = 1;

if V(I,J-1,K) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end

%(0,ymax,0)
I = M;
J = 1;
K = 1;
```

```
if V(I-1,J,K) == 1
       Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
       col = col+1;
end

%(xmax,ymax,0)
I = M;
J = N;
K = 1;

if V(I-1,J-1,K) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end

%(0,0,zmax)
I = 1;
J = 1;
K = P;

if V(I,J,K-1) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end


%(xmax,0,zmax)
I = 1;
J = N;
K = P;

if V(I,J-1,K-1) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end


%(0,ymax,zmax)
I = M;
J = 1;
K = P;

if V(I-1,J,K-1) == 1
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end

%(xmax,ymax,zmax)
I = M;
J = N;
K = P;

if V(I-1,J-1,K-1) == 1
```

```
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
end

%-------------------------------------------------------------------------
% Edges
%Check edges of the stacked image
%Edge nodes will contact 2 elements
%Regrettably, there are 12 edges, each of which must be treated
%   individually.

% x-axis, y=0, z=0
I = 1;
    % J is the looping index
K = 1;

for J = 2:(N-1)
    tracker = V(I,J-1,K) + V(I,J,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% x-axis, y=ymax, z=0
I = M;
    % J is the looping index
K = 1;

for J = 2:(N-1)
    tracker = V(I-1,J-1,K) + V(I-1,J,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% y-axis, x=0, z=0
    % I is the looping index
J = 1;
K = 1;

for I = 2:(M-1)
    tracker = V(I-1,J,K) + V(I,J,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% y-axis, x=xmax, z=0
    % I is the looping index
J = N;
K = 1;
```

```
for I = 2:(M-1)
    tracker = V(I-1,J-1,K) + V(I,J-1,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% x-axis, y=0, z=zmax
I = 1;
    % J is the looping index
K = P;

for J = 2:(N-1)
    tracker = V(I,J-1,K-1) + V(I,J,K-1);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% x-axis, y=ymax, z=zmax
I = M;
    % J is the looping index
K = P;

for J = 2:(N-1)
    tracker = V(I-1,J-1,K-1) + V(I-1,J,K-1);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% y-axis, x=0, z=zmax
    % I is the looping index
J = 1;
K = P;

for I = 2:(M-1)
    tracker = V(I-1,J,K-1) + V(I,J,K-1);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% y-axis, x=xmax, z=zmax
    % I is the looping index
J = N;
K = P;

for I = 2:(M-1)
```

```
    tracker = V(I-1,J-1,K-1) + V(I,J-1,K-1);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% z-axis, x=0, y=0
I = 1;
J = 1;
    %K is the looping index

for K = 2:(P-1)
    tracker = V(I,J,K-1) + V(I,J,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% z-axis, x=xmax, y=0
I = 1;
J = N;
    %K is the looping index

for K = 2:(P-1)
    tracker = V(I,J-1,K-1) + V(I,J-1,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% z-axis, x=0, y=ymax
I = M;
J = 1;
    %K is the looping index

for K = 2:(P-1)
    tracker = V(I-1,J,K-1) + V(I-1,J,K);
    if tracker > 0
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end

% z-axis, x=xmax, y=ymax
I = M;
J = N;
    %K is the looping index

for K = 2:(P-1)
    tracker = V(I-1,J-1,K-1) + V(I-1,J-1,K);
    if tracker > 0
```

```
        Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
        col = col+1;
    end
end
%-------------------------------------------------------------------------
% Faces
%Check faces of stacked image
%Face nodes will contact 4 elements
%There are 6 faces to be checked individually, which is less than 12.

% z=0
    % I and J are looping indices
K = 1;

for J = 2:(N-1)
    for I = 2:(M-1)
        tracker = V(I-1,J-1,K) + V(I-1,J,K) ...
            + V(I,J-1,K) + V(I,J,K);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end

% z=zmax
    % I and J are looping indices
K = P;

for J = 2:(N-1)
    for I = 2:(M-1)
        tracker = V(I-1,J-1,K-1) + V(I-1,J,K-1) ...
            + V(I,J-1,K-1) + V(I,J,K-1);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end

% y=0
    % I and K are looping indices
J = 1;

for K = 2:(P-1)
    for I = 2:(M-1)
        tracker = V(I-1,J,K-1) + V(I,J,K-1) ...
            + V(I-1,J,K) + V(I,J,K);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end
```

```
% y=ymax
    % I and K are looping indices
J = N;

for K = 2:(P-1)
    for I = 2:(M-1)
        tracker = V(I-1,J-1,K-1) + V(I,J-1,K-1) ...
            + V(I-1,J-1,K) + V(I,J-1,K);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end

% x=0
    % J and K are looping indices
I = 1;

for K = 2:(P-1)
    for J = 2:(N-1)
        tracker = V(I,J-1,K-1) + V(I,J,K-1) ...
            + V(I,J-1,K) + V(I,J,K);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end

% x=xmax
   % J and K are looping indices
I = M;

for K = 2:(P-1)
    for J = 2:(N-1)
        tracker = V(I-1,J-1,K-1) + V(I-1,J,K-1) ...
            + V(I-1,J-1,K) + V(I-1,J,K);
        if tracker > 0
            Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
            col = col+1;
        end
    end
end
%-------------------------------------------------------------------------
% Volume
%Check remaining internal volume nodes for element contact
%Volume nodes will touch 8 elements
%There is only 1 volume, a welcome reprieve

for K = 2:(P-1)
    for I = 2:(M-1)
        for J = 2:(N-1)
```

```
            tracker = sum(V((I-1):I,(J-1):J,(K-1):K),'all');
                %this takes the sum of the 2x2 matrix surrounding the node
            if tracker > 0
                Node_Coord(:,col) = Node_Mat(I, J, K, M, N, pw);
                col = col+1;
            end
        end
    end
end
%-------------------------------------------------------------------------
% Sort Nodal Coordinate Matrix

Sorted = sortrows(Node_Coord');

end %end the first function
%-------------------------------------------------------------------------

% Subfunction Node_Mat
%-------------------------------------------------------------------------
% The following subfunction is to save space in writing the above function,
% as it is called so often.

function Col = Node_Mat(I, J, K, M, N, pw)
    N_n = (I-1)*N + J + M*N*(K-1); %Node Number
    x_c = pw*(J-1);
    y_c = pw*(I-1);
    z_c = pw*(K-1);
    Col = [N_n; x_c; y_c; z_c];
%     fprintf(fileID,'%8i, %8.4f, %8.4f, %8.4f\n', N_n, x_c, y_c, z_c);
end
```

Published with MATLAB® R2019b

Elem_Conn

```
function [Elem_conn] = Elem_Conn(V)
% This function writes the element connectivity array, where the first
% column is the element number, followed by the 8 nodes that make up that
% element.
%   V is the assembled image (logical matrix)

[m,n,p] = size(V);
E_num = sum(V(:)); %total number of elements.

Elem_conn = zeros(E_num,9); %initialize for speed.

row = 1;
for k = 1:p
    for i = 1:m
        for j = 1:n
            if V(i,j,k) == 1
            Elem_conn(row,1) = (i-1)*n + j + m*n*(k-1); %Element Number
            Elem_conn(row,2) = j+(n+1)*(i-1) + (m+1)*(n+1)*(k-1);
            Elem_conn(row,3) = Elem_conn(row,2) + 1;
            Elem_conn(row,5) = Elem_conn(row,2) + (n+1);
            Elem_conn(row,4) = Elem_conn(row,3) + (n+1);
            % 5 and 4 are switched for Abaqus input order
            Elem_conn(row,6) = Elem_conn(row,2) + (m+1)*(n+1);
            Elem_conn(row,7) = Elem_conn(row,3) + (m+1)*(n+1);
            Elem_conn(row,8) = Elem_conn(row,4) + (m+1)*(n+1);
            Elem_conn(row,9) = Elem_conn(row,5) + (m+1)*(n+1);
            row = row + 1;
            end
        end
    end
end

end
```

Published with MATLAB® R2019b

Layer_Nodes

```
function [z_min,z_max] = Layer_Nodes(NC, Layers)
% Layer_Nodes generates nodal coordinate matrices for the top and bottom
%   layers of a given nodal coordinate matrix.
%
%   [z_min,z_max] = Layer_Nodes(NC, Layers) for the nodal coordinate array
%   NC returns the new nodal coordinate arrays z_min and z_max for the
%   number of Layers specified.
%
%   NC is a Nx4 array, where the first term of each row is the node number,
%   and the following terms are the x, y, and z coordinate of the node,
%   respectively.  NC must already be sorted by Node Number. *If the nodal
%   coordinate matrix was generated using Node_Coord(), the return value
%   is already sorted.
%
%   Layers is an integer.
%
%   z_min and z_max are Nx4 arrays.  z_min starts at z=0 (lowest node
%   numbers) and goes the specified number of layers.  z_max does the same
%   but ending at z-max (highest node numbers).
%
%   The nodal coordinates remain in these matrices for use with other
%   functions in determining RPC Platen area nodes.  Only the node numbers
%   are required for node set generation in Abaqus.
%-------------------------------------------------------------------------
D = size(NC);

[~, ia, ~] = unique(NC(:,4),'last'); %find unique values of z coordinate

idx1 = ia(Layers); % use index of last value for desired layers

z_min = NC(1:idx1,:); %Assign  relevant rows of Nodal Coordinate matrix
    % to new matrix.

max1 = length(ia); %total number of layers
idx2 = ia(max1 - (Layers)); %index of z-max layers

z_max = NC((idx2+1):D(1),:); %reassign values for z-max layers to new
%   matrix

end
```

Published with MATLAB® R2019b

RPC_Nodes

```
function [RPC_circle, Closest_Node] = RPC_Nodes(NC,C, R)
% RPC_Nodes takes the nodal coordinate matrices for the boundary condition
% layers and makes new nodal coordiate arrays for the RPC contact areas.
%
%   [RPC_circle, Closest_Node] = RPC_nodes(NC,C, R) returns a Nx4 nodal
%   coordinate array inside a circle of radius R around Centroid C from
%   the layer data in NC. Closest_Node is the number of the node closest to
%   the center of the RPC contact area, for use in the smooth BCs.
%
%   NC and an Nx4 nodal coordinate array.
%
%   C is a 2x1 array with the x and y component of the centroid on the
%   desired face.
%
%   R is a numerical value for the radius of the desired circle.
%-------------------------------------------------------------------------

x_c = C(1); %x-coordinate of centroid
y_c = C(2); %y-coordinate of centroid

r = ((NC(:,2)-x_c).^2 + (NC(:,3)-y_c).^2).^(1/2); %radial distance from
    % each node to the assigned centroid.

[~, i] = min(r); % index of closest node.
Closest_Node = NC(i,1);

Idx = (r <= R);  % logical index of nodes that are inside the platen area.

RPC_circle = NC(Idx,:);

end
```

Published with MATLAB® R2019b

Inputs_Generation

```
%Inputs_Generation:
% This script calls the functions to write the input files for Abaqus and
% the ADA cluster of TAMU HPRC. The user needs to change the values for
% material properties and computational specifications.

%-------------------------------------------------------------------------
% User Input Parameters:

% Element Material Properties:

Young = 10e9; % Young's Modulus in [Pa]
Poisson = 0.3; % Poisson's Ratio

% Ada Input Parameters:

wall_clock = '1:00'; % enforced wallclock for simulations.
mem = 15000; % amount of memory reserved in MB.

%-------------------------------------------------------------------------
% Write inputs

% Whole, Rough
Spec = 'Whole';
Fric = 'Rough';
job = append(Specimen_Number, '_WR');
Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
    Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
    Bottom_Centroid_Node, Top_Centroid_Node)
Ada_Write(job, wall_clock, mem);

% Whole, Smooth
Spec = 'Whole';
Fric = 'Smooth';
job = append(Specimen_Number, '_WS');
Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
    Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
    Bottom_Centroid_Node, Top_Centroid_Node)
Ada_Write(job, wall_clock, mem);

% RPC, Rough
Spec = 'RPC';
Fric = 'Rough';
job = append(Specimen_Number, '_RR');
Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
    Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
    Bottom_Centroid_Node, Top_Centroid_Node)
Ada_Write(job, wall_clock, mem);

% RPC, Smooth
Spec = 'RPC';
```

```
Fric = 'Smooth';
job = append(Specimen_Number, '_RS');
Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
    Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
    Bottom_Centroid_Node, Top_Centroid_Node)
Ada_Write(job, wall_clock, mem);

%--------------------------------------------------------------------------
% Save workspace variable to .mat file for future use and reference.

clear Fric job Spec
save(append(Specimen_Number, '.mat'));
```

Published with MATLAB® R2019b

Input_Write

```
function [] = Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
    Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
    Bottom_Centroid_Node, Top_Centroid_Node)
%Input_Write writes an input file for Abaqus. A very large text file will
%be generated and saved in the working directory when this function is
%called.  Format information can be found in Abaqus documentation.
%
%   Input_Write(job, Specimen_Number, Spec, Fric, Young, ...
%   Poisson, NC, EC, z_min, z_max, RPC_z_min, RPC_z_max, ...
%   Bottom_Centroid_Node, Top_Centroid_Node)
%
%   job, Specimen_Number, Spec, Fric are strings that specify the filename
%   to be created and the type of boundary conditions to be applied.
%
%   Young, Poisson are numeric values for the material properties.
%
%   NC, EC are the nodal coordinate and element connectivity arrays.
%
%   z_min, z_max, RPC_z_min, RPC_z_max are nodal coordinate arrays for the
%   boundary condition node sets.
%
%   Bottom_Centroid_Node, Top_Centroid_Node are the node numbers for the
%   closest nodes to the RPC area centroid on their respective faces.
%------------------------------------------------------------------------

%------------------------------------------------------------------------
% Create File:
filename = append(job, '.inp');
fileID = fopen(filename, 'w');

%------------------------------------------------------------------------
% Heading:

fprintf(fileID, '*Heading\n');
fprintf(fileID, '** Specimen Number: %s\n', Specimen_Number);
fprintf(fileID, '** Model Type: %s %s\n', Spec, Fric);
fprintf(fileID, '** Generated by: Matlab code Input_Write.\n');
fprintf(fileID, '** %s\n', char(datetime));
fprintf(fileID, '**\n');
fprintf(fileID, '*Preprint, echo=NO, model=NO, history=NO, contact=NO\n');

%------------------------------------------------------------------------
% Part Definition with nodes and elements

fprintf(fileID, '**\n** PARTS\n**\n');
fprintf(fileID, '*Part, name=Part-1\n'); %'Part-1' is default,
    %make sure consistent in the instance setion of assembly if changed.

fprintf(fileID,'*Node, nset=Whole_Body_Nodes\n');
fprintf(fileID,'%8i, %8.4f, %8.4f, %8.4f\n', NC');
```

```
fprintf(fileID,'*Element, type=C3D8, elset=Whole_Body_Elements\n');
fprintf(fileID,...
          '%8i, %8i, %8i, %8i, %8i, %8i, %8i, %8i, %8i\n',EC');

fprintf(fileID, '** Section: Section-1\n');
fprintf(fileID,...
    '*Solid Section, elset=Whole_Body_Elements, material=Bone\n,\n');
fprintf(fileID, '*End Part\n');

%------------------------------------------------------------------------
% Assembly

fprintf(fileID, '**\n**\n** ASSEMBLY\n**\n');
fprintf(fileID, '*Assembly, name=Assembly\n**\n');
fprintf(fileID, '*Instance, name=Part-1-1, part=Part-1\n');
fprintf(fileID, '*End Instance\n');
fprintf(fileID, '**\n');

Write_Nset(fileID, 'z_min_Nodes', z_min(:,1));
Write_Nset(fileID, 'z_max_Nodes', z_max(:,1));

Write_Nset(fileID, 'RPC_z_min_Nodes', RPC_z_min(:,1));
Write_Nset(fileID, 'RPC_z_max_Nodes', RPC_z_max(:,1));

Write_Nset(fileID, 'Bottom_Centroid_Node', Bottom_Centroid_Node);
Write_Nset(fileID, 'Top_Centroid_Node', Top_Centroid_Node);

fprintf(fileID, '*End Assembly\n');

%------------------------------------------------------------------------
% Materials

fprintf(fileID, '**\n** MATERIALS\n**\n');
fprintf(fileID, '*Material, name=Bone\n');
fprintf(fileID, '*Elastic\n %8.4f, %4.3f\n', Young, Poisson);
fprintf(fileID, '** ----------------------------------------------\n');

%------------------------------------------------------------------------
% Step

fprintf(fileID, '**\n** STEP: Step-1\n**\n');
fprintf(fileID, '*Step, name=Step-1, nlgeom=NO, solver=iterative\n');
fprintf(fileID, '*Static\n1., 1., 1e-05, 1.\n'); % these are the default
        %options as specified in abaqus.

%------------------------------------------------------------------------
% Boundary Conditions

fprintf(fileID, '**\n** BOUNDARY CONDITIONS\n**\n');

Boundary_Condition(fileID,Spec,Fric);
```

```
%-------------------------------------------------------------------------
% Outputs

fprintf(fileID, '**\n**OUTPUT REQUESTS\n**\n');
% fprintf(fileID, '*Restart, write, frequency=0\n');
fprintf(fileID, '**\n** FIELD OUTPUT: F-Output-1\n**\n');
fprintf(fileID, '*Output, field, variable=PRESELECT\n');
fprintf(fileID, '**\n** HISTORY OUTPUT: H-Output-1\n**\n');
fprintf(fileID, '*Output, history, variable=PRESELECT\n');
% fprintf(fileID, '**\n** Output stress-strain values to .dat file\n**\n');
% fprintf(fileID, '*EL PRINT, POSITION=CENTROIDAL\nS, MISES\nE\n');
% fprintf(fileID, '**\n** Output RF values to .dat file\n**\n');
% fprintf(fileID, '*NODE PRINT\nRF\n**\n');
fprintf(fileID, '*End Step\n');

fclose(fileID);

end
%-------------------------------------------------------------------------
% Subfunction for writing node sets.
function [] = Write_Nset(fileID, Nset_Name, Node_List)

fprintf(fileID, '*Nset, nset=%s, instance=Part-1-1\n', Nset_Name);
fprintf(fileID, '%10i, %10i, %10i, %10i, %10i, %10i\n', Node_List);
fprintf(fileID, '\n**\n'); %New line in case Node_List doesn't make it all
    % the way to a new line.

end

%-------------------------------------------------------------------------
% Subfuncntion for writing Boundary Conditions

function [] = Boundary_Condition(fileID, Spec, Fric)

switch Spec
    case 'Whole'
        Top = 'z_max_Nodes';
        Bottom = 'z_min_Nodes';
    case 'RPC'
        Top = 'RPC_z_max_Nodes';
        Bottom = 'RPC_z_min_Nodes';
end

switch Fric
%-------------------------------------------------------------------------
    case 'Rough'
% Rough

    % top face displacement 0.1mm in z direction.
fprintf(fileID, '** Top face displacement\n');
fprintf(fileID, '*Boundary\n');
fprintf(fileID, '%s, 1, 2\n', Top);
fprintf(fileID, '%s, 3, 3, -0.1\n', Top);
```

```
    % fix bottom face in all 3 directions.
fprintf(fileID, '** Bottom face fixing\n');
fprintf(fileID, '*Boundary\n');
fprintf(fileID, '%s, 1, 3\n', Bottom);
%------------------------------------------------------------------------

    case 'Smooth'
% Smooth

    % top face displacement 0.1mm in z direction.
fprintf(fileID, '** Top face displacement\n');
fprintf(fileID, '*Boundary\n');
fprintf(fileID, '%s, 3, 3, -0.1\n', Top);

    % fix bottom face in z direction.
fprintf(fileID, '** Bottom face fixing\n');
fprintf(fileID, '*Boundary\n');
fprintf(fileID, 'Bottom_Centroid_Node, 1, 2\n');
fprintf(fileID, '%s, 3, 3\n', Bottom);
end

end
```

Published with MATLAB® R2019b

Ada_Write

```
function [] = Ada_Write(job, wallclock, memory)
%Ada_Write writes a .txt file with the necessary information to submit an
%Abaqus input file to be solved on the ADA cluster of TAMU HPRC.  Format
%information can be found on the ADA wiki at hprc.tamu.edu.
%
%   job is a string that will be the filename.
%
%   wallclock is a string that sets the time limit for the simulation.
%
%   memory is a numeric value for the requested memory in MB.
%
filename = append(job, '.txt');
input = append(job, '.inp');
fileID = fopen(filename, 'w');

fprintf(fileID, '#BSUB -J AbaqusJob\n');
fprintf(fileID, '#BSUB -L /bin/bash\n');
fprintf(fileID, '#BSUB -W %s\n', wallclock);
fprintf(fileID, '#BSUB -n 1\n');
fprintf(fileID, '#BSUB -R "span[ptile=1]" \n');
fprintf(fileID, '#BSUB -R "rusage[mem=%i]" \n', memory);
fprintf(fileID, '#BSUB -M %i\n', memory);
fprintf(fileID, '#BSUB -o Abaqus.%%J\n');
fprintf(fileID, '\n\n## Load the modules\n');
fprintf(fileID, 'module load ABAQUS/2018\n');
fprintf(fileID, '\n## Launch Abaqus with proper parameters\n');
fprintf(fileID, 'abaqus memory="%iMB" cpus=1 job=%s input=%s', ...
    memory, job, input);

fclose(fileID);
end
```

Published with MATLAB® R2019b

APPENDIX D SAMPLE INPUT FILES

ABAQUS Input File

```
Heading
** Specimen Number: T160416
** Model Type: RPC Rough
** Generated by: Matlab code Input_Write.
** 28-Oct-2019 18:27:29
**
*Preprint, echo=NO, model=NO, history=NO, contact=NO
**
** PARTS
**
*Part, name=Part-1
*Node, nset=Whole_Body_Nodes
    8606,   2.1800,   0.4800,   0.0000
    8607,   2.2000,   0.4800,   0.0000
    8618,   2.4200,   0.4800,   0.0000
    8619,   2.4400,   0.4800,   0.0000
…
 9925281,   3.6400,   4.7400,   2.0000
 9925282,   3.6600,   4.7400,   2.0000
 9925283,   3.6800,   4.7400,   2.0000
 9925284,   3.7000,   4.7400,   2.0000
*Element, type=C3D8, elset=Whole_Body_Elements
    8582,     8606,     8607,     8961,     8960,   107018,   107019,   107373,   107372
    8594,     8618,     8619,     8973,     8972,   107030,   107031,   107385,   107384
    8933,     8958,     8959,     9313,     9312,   107370,   107371,   107725,   107724
    8934,     8959,     8960,     9314,     9313,   107371,   107372,   107726,   107725
…
 9763808,  9826513,  9826514,  9826868,  9826867,  9924925,  9924926,  9925280,  9925279
 9763809,  9826514,  9826515,  9826869,  9826868,  9924926,  9924927,  9925281,  9925280
 9763810,  9826515,  9826516,  9826870,  9826869,  9924927,  9924928,  9925282,  9925281
 9763812,  9826517,  9826518,  9826872,  9826871,  9924929,  9924930,  9925284,  9925283
** Section: Section-1
*Solid Section, elset=Whole_Body_Elements, material=Bone
,
*End Part
**
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Part-1-1, part=Part-1
*End Instance
**
*Nset, nset=z_min_Nodes, instance=Part-1-1
      8606,       8607,       8618,       8619,       8958,       8959
      8960,       8961,       8962,       8963,       8964,       8965
…
    288707,     288708,     288709,     288710,     288711,     288712
    288713,     288714,     289060,     289061,
**
*Nset, nset=z_max_Nodes, instance=Part-1-1
   9658001,    9658002,    9658003,    9658332,    9658333,    9658334
   9658335,    9658336,    9658337,    9658338,    9658339,    9658340
…
   9925274,    9925275,    9925276,    9925277,    9925278,    9925279
   9925280,    9925281,    9925282,    9925283,    9925284,
**
*Nset, nset=RPC_z_min_Nodes, instance=Part-1-1
```

```
     32733,      32734,      32735,      33087,      33088,      33089
     33442,      33443,      33805,      33806,      33807,      34132

…
    264268,     264269,     264270,     264271,     264272,     264273
    264274,
**
*Nset, nset=RPC_z_max_Nodes, instance=Part-1-1
   9674279,    9674280,    9674631,    9674632,    9674633,    9674634
   9674635,    9674982,    9674983,    9674984,    9674985,    9674986
…
   9907930,    9907931,    9907932,    9907933,    9908281,    9908282
   9908283,    9908285,    9908286,    9908287,
**
*Nset, nset=Bottom_Centroid_Node, instance=Part-1-1
     48329,
**
*Nset, nset=Top_Centroid_Node, instance=Part-1-1
   9697309,
**
*End Assembly
**
** MATERIALS
**
*Material, name=Bone
*Elastic
 10000000000.0000, 0.300
** ----------------------------------------------------
**
** STEP: Step-1
**
*Step, name=Step-1, nlgeom=NO, solver=iterative
*Static
1., 1., 1e-05, 1.
**
** BOUNDARY CONDITIONS
**
** Top face displacement
*Boundary
RPC_z_max_Nodes, 1, 2
RPC_z_max_Nodes, 3, 3, -0.1
** Bottom face fixing
*Boundary
RPC_z_min_Nodes, 1, 3
**
**OUTPUT REQUESTS
**
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
*End Step
```

ADA Input File

```
#BSUB -J AbaqusJob
#BSUB -L /bin/bash
#BSUB -W 3:00
#BSUB -n 1
#BSUB -R "span[ptile=1]"
#BSUB -R "rusage[mem=35000]"
#BSUB -M 35000
#BSUB -o Abaqus.%J


## Load the modules
module load ABAQUS/2018

## Launch Abaqus with proper parameters
abaqus memory="35000MB" cpus=1 job=T160416_RR input=T160416_RR.inp
```

More information on input formatting and submission is available at

https://hprc.tamu.edu/wiki/Ada.