Politecnico di Milano

School of Design

Master of Science in Design and Engineering

DESIGN OF TRABECULAR STRUCTURE USING VOXEL-BASED MODEL

Supervisors:

Prof. Laura Vergani, Prof. Mario Guagliano

Co-Supervisors:

Ph.D. Mohammad J. Mirzaali, Ph.D. Flavia Libonati

Author:

Harsharoop Singanamalla

850718

Academic year 2017-18

Contents

Contents

Acknowledgements

Abstract

Micro-computed tomography (μCT) scan produces cross-sectional images of specific areas of a scanned object. Compression tests were induced experimentally on trabecular bone samples of porcine vertebrae and μCT scans were performed on them. These images are the input parameters and will be used for numerical analysis in this study. Mechanical and morphological properties were evaluated before and after the tests using μCT and DXA (dual-energy X-ray absorptiometry).

Designing a voxel-based model from the pre-damage μCT images using a series of image processing techniques using ImageJ and a MATLAB code that binarizes the greyscale images and eventually generates an input file code for ABAQUS.

Finite element modelling of bone samples was performed using the generated voxel-based model. This model was chosen because of its unique properties and ability to generate a mesh that is fully automated with the elements directly created from the μCT scans.

The primary objective of this thesis to see how close the designed model stands with respect to experimental results. Understanding how this finite element analysis can be used in future studies on a larger scale for predictions of diseases like osteoporosis was examined.

An analysis of calibration factors of Young's modulus was performed after defining plasticity model and implementing damage in the model. A full-fledged analysis considering its unique mechanical properties for various samples has been executed. A brief and interesting analysis considering mean values of all the required mechanical properties has been performed.

It has been proved that the concrete damage plasticity model, used in this study has been good enough in predicting the elastic and yield behaviour of the

samples. Though for few samples, the damage prediction was accurate, it was limited to hardened curves and lower strains.

Keywords: µCT images, voxel-based model, design, finite element modelling, concrete damage plasticity model, damage implementation

Sommario

La scansione della tomografia micro-computerizzata (μCT) produce immagini trasversali di aree specifiche di un oggetto scansionato. Test di compressione sono stati condotte sperimentalmente su campioni di osso trabecolare di vertebre porcine e su di essi sono state eseguite scansioni di μCT. Queste immagini sono i parametri di input e saranno utilizzate per l'analisi numerica in questo studio. Le proprietà meccaniche e morfologiche sono state valutate prima e dopo i test utilizzando μCT e DXA (assorbimetria a raggi X a doppia energia).

Progettare un modello basato su voxel dalle immagini μCT pre-danno utilizzando una serie di tecniche di elaborazione delle immagini usando ImageJ e un codice MATLAB che binarizza le immagini in scala di grigi e alla fine genera un codice di file di input per ABAQUS.

La modellazione ad elementi finiti di campioni ossei è stata eseguita utilizzando il modello basato su voxel generato. Questo modello è stato scelto per le sue proprietà uniche e la capacità di generare una mesh completamente automatizzata con gli elementi creati direttamente dalle scansioni μCT.

L'obiettivo principale di questa tesi è vedere quanto si avvicina il modello progettato rispetto ai risultati sperimentali. È stato esaminato il modo in cui questa analisi ad elementi finiti può essere utilizzata in studi futuri su scala più ampia per le previsioni di malattie come l'osteoporosi.

Un'analisi dei fattori di calibrazione del modulo di Young è stata eseguita dopo aver definito il modello di plasticità e implementato il danno nel modello. Un'analisi a tutti gli effetti, considerando le sue proprietà meccaniche uniche per vari campioni, è stata eseguita. È stata eseguita un'analisi breve e interessante considerando i valori medi di tutte le proprietà meccaniche richieste.

È stato dimostrato che il modello di plasticità di danno concreto, usato in questo studio è stato buono abbastanza accurato nel predire comportamento elastico e di snervamento dei campioni. Sebbene siano pochi campioni, la predizione del danno é stata accurata, limitata a curve indurite e deformazioni più basse.

Parole chiave: Immagini µCT, modello basato su voxel, design, modellazione ad elementi finiti, modello di plasticità del danno concreto, implementazione del danno.

List of symbols

E	-	Young's modulus
G1%	-	Strain group 1
G2%	-	Strain group 2
G3.5%	-	Strain group 3
G5%	-	Strain group 4
ε_{in}	-	Inelastic strain
ε_{pl}	-	Plastic strain
ε_{t}	-	Total strain
ε	-	Strain
σ_{y}	-	Yield stress
σ_{c}^{0}	-	Uniaxial compressive yield stress
p_{t}	-	Yield strength in hydrostatic tension
p_{c}	-	Initial yield stress in hydrostatic compression
d	-	Damage
C_{f}	-	Calibration factor
k	-	Compression yield stress ratio
k_{t}	-	Hydrostatic yield stress ratio
w_{c}	-	Stiffness recovery variable

List of abbreviations

BV/TV	-	Bone volume fraction
CDP	-	Concrete Damage Plasticity
CF	-	Crushable Foam
CP	-	Classical Plasticity
DXA	-	Dual-energy X-ray absorptiometry
μCT	-	Micro Computed Tomography
MRI	-	Magnetic Resonance Imaging
BMD	-	Bone Mineral Density
TBS	-	Trabecular Bone Score
.bmp	-	Bitmap Image file
.inp	-	ABAQUS Input file
.mhd	-	Meta-Image Header
.raw	-	Raw Image Format
RP	-	Reference Point
FEA	-	Finite Element Analysis
3D	-	Three Dimensional
μm	-	Micrometres
MPa	-	Mega Pascal
HR-pQCT	-	High-resolution peripheral quantitative computed tomography

Chapter 1

Introduction

In this chapter, a brief insight into the anatomy of bone, in specific, lumbar vertebrae explaining the choice of porcine bone over human bone. Later, the input parameters for this study are described, primarily the µCT images obtained from experimental tests, which is the basis of this whole work.

1.1 Lumbar vertebrae

In this section, a brief illustration of lumbar vertebrae, which is the subject of study in this thesis will be described. It's preceded by the anatomy of bone which is essential to understand the succeeding section explaining the reasons for choosing porcine vertebrae over human vertebrae.

1.1.1 Anatomy of bone

Bone is a rigid body tissue consisting of cells embedded in an abundant, hard intercellular material. It provides structural support for the mechanical action of soft tissues, protects vital organs and tissues, produces blood cells and acts as a mineral reservoir. We are born with about 300 soft bones. During childhood and adolescence, the cartilage grows and is slowly replaced by hard bone. Some of these bones later fuse together, so that the adult skeleton has 213 bones [20] while the number may decrease in older people due to the union of bones which are normally independent.

Bones are composed of two types of tissues, a hard outer layer called cortical (compact) bone, which is strong, dense and tough and a spongy inner layer called trabecular (cancellous) bone, which is analysed in this study. Within

the trabecular portions of bones, lies a spongy tissue called Bone marrow, which is the producer of red and white blood cells and platelets.

Bones are categorized into long, flat and short bones based on their general conformation. While vertebrae are examples of short bones, frontal bone in the skull, the thoracic cage (sternum and ribs); and femur, tibia and phalanges are examples of flat and long bones respectively.

Cortical bone

Trabecular bone

Bone marrow
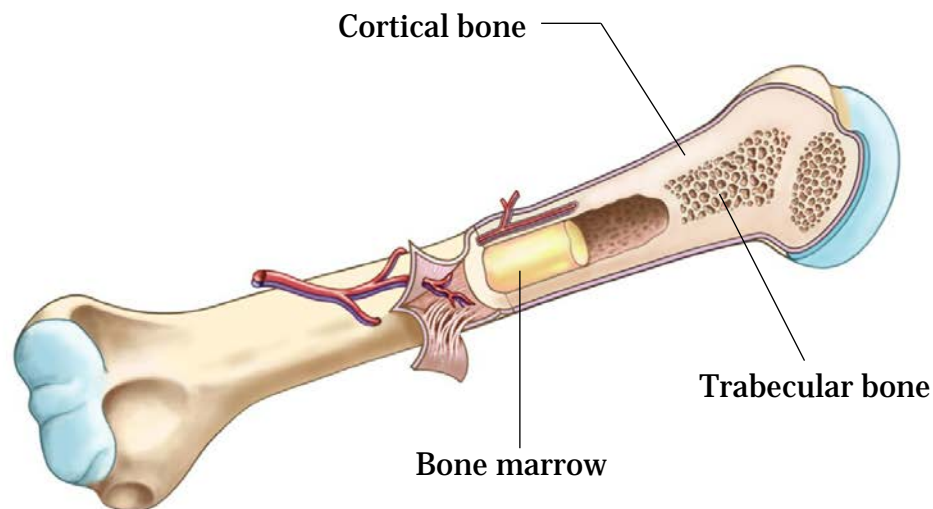
Figure 1.1 – Anatomy of a long bone

1.1.2 Structure of Vertebral Column

The vertebral column, also called spine or backbone, consists of 33 bones [21], the vertebrae, each of which is separated and united by an intervertebral disc. Together, the vertebrae and intervertebral discs form the vertebral column. It forms the central axis of the body, carries the weight of the body above the pelvis and protects the spinal cord.

Vertebrae are short bones, mainly made by trabecular bone tissue covered by a thin layer of compact bone tissue. Lumbar vertebrae are the largest of the five different types of vertebrae. They act to support the weight of the upper body and have various specialisations to enable them to do this. Lumbar vertebrae have very large vertebral bodies, which are kidney-shaped, as shown in

Figure 1.2 (b). They are five in number, while the others, viz. Cervical (7), Thoracic (12), Sacrum (5) and Coccyx (4) constitute rest of the 28 vertebrae [21].

(a)

(b)
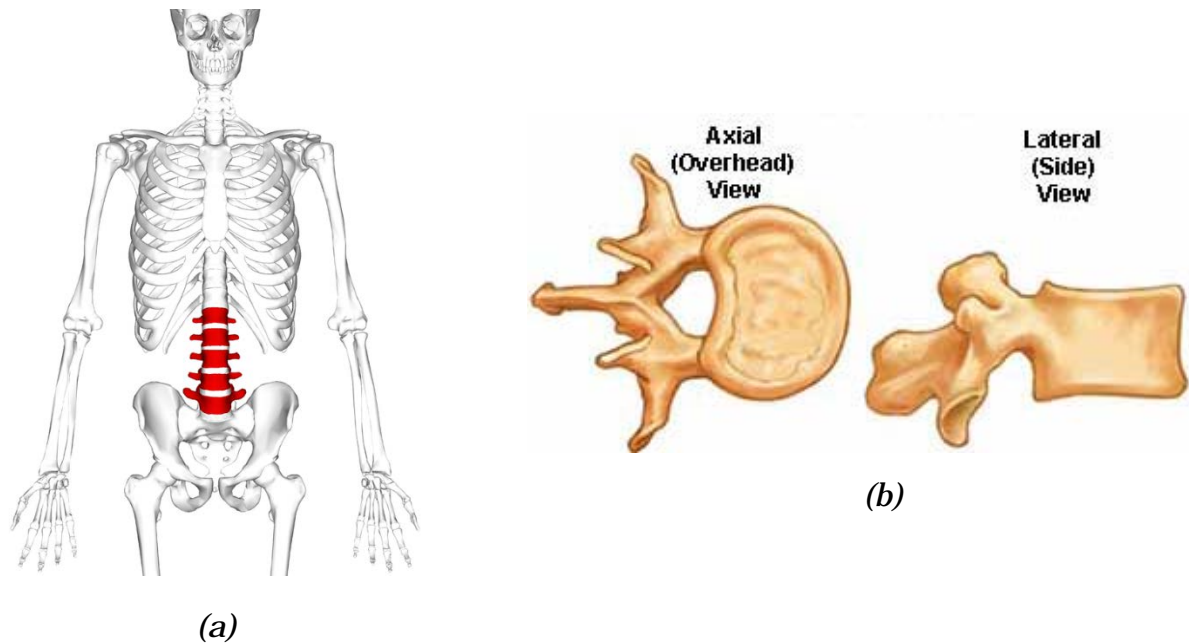
Figure 1.2 – Position (in red) (a) and shape (b) of Lumbar vertebrae

1.1.3   Porcine spine and its resemblance to the human spine

New spinal implants and surgical procedures are often tested pre-clinically on human cadaver spines. However, the availability of fresh frozen human cadaver material is very limited and alternative animal spines are more easily available in all desired age groups and have more uniform geometrical and biomechanical properties. According to [2], In Italy, though legislation is clear on the importance of scientific research in health protection, the rules are too sparse and obsolete, dating back to the mid-900 and donors are not completely aware of donating their bodies after death. Though this is almost irrelevant for this study, it is important to know the underlying motive behind having to choose a porcine bone.

The anatomical dimensions of the cervical, thoracic, and lumbar vertebrae of the human and porcine spine has been analysed in [1]. The porcine spines have 7 cervical, 15 thoracics, and 6 lumbar vertebrae making it a total of 28, in comparison to the human spines which had 7, 12, and 5 respectively, whose aggregate number is 24. Subtracting the four extra vertebrae from the porcine spine makes it shorter. However, the mean total spine length between the human and porcine spines had barely any difference. Comprehending the aforementioned description, it can be said that porcine spine is the most representative model for the human spine, indeed making it the best choice.

1.2   Input parameters for design of bone sample

In this section, the input data on which the design and analysis in this study has been done will be introduced. The chronological order of tests carried out to obtain the input data for this study is as described below. Firstly, six different vertebral columns provided from a local butcher were scanned with a dual-energy X-ray absorptiometry (DXA) measuring the bone mineral density (BMD) and to check the quality of the full vertebra. Forty porcine trabecular specimens were extracted from the lumbar vertebrae using necessary machining techniques. This was followed by a micro-computed tomography (µCT) scan, which is explained in the later part of this chapter, in order to compute the most relevant architectural parameters. The images obtained by this scan are the input data for this study. Then monotonic compression tests were carried out to induce a mechanical damage in the samples. After the tests, the µCT scan was performed again resulting in another set of images. The aim of this thesis is to see how accurate the results can be achieved numerically considering the pre-damage µCT images.
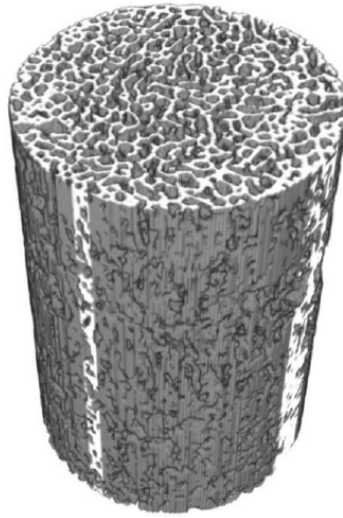
Figure 1.3 – 3D view of the cylindrical bone sample

The compression tests were performed on cylinder-shaped samples (Figure 1.3) with diameter 13.85 mm and height 30 mm approximately [19]. Firstly, a full vertebra DXA scanning was performed followed by embedding the sample in endcaps on both sides to provide a rigid surface for the test. A pre-damage and post-damage DXA scanning and µCT imaging have been done before and after the test respectively. The test was performed placing the sample in a saline solution (NaCl 0.9%) to ensure the real ambience.

Forty specimens were randomly divided into four groups of 10 samples, each group loaded with a different strain value. The specimens that fall into Group 1% (G1%) were loaded to a strain of 1%, Group 2% (G2%) were loaded to a strain of 2%, Group 3.5% (G3.5%) were loaded to a strain of 3.5%, Group 5% (G5%) were loaded to a strain of 5%.

Though there are many material properties obtained from the experimental data, there are only a few parameters predominantly used in this study. They are Bone Volume fraction (BV/TV), which is the ratio of Bone volume and the total volume; the amount of bone mineral in bone tissue, also known as Bone Mineral Density (BMD); Trabecular Bone Score (TBS), which is a measure

of bone texture correlated with bone microstructure and a marker for the risk of osteoporosis, a bone disease with an increase in susceptibility to fracture.

1.2.1 μCT Images

A CT scan, also known as computed tomography scan, makes use of computer-processed combinations of many X-ray measurements taken from different angles to produce cross-sectional images of specific areas of a scanned object, allowing the user to see inside the object without cutting. Computed tomography operates by using an X-ray generator that rotates around the object; X-ray detectors are positioned on the opposite side of the circle from the X-ray source. Once the scan data has been acquired, the data must be processed using a form of tomographic reconstruction, which produces a series of cross-sectional images. These are called CT images. When the pixel sizes of the cross-sectional images are in the micrometre range, it is termed as a μCT image. The smallest pixel size in an image is called resolution.

Trabecular bones, owing to their variable internal microstructure, are unpredictable in their mechanical properties. This is the major reason behind having to choose μCT scanning techniques to get a glimpse of its microstructure. In this study, the pre-damage μCT images obtained before the damage test, as mentioned above, are the input parameters. These images, having a resolution of 26.3 μm, appear as shown below.
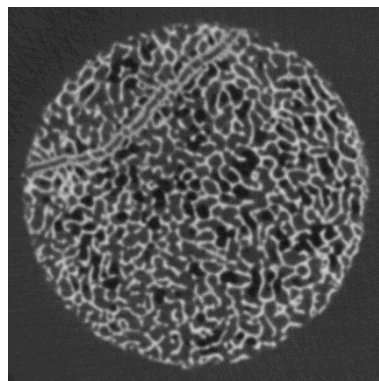

Figure 1.4 – μCT image of the cylindrical sample

1.3 Methodology and objectives

The μCT images are further subjected to a series of **image processing** techniques which will eventually lead to the generation of a computational 3D model or a finite element (FE) model. A computational model of human anatomy is a mathematical representation of the external envelope of the human body shape together with the boundaries of the internal organs and tissues. In principle, there are two basic concepts for generating FE models from CT scans, which are `geometry-based' and `voxel-based' [4]. The model used in this study is a voxel-based model. The geometry-based meshing is more common and requires the extraction of inner and outer contours from CT scans of bones. The voxel-based meshing usually implies that the element faces are oriented parallel to the three orthogonal axes defined by the coordinate system of the CT scanner. This structure is achieved by arranging the nodes in the form of a cubic lattice with this orientation and converting a voxel or several adjacent voxels directly into an eight-node iso-parametric brick element [4].

The pixel data from μCT images, when extended into the third dimension become cuboidal volume elements called voxels. This data may be used to create a three-dimensional digital representation of the shape, volume and composition of the trabecular bone structure. Voxel models are the most faithful representation of human anatomy currently available as they are designed from the real anatomy of an individual human, in this case, a porcine. The generation of this voxel model was possible, thanks to a MATLAB code, which was adapted and improved to match the demands of this study, which will all be elaborated in the next chapter.

The generated finite element model was imported to ABAQUS for analysis. The input parameters were adapted from the experimental data and the samples were loaded to displacements corresponding to different strain values. Young's modulus needed to be calibrated for the results to be matched with the experimental data. After the results obtained were in good agreement with the

experimental ones, the next challenging task was to define plasticity. Three plasticity models were chosen based on literature and the analyses were run with all the three of them. Then the results were compared with the experimentally obtained stress-strain data to choose the best suitable material model for this study.

Compression damage was then implemented in the model in the chosen plasticity model itself. Though it doesn't provide accurate crack data like crack width, it approximates cracks by the help of necessary damage indices, however, it is not the focus of this study. Resembling the experiments, the final step performed was unloading the model. To do this, a small strain in terms of displacement was applied to the sample only to extrapolate the obtained curve later. The main objective of this study is to verify how close numerical analysis using a voxel-based model resembles the experiments and how changes in material properties affect the whole analysis.

Furthermore, the differences in calibration of E, pre-damage and post-damage will be explained in the later parts of the study. Eight samples were primarily chosen for the analysis from the forty specimens. Since each sample has unique mechanical properties, all the samples were analysed with a single set of average calibrated properties to see how they correlate with each other, as every property pertains to trabecular bone.

Chapter 2

Designing voxel-based model from µCT images

In this chapter, the design of a voxel-based 3D model from µCT images using a series of image processing techniques will be conferred. The processing that images undergo before and during the execution of a MATLAB code is described and is followed by its detailed algorithmic illustration. An input file for ABAQUS is generated using the code with a desired number of elements by optimising specific parameters, not to transcend the limitations of a feasible computation. In the later part, automatizing the material properties of the bone in the input file, which indeed reduces manual effort will be examined.

2.1 Image processing

The CT scan of each sample produced at least 700 slices of images with a minimum size of 500x500 pixels each spreading throughout the circular cross sections of the 30mm tall cylindrical sample of trabecular bone including the cap fixtures on both sides, which approximately accounts to 37.5 µm thickness each. The resolution of these images was 26.3 µm. These images were in '*.mhd' format, an abbreviation for Meta-Image Header, which is a text-based tagged file format for medical images. But the required input for the MATLAB code was supposed to be in .bmp (*Bitmap*) file format. To achieve this and few other image modifications, we found an excellent pick in an image processing software called ImageJ.

2.1.1 ImageJ

ImageJ is an open source image processing program designed for scientific multidimensional images. ImageJ was designed with an open architecture that provides extensibility via Java plugins and recordable macros. The most beneficial feature of ImageJ was that it could display, edit, analyse, process, save and print 8-bit colour and grayscale, 16-bit integer, and 32-bit floating point images. It can read many image file formats, including TIFF, PNG, GIF, JPEG and BMP, as well as raw formats. It also supports image stacks, a series of images, which can be edited at once, in turn, not disturbing the extraction of the 3D model. This was the reason behind choosing ImageJ over other image editing softwares like Adobe Photoshop etc. There is also an external plugin exclusively for bone image analysis called BoneJ, which provides free, open source tools for trabecular geometry and whole bone shape analysis.

The first step was to delete the images that do not belong to the circular cross-section of the sample. This is because the CT scan also includes the cap fixtures on either side, which is the light grey part in Figure 2.1. To do this, a square region was specified and then cropped. Now, all the images in the sequence will be cropped. This image stack was saved in the format '.bmp', considering the fact that the MATLAB code which converts these images into a voxel-based model demanded this type of file format. A '.inp' file was generated from the MATLAB code which will be elaborated in the later part of this chapter.
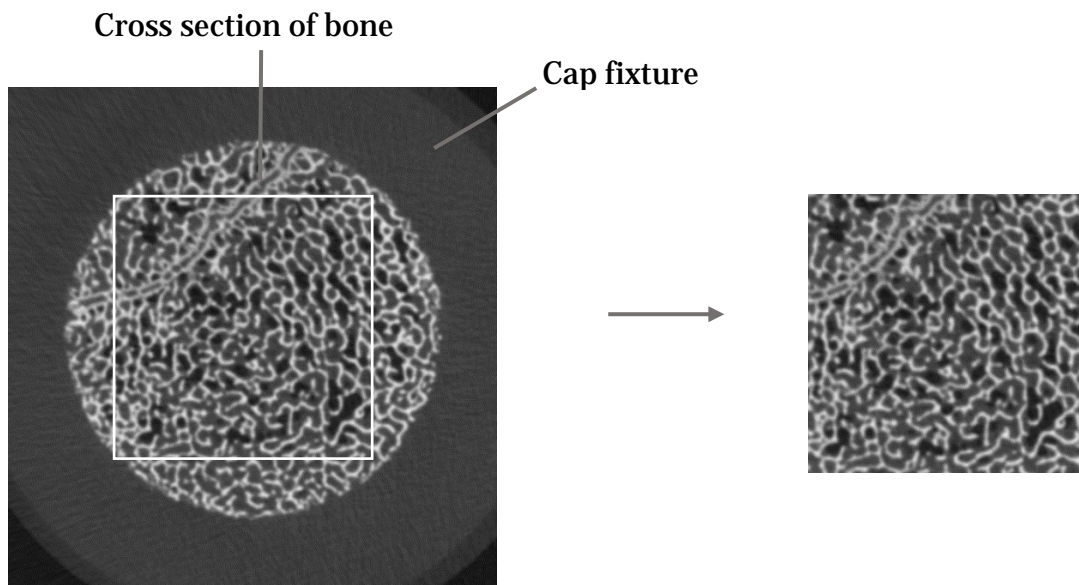
Cross section of bone

Cap fixture

Figure 2.1 – Cropped cross-section of the trabecular bone sample

2.2 Finite element modelling

After obtaining a series of images through image processing and segmentation, the next step is to generate a finite element mesh which aptly suits this study. So a literature review was performed to have an insight into the presently available finite element modelling techniques based on CT images.

Recent improvements in 3D imaging technology, among them the development of micro-CT and micro-MRI (Magnetic Resonance Imaging) scanners, allow in-vivo 3D high-resolution scanning and reconstruction of large specimens or even whole bone models. Thus, the tendency today is to evaluate bone features using 3D assessment techniques rather than traditional 2D methods. For this purpose, high-quality meshing methods are required. However, the 3D meshes produced from current commercial systems usually are of low quality with respect to analysis and rapid prototyping. Besides voxel-based model, there are quite a few finite element modelling techniques based on images like geometry-based modelling, statistical models like spring network model, neural network model.

As compared in [4], geometry-based meshing technique is not fully automated and demands the determination of geometry between CT scanning and mesh-generation. In contrary, voxel-based mesh generation technique is fully automated with the elements directly created from the µCT scans. Though there are a variety of advantages of using a voxel-based modelling, here are a few: same complexity for all objects, Heterogeneity and anisotropy can be easily incorporated into the analysis, it is simple, intuitive and unambiguous [15]. As with any other technique, there are a few limitations to using a voxel-based meshing technique. Large memory requirements, larger processing times, inaccurate contacts of voxels compared to the trabecular connections are a few of them.

Both hexahedral and tetrahedral voxel-based meshing techniques were analysed and discussed in [8] concluding that the preferable meshing technique depends on the trabecular morphology, in particular, its trabecular thickness. In situations where a plain hexahedron meshing method technique results in a substantial loss of connected bone tissues, the tetrahedron method was suggested to be used. But, for computational feasibility and the availability of MATLAB code for generating hexahedron elements, the same was chosen in this study.

2.3 Design of voxel-based model

The schematic representation of the process underlying the conversion of a trabecular bone sample to the generation of a voxel-based model is shown in Figure 2.2. Firstly, the bone sample undergoes a CT scan and the µCT greyscale images are obtained [Figure 2.2(b)]. Then the conversion of greyscale images to the final finite element model is done using a MATLAB code which is illustrated in the later part of this section.
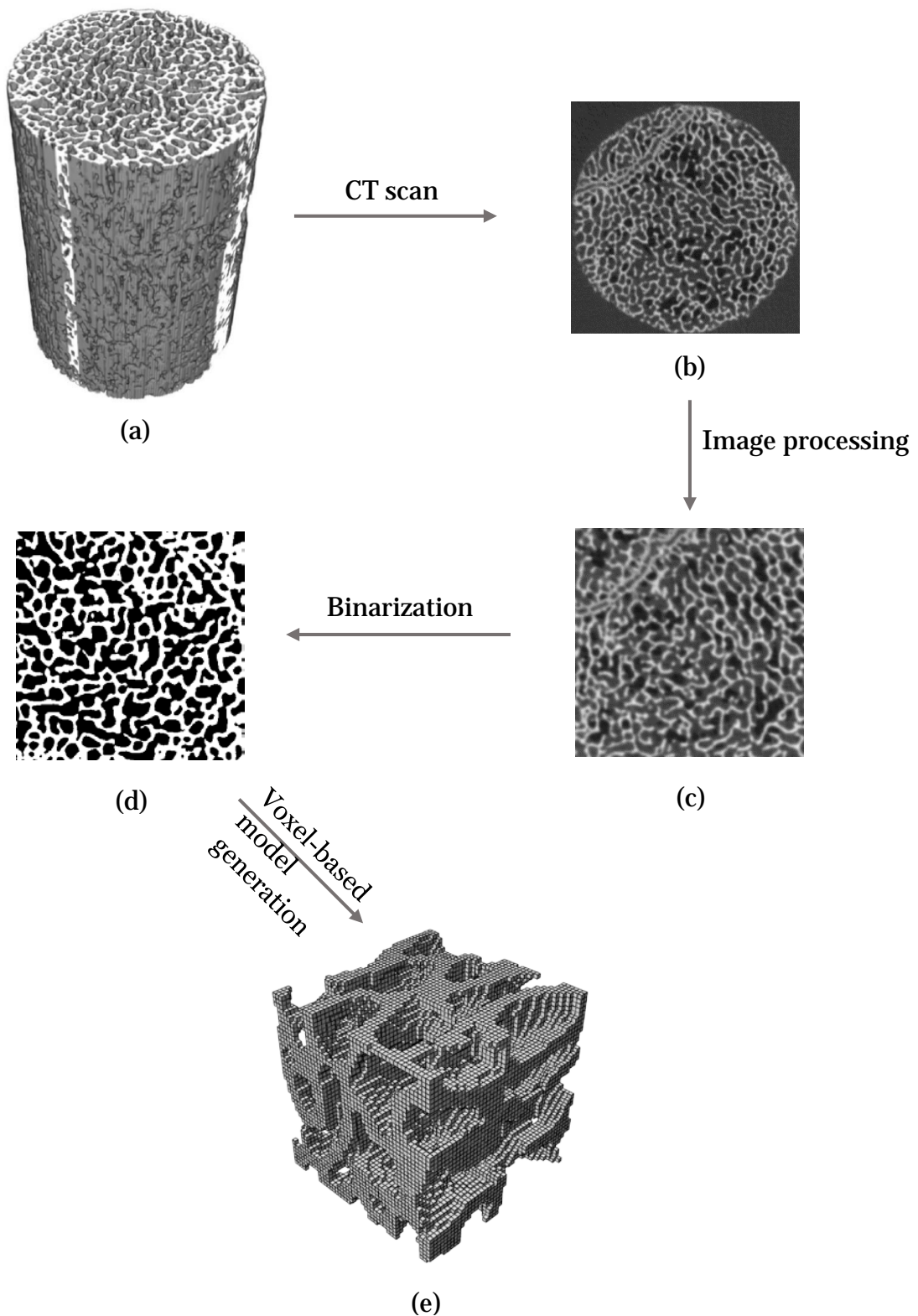
(a)

CT scan

(b)

Image processing

(c)

Binarization

(d)

Voxel-based model generation

(e)

Figure 2.2 – Flowchart showing the conversion of bone sample to generation of voxel-based model

2.3.1 Input parameters for MATLAB code

Before starting on with the algorithm, discussing the input parameters for this code, the μCT image stack obtained after cropping from ImageJ is the primary input. Secondly, the code requires the input of few other parameters namely the cube side, delta x, delta y, delta z and compression factor. The cube side to be inserted is equal to the product of a number of voxels required on each side and the compression factor. Delta x, delta y and delta z are taken as 1. As defined in the MATLAB code, compression factor accepts three values, viz., 2, 3 and 6.

If the compression factor is 2, it means that the number of voxels reduces to half on each side of the cube. So, if one inputs 100 as cube side and 2 as the compression factor, the resulting voxel-based model will have a volume of 50x50x50. As the compression factor increases, the trabecular network gets less accurate, thereby demanding a need to maintain a balance between them. When a model of 90x90x90 was imported into ABAQUS with different compression factors 2, 3 and 6, the results obtained are shown in Figure 2.3.

(a)   (b)
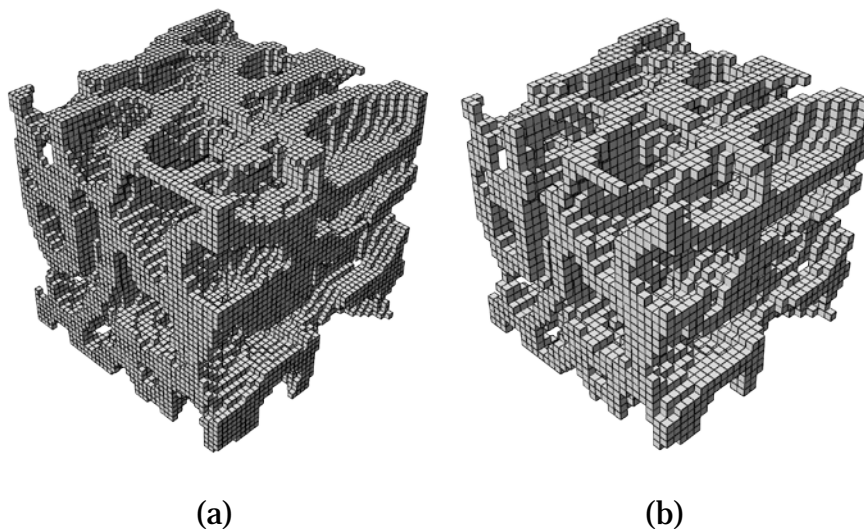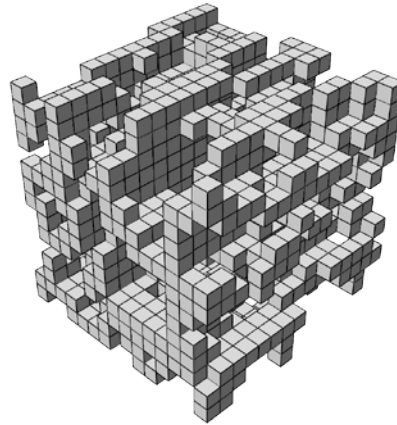
(c)

Figure 2.3 – Finite element models of a representative cube of length 2.36 mm and voxel resolution of 52.6 μm (a), 78.9 μm (b) and 157.8 μm (c)

All the models in Figure 2.3 have an equal side length of 2.36 mm but considering different compression factors resulted in three different resolutions. As it's clearly evident, the model with a compression factor 2 has a very accurate trabecular network when compared to the rest. When one is ready to compromise on the accuracy rather than on the length of the cube side, the model with a compression factor of 6 could be chosen. But the choice made here was to obtain a model with higher accuracy compromising on the length, thereby choosing the model with a compression factor of 2.

2.3.2   Choosing the ideal model for computation

One of the major challenges was to choose a finite element model with a reasonable number of elements for computation. Since computational feasibility was essential to be optimal, various side lengths of the cube were considered and the numbers of elements were noted simultaneously. The length of the cube was gradually decreased till an optimal number of elements were reached. The process was started by choosing an arbitrary representative cube of volume 250x250x250 and the number of eight-node brick elements was found to be 3,322,560, which is too huge for computation. Then the size of the cube was

gradually reduced to 200, 100, 50, 40 and 35 and the numbers of elements were noted simultaneously as shown in Table 2.1. Finally, the cube of size 40x40x40 was chosen for analysis in this study, whose length of each side is 2.1 mm. All the values are considered for the sample S1_G3.5% (Refer Appendix B for properties). A compression factor of 2 was used in all the cases since 3 and 6 were ruled out as illustrated in the previous section. Few other samples with different mesh sizes and the number of elements are listed in Appendix C.

S.No	Length (mm)	Mesh Size	Number of elements	Compression factor
1	13.15	250x250x250	3,322,560	2
2	10.52	200x200x200	1,804,314	2
3	5.26	100x100x100	365,600	2
4	2.63	50x50x50	49,854	2
5	2.10	40x40x40	26,089	2
6	1.84	35x35x35	17,320	2

Table 2.1 - Variation of number of elements with length and mesh size of the finite element model

2.3.3 MATLAB Algorithm

The images when imported to MATLAB undergo the following sequence of steps by using an improvised pre-existing code. The phenomenon involved in the conversion of μCT images to the generation of the input file are explained in this section.

Firstly, the intensities of the μCT images, being greyscale images, are recorded in an array. The size of this 3D array would be equal to the size of the image in a 2D matrix with the number of images being the third dimension. For a greyscale image, the pixel value is a single number that represents the brightness of the pixel. The most common pixel format is the byte image, where this number is stored as an 8-bit integer giving a range of possible values from 0

to 255. Typically zero is taken to be black, and 255 is taken to be white. Values in between make up the different shades of grey.

The next step is to binarize the greyscale images. This is done by using a process called binarization. It is the process of converting a pixel image to a binary image. Initially, an image is converted to grayscale, which is invalid in this case. Later, a threshold is applied to the image. Finally, a better adaptive threshold is calculated by an algorithm from the histogram (Figure 2.4) of 'pixels vs grey level value'. All the pixels that have intensities below this threshold grey scale value will be assigned '0' and the rest, '1' and thus a binary image is generated.
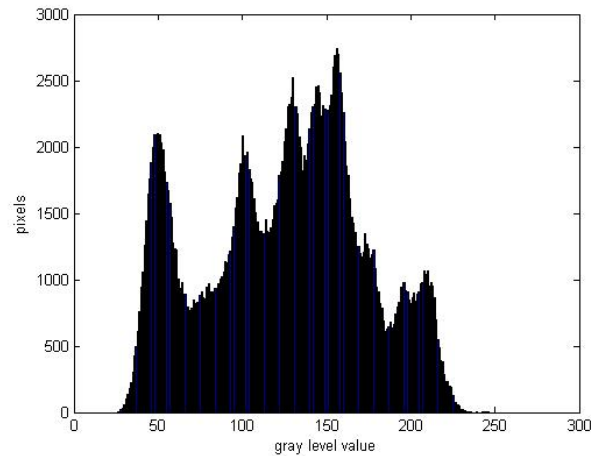

Figure 2.4 - Binarization histogram

There are a few techniques to find the optimal threshold, of which, one of the simplest is by trial and error method. But considering a bimodal image, which is an image whose histogram has two peaks, we can approximately take a value in the middle of those peaks as the threshold value. That is what Otsu binarization does. It automatically calculates a threshold value from image histogram for a bimodal image. MATLAB uses Otsu's method to achieve binarization. It computes the global threshold by the command GRAYTHRESH that can be used to convert an intensity image to a binary image with IMBINARIZE.

The binary images are now stored as a single bit, i.e., a 0 or 1. Now, an input file (*.inp) is created which consists of all the nodes and the elements of the final mesh. An input file consists of the coordinates of all the nodes, the elements formed using these nodes, the material properties assigned to them and the boundary conditions. A typical input file, when imported into ABAQUS, looks like the one shown in Figure 2.3 (a).

2.4   Bone tissue parameters

The elastic parameters were used in the isotropic form in this study and chosen from the experimental results. The Poisson's ratio was taken as 0.3, as considered in [7]. The Young's modulus obtained from experimental results is the value of the whole sample, the calibration of Young's modulus, which is required to calculate its value for a single voxel will be discussed in the later parts of the study. The input file obtained from the MATLAB code was integrated with the parameters required to run the analysis automatically. Simply put, the code was automatized such that the only task left to do was to create a job file in ABAQUS before running it.

2.4.1   Modelling of in-situ compression test

It is very evident from Figure 2.3 that a higher compression factor leads to less real bone representative model, although being a model of higher side length. As mentioned earlier, the voxel resolution of this model, considering a compression factor of 2 is 52.6 μm. The loading is applied by uniformly distributed displacements on the loaded area with a maximum displacement equal to 2% of strain, which is 0.04208 mm. This displacement was applied on Z-direction to replicate the experiments. Since the cross-section of the images is circular, which indicates the X and Y axes, the other perpendicular direction is the loading direction.

Two reference points were created namely RP-1 and RP-2 in the centre of the top and bottom surfaces perpendicular to the direction of the loading. A kinematic coupling was defined considering RP-1 and RP-2 as master nodes and the rest of the nodes in their respective surfaces as slave nodes. In Figure 2.5, RP-2 on the top of the model is connected to all the other nodes of the surface, and so is the case with RP-1. Now the displacement was given at RP-2 and RP-1 was fixed in all the directions. A static/general step was employed with non-linear geometry turned on and a full-Newton solution technique was applied. The total time was 1 and the increments were set to 1000000 with minimum increment size of 1E-015. The instance where an additional step had to be defined, for the unloading part of the curve, will be explained in Chapter 3.
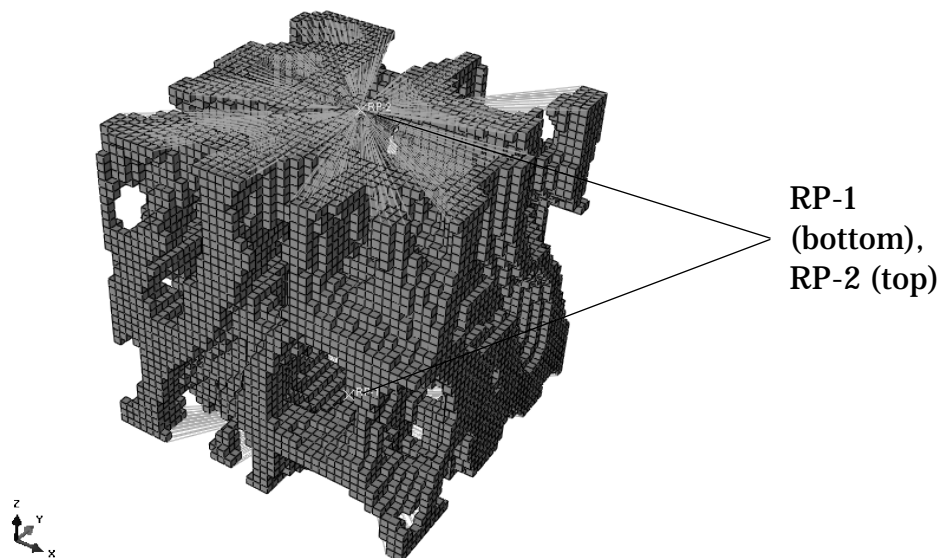

RP-1 (bottom), RP-2 (top)

Figure 2.5 – Kinematic coupling

2.4.2 Automatizing MATLAB code

After the generation of nodes and elements in the code, a section was created with a pre-defined material name, followed by the definition of assembly. Two reference points were created at the centres of both top and bottom XY-planes (perpendicular to Z-axis). Following is the definition of the bottom and

top surfaces required for the definition of boundary conditions. Now, kinematic coupling is defined followed by the material properties. Finally, the definition of step will be done succeeded by the output requests. All these parameters are defined in a hierarchical order for ABAQUS to understand and run the analysis immediately.

This step is essential in reducing the manual effort to define the material properties of every sample that has been analysed in this study. Though the material properties differ from each other, considering the fact that couplings, assembly features remain the same, it was very useful. Refer [Appendix −A] for the detailed MATLAB code.

Chapter 3

Calibrating material properties, defining plasticity and implementing damage

Having discussed the modelling of the compression test in the previous chapter, in this chapter, the properties that need to be calibrated, defining plasticity and implementing damage will be discussed. Choosing the most feasible plasticity model out of three models will be followed by the calculations involved in implementing damage in the model will be scrutinized.

3.1 Calibrating material properties

Trabecular bone, being a porous material has different global and local properties. Calibrating the mechanical properties of the samples were performed in this study to achieve the right results. Though all the samples should ideally possess similar properties like Young's moduli and yield stresses, there was significant variation within them. For instance, Young's moduli of the samples varied from 815 MPa to 2250 MPa. For more accurate results, calibration was done individually for all the samples tested in this study. In the next chapter, the stress-strain behaviour of the bones considering average calibrated properties will also be looked at. But in this section, the properties that were calibrated, the procedures and results will be described.

Young's modulus and yield stress are calibrated in this study leaving aside the calculation of effective area. As the primary objective of this study is to validate the experimental results and since the nominal area was considered during the calculation of experimental stress-strain data, the same approach is applied here.

3.1.1 Stiffness Calibration

The experimentally obtained Young's moduli for all the samples were calculated using the stress-strain curves, which is the global value. The input parameters in ABAQUS are supposed to be local values or that of each voxel. To obtain these, it is required to calibrate them with reference to global values and thereby the following procedure was followed. Firstly, a random local value of Young's modulus (1000 MPa) was chosen as input considering the sample S2_G3.5% (Refer Appendix B for properties) in ABAQUS. The analysis was run and the stress-strain curve was obtained. The Young's modulus obtained from this curve was found to be 247 MPa, which is the global value. The question now is what value of Young's modulus is required as an input to obtain the experimental value (2108 MPa) from the stress-strain curve, which implies the following equation.

$$\text{Calibrated E} = \frac{\text{Experimental E}}{\text{Numerical E obtained}} \text{ x local E}$$

$$= \frac{2108}{247} \text{ x } 1000 = 8507 \text{ MPa}$$

Implementing the above result as input E, the resulting graph obtained when the sample has been loaded to 0.5 % strain is shown in Figure 3.1. The overlap clearly explains that the Numerical analysis post calibration of E matches with experimental E.
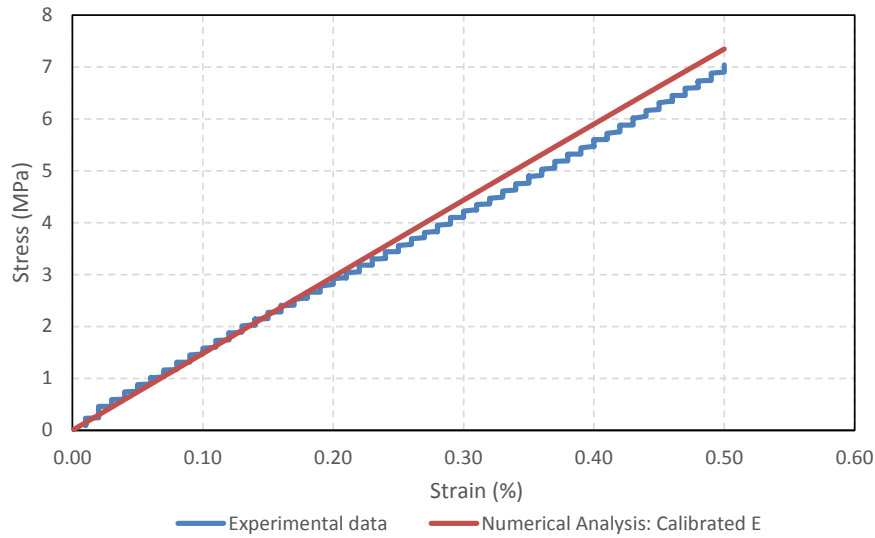
Figure 3.1 – Stress-strain curve of S2_G3.5% after calibration of Young's modulus

In the above scenario, the experimental E and Calibrated E are 2108 MPa and 8507 MPa respectively. The ratio of Calibrated E to the experimental E has been named as Calibration factor (C_f). A detailed analysis of C_f pre-damage and post-damage considering 5 different samples from a single strain group will be illustrated in the next chapter. The factor of calibration of E from the above example (Figure 3.1) is the pre-damage C_f, which is obtained in the elastic region.

$$\text{For S2\_G3.5\%, } C_f = \frac{8507}{2108} = 4.03$$

The factor of calibration of E obtained after the damage is called post-damage C_f. This parameter will be calculated after defining plasticity and implementing damage in the model.

3.1.2 Calibrating Yield stress

The calibration of Yield stress (σ_y) was performed using trial and error method. As a benchmark, the same factor used for stiffness calibration was used. Then depending on the resulting stress-strain curve, the calibration factor

was either increased or decreased. In table 3.1, the C_f values of E and σ_y of six samples are tabulated. We can see that the maximum difference between the calibration factors of Young's moduli and yield stresses is 1.2 in the case of S3_G5% (Refer Appendix B for properties). Implicationally, there was only one or two trial and error operations to be made for each sample during analysis before achieving the final results. Firstly, if the calibration factor of E is 4.3 (for S3_G3.5%), σ_y was calibrated with a factor of 4.5 and verified if the results are satisfactory before proceeding to 5.

In Figure 3.2, the results of various mentioned samples when calibrated E and σ_y are used can be observed. Though the results obtained in this figure were achieved considering concrete damaged plasticity model, it will be elaborated in the next section.

Sample	Calibration factor	
	E	σ_y
S1_G3.5%	4.03	4.40
S2_G3.5%	5.10	6.20
S3_G3.5%	4.30	5.00
S1_G5%	2.45	3.10
S2_G5%	2.30	2.60
S3_G5%	4.70	5.90

Table 3.1 - Comparison of calibration factors of E and σ_y
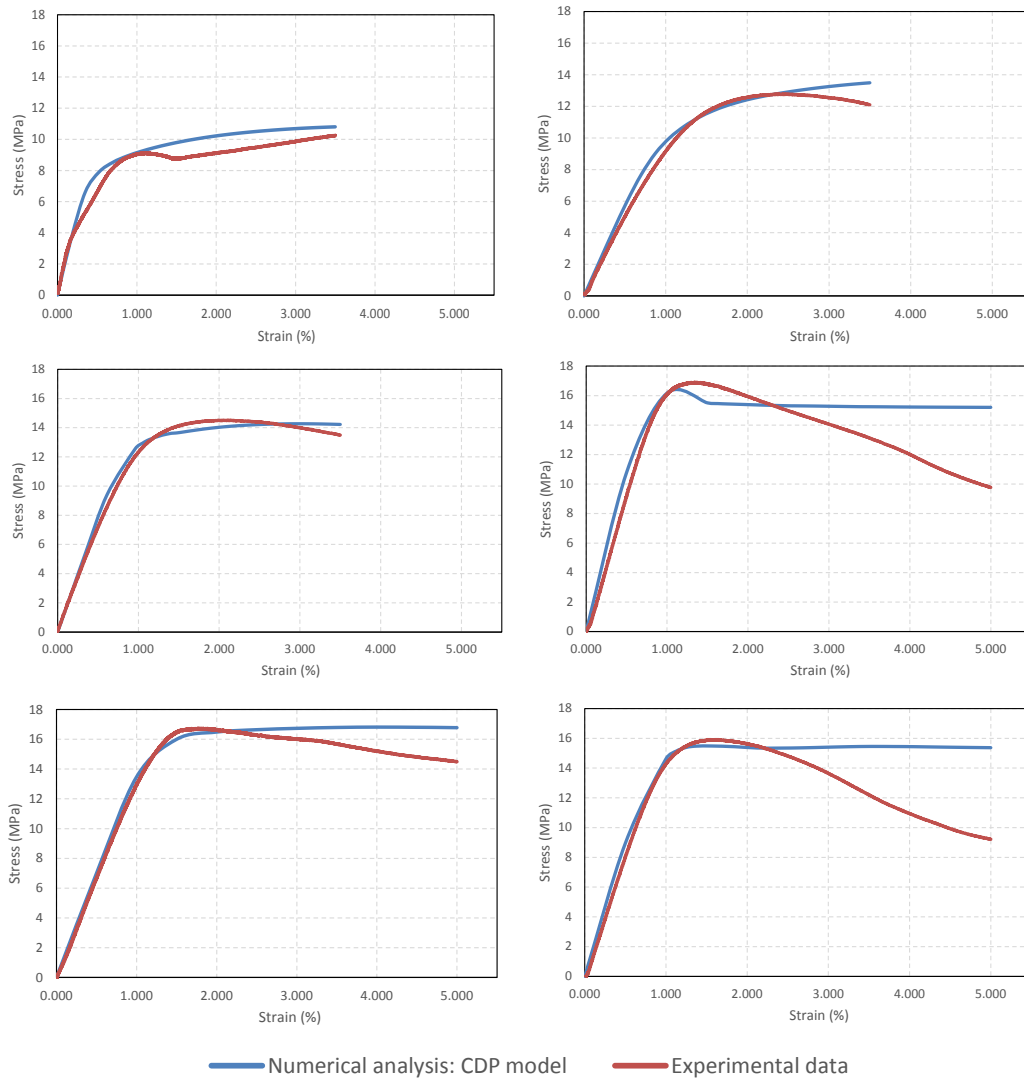
Figure 3.2 – Stress-strain behaviour of S1_G3.5%, S2_G3.5%, S3_G3.5%, S1_G5%, S2_G5% and S3_G5% (left to right) respectively, post-calibration of E and σy

3.2   Choosing plasticity model

After defining the elasticity, the next step was to define plasticity model which is the right fit for this study. To do this, a literature review was carried out to find the models used in numerical analyses of bones. Two models namely Concrete damage plasticity and Crushable foam models were found in [10] and [11]. Concrete damage plasticity model provides general capability for modelling

quasi-brittle materials in all types of structures and crushable foam model can be used to simulate the ability of a foam-like material to deform in compression tests. One more material model that was found promising was classical plasticity model. In the latter part of the study, the three material models were compared with the experimental results to choose the best fit.

Two samples from G3.5% were analysed with all the three plasticity models. Samples from G3.5% were chosen to given enough chance to decide Brief explanation of all the three plasticity models and the stress-strain curves will be followed by a comparison to find the best match. Assuming they are S1_G3.5% and S2_G3.5%, the BV/TV values being 0.416 and 0.419 respectively. In [Appendix −B], a column showing the samples considered and their respective properties are tabulated.

Each plasticity model demanded a variety of input parameters, which were adapted from literature and will be described in the following sections. In common, the required post-yield compression data was taken from the respective experimental data by plotting a curve between Strain (%) and Stress (MPa) starting from the yield point as shown in Figure 3.2. A polynomial equation is obtained from the curve for each sample and the values of stresses are calculated by substituting the values of strains at regular intervals. Finally, plastic strains were calculated at all these points and substituted in ABAQUS along with the values of stresses. This process of obtaining the post-yield compression data was a necessary step as the experimental data had more than one stress value for a single strain value, which is legitimate.
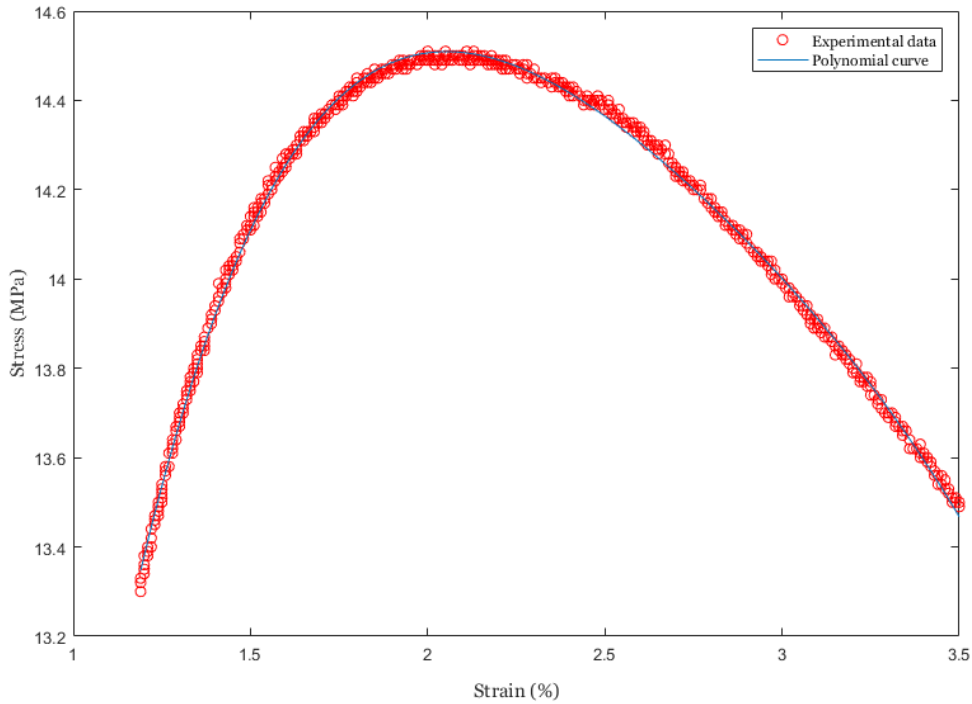
Figure 3.3 – Polynomial curve fitting to obtain compression data values

3.1.1 Concrete damage plasticity

The CDP (concrete damage plasticity) model is proved to be capable of modelling concrete and other quasi-brittle materials in all types of structures. To compute the inelastic behaviour of the materials, this model uses concepts of isotropic damaged elasticity in combination with isotropic tensile and compressive plasticity. It should be noted that unlike discrete crack models, the CDP model uses damage indices such as DAMAGEC for compression and DAMAGET for tension to approximate cracks, but is not capable of providing accurate crack data such as crack width. (Refer Appendix D for formulation)

Hardening data for CDP model are given by in terms of an inelastic strain, ε_{in}, instead of plastic strain, ε_{pl}. The compressive inelastic strain is defined as the total strain minus the elastic strain corresponding to the undamaged material,

$$\varepsilon_{in} = \varepsilon_t - \varepsilon_{el}, \text{ where } \varepsilon_{el} = \sigma_y/E$$

Five other parameters are defined in the CDP model: the dilation angle in degrees, the flow potential eccentricity, the ratio of initial equi-biaxial compressive yield stress to initial uniaxial compressive yield stress, the ratio of the second stress invariant on the tensile meridian to that on the compressive meridian and the viscosity parameter. Their values, 31, 0.1, 1.16, 0.667 and 0 as mentioned in [11] and [12]. The tensile yield stress is considered as 4.5 MPa and the compressive yield stress is adapted from the experimental results.
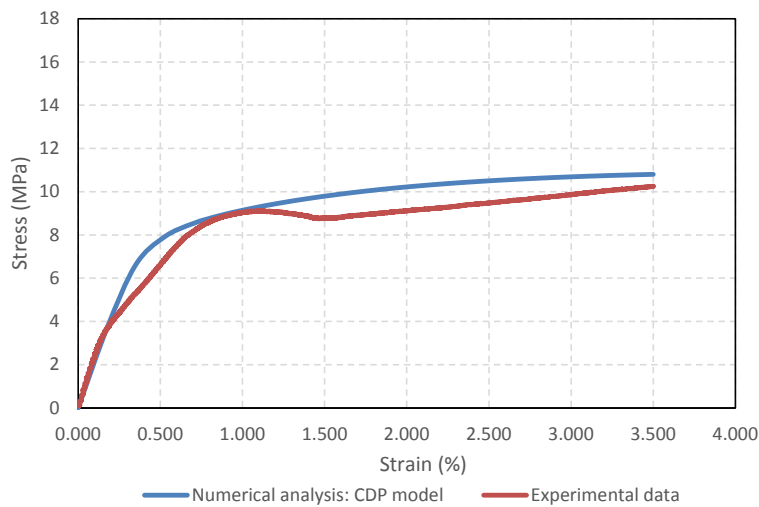
(a)
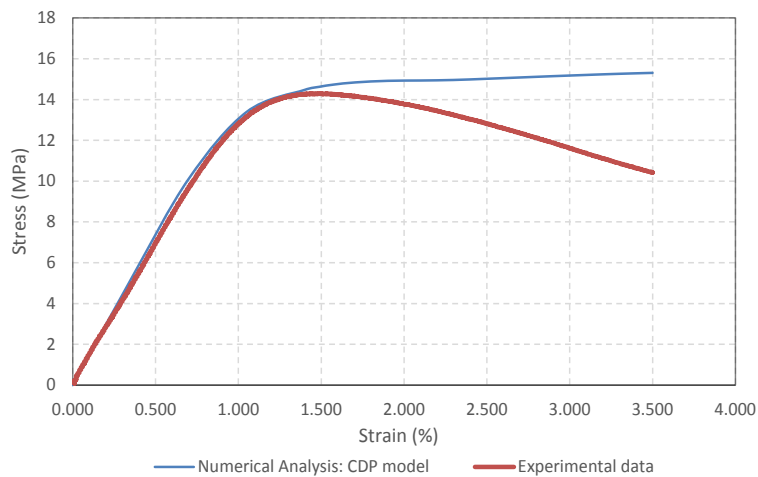
(b)

Figure 3.4 - Comparison of experimental data with numerical analysis considering CDP model for samples S1_G3.5% (a), S5_G3.5% (b)

The numerical analysis curve in Figure 3.1 (a) is very close to that of the experimental curve even in the plastic region. The plastic region in Figure [3.1(b)] differs quite a bit between both the curves. Overall, clearly, the elastic region in both the graphs perfectly match with each other, while in the plastic region, the CDP model works fine when there's only hardening rather than softening in the curve.

3.2.2 Crushable foam model

CF (Crushable foam) model is used to simulate the ability of not only crushable foams but also other crushable materials other than foams to deform in compression. This model is based on the assumption that the resulting deformation is not recoverable instantaneously and can be idealized as plastic for short duration events. FE predictions of strength and damage locations obtained with the CF model for vertebral body samples were compared with experimental results and one of the most accurate damage models in [13], [14] and [15]. The predictions were good enough to propel into choosing this model.

The hardening was set to volumetric. The CF model with volumetric hardening uses a yield surface with an elliptical dependence of deviatory stress on pressure stress. It assumes that the evolution of the yield surface is controlled by the volumetric compacting plastic strain experienced by the material. The yield surface evolves in a self-similar fashion and the shape factor can be computed using the initial yield stress in uniaxial compression, σ_c^0, the initial yield stress in hydrostatic compression, p_c^0 and the yield strength in hydrostatic tension, p_t. (Refer Appendix D for formulation)

To define the shape of the yield surface, the values of k and k_t should be provided. These material parameters of CF plasticity model were identified from [13]. The ratio of initial yield stress in uniaxial compression to initial yield stress in hydrostatic compression, compression yield stress ratio (k) is taken as 1.181 and the ratio of yield stress in hydrostatic tension to initial yield stress in

hydrostatic compression, Hydrostatic yield stress ratio (k_t) was considered to be 0.54.
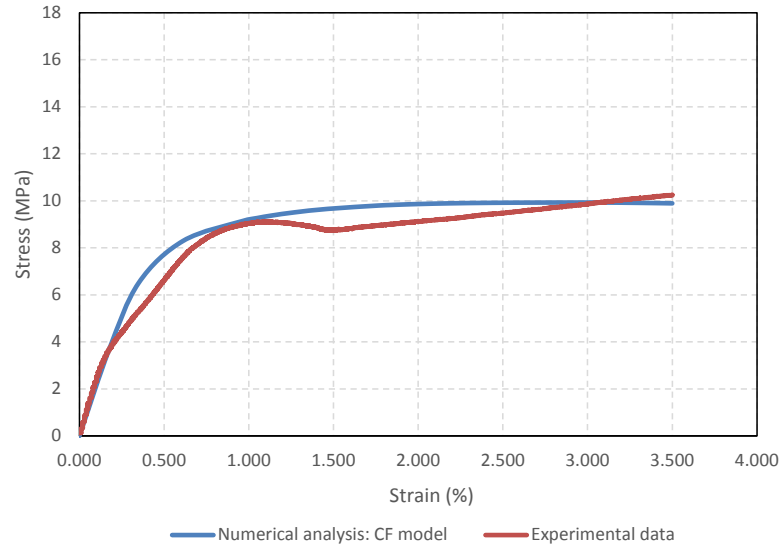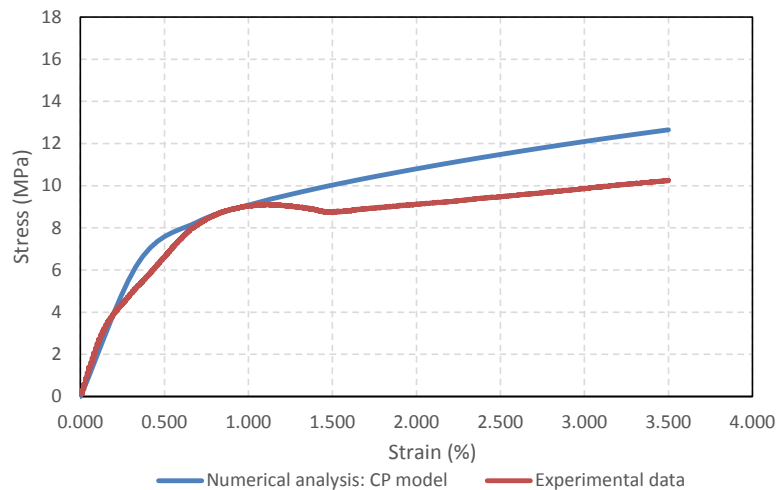

(a)
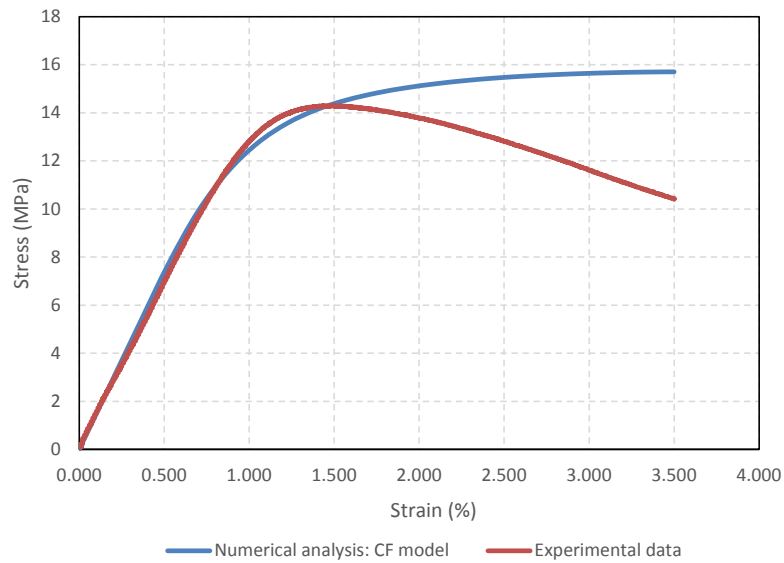
(b)

Figure 3.5 - Comparison of experimental data with numerical analysis considering CF model for samples S1_G3.5% (a), S5_G3.5% (b)

Figure 3.5 compares the stress in MPa and Strain in % of experimental results and numerical analysis considering CF model. Though the results look quite similar to that of CDP model, a closer look is needed to decide the better match and will be compared in detail in the later part of the chapter.

3.2.3 Classical Plasticity Model

The classical plasticity model allows defining the yield and inelastic flow of a material at relatively low temperatures, where loading is relatively monotonic and creep effects are unimportant. The yield stress and plastic strain are considered from the experimental data as a tabular function.

The hardening considered was assumed to be isotropic, which means that the yield surface changes size uniformly in all directions such that the yield stress increases (or decreases) in all stress directions as plastic straining occurs. Although the model is referred to as a "hardening" model, strain softening or hardening followed by softening can be defined. (Refer Appendix D for formulation)


(a)
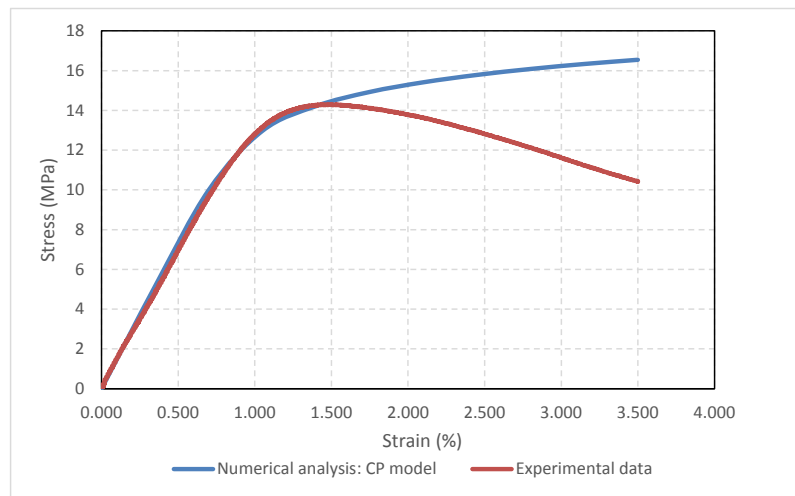
(b)

Figure 3.6 - Comparison of experimental data with numerical analysis considering CP model for samples S1_G3.5% (a), S5_G3.5% (b)

Clearly, the peak value of both the curves varies quite a lot from the experimental data. Though CP model appears to be a bad match compared to the other two models, it will be reviewed in the next section.

3.2.4 Comparison of plasticity models

After individually examining the three plasticity models, the better suitable model for this study has to be chosen to proceed further to implementing damage in the analysis.
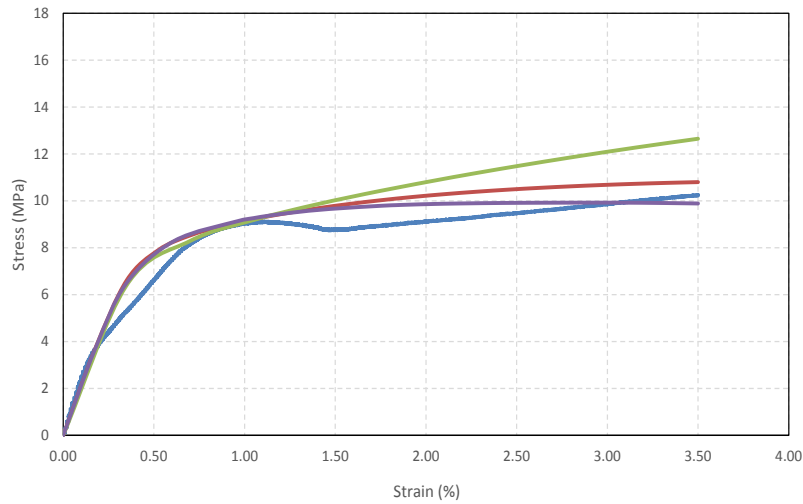
(a)
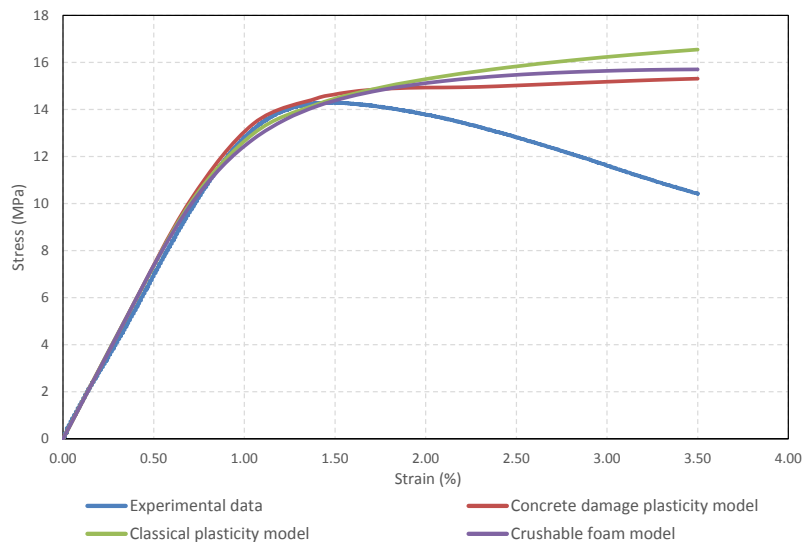
(b)

Figure 3.7 – Comparison of three plasticity models with experimental data for samples S1_G3.5% (a) and S5_G3.5% (b)

Clearly, CP model hasn't been as effective as the other two plasticity models. CF and CDP models have been the closest approximations to Sample 1 and Sample 2 respectively. Considering the peak stresses for sample 1, the CF model stands at 9.89 MPa in contrast to CDP which is 10.8 MPa, while the experimental peak stress is 10.23 MPa. For sample 2, the experimental peak

stress is 10.41 MPa and the CDP and CF models stand at 15.3 MPa and 15.7 MPa. Though there's not much difference between them, CDP model has been chosen for this study over CF model as it is slightly closer to the experimental data.

3.3   Implementation of Damage

After choosing the plasticity model, the next step was to implement damage. A variety of procedures were available in the literature to implement damage in the model including using a user subroutine like UMAT. But bearing in mind the time constraints of this project, we found a better option in CDP model itself. As CDP model uses damage indices for compression for computation, the damage data input plays a crucial role here. The inelastic strains and their respective damage parameters are to be entered. To calculate this, experimental data was taken as a reference once again. Since experimental tests were conducted on at least 30 samples, it was adequate enough to plot a curve between damage and plastic strains of all of them, resulting in Figure 3.6. A mathematical function was generated following the trend of this curve. A polynomial equation was preferred to exponential and linear interpolation owing to the value of higher R-square, which is a statistical measure of how close the data are to the fitted regression line or curve.
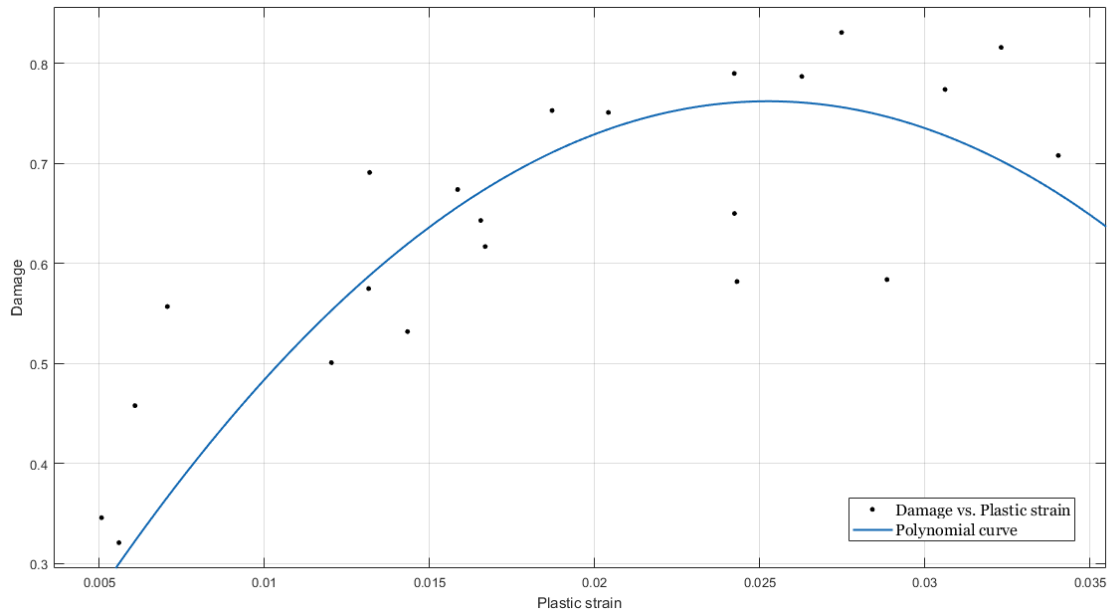
Figure 3.8 – Polynomial curve fitting for damage vs plastic strain

The polynomial equation obtained is:

$$D = -1195*(\varepsilon_p)^2 + 60.36*(\varepsilon_p)$$

Where D – damage

ε_p – plastic strain

After acquiring the polynomial function from the curve, the data should be aptly provided to ABAQUS. Unloading data are provided in terms of compression damage curves. ABAQUS uses the following formula to convert the inelastic strain to plastic strain: $\varepsilon_{el} = \sigma_y$

$$\varepsilon_{pl} = \varepsilon_{in} - \frac{d}{1-d} \times \frac{\sigma_y}{E} \qquad \ldots(\text{Eq. 1})$$

Where d= uniaxial compression damage variable

In ABAQUS, the damage variables are treated as non-decreasing material point quantities. At any increment during the analysis, the new value of each

damage variable is obtained as the maximum between the value at the end of the previous increment and the value corresponding to the current state. A stiffness recovery variable, w_c, explains ABAQUS if the compressive stiffness is fully recovered and accepts a range of values from 0 to 1. In this case, w_c was considered as 0, which means that there is no stiffness recovery.

Provided the values of plastic strain are available from the experimental data, inelastic strain was computed from (Eq. 1). Thus, uniaxial compression damage variable was defined as a tabular function of inelastic strain.

Chapter 4

Results and Discussion

The results of this study after implementing damage and plasticity models will be discussed in this chapter. This will be followed by considering a common material model obtained from the mean values of experimental results for four samples and understand their behaviour. An analysis of Young's modulus calibration factors has been made to comprehend the parameters that affect it.

4.1 Final results

To achieve the exact replica of the experiments, the sample had to be unloaded, which is also used to calculate numerically obtained damage. The unloading part was achieved by applying a tensile displacement on the sample in a different static step using the same incrementation values as in the loading part. For a sample that was loaded to 5% strain, the displacement applied during loading was 0.105 mm, considering the fact that the side of the cube is 2.104 mm. The same sample was unloaded until zero stress was reached.

The results of the analysis before the implementation of damage are shown in Figure 4.1. Looking at them, it is very clear that the material model is able to predict the elastic and the yield behaviour of the curve. But, the post-yield behaviour is not completely satisfying, but definitely considerable. Figures (a), (b) and (c) have closer approximations of post-yield behaviour rather than the rest. The stress at the last strain point is bigger for (f) and (g) when compared to the others. Considering these, a careful observation illustrates that the material model is able to predict the post-yield behaviour well if there is not much softening in the curve. Moreover, even if there is softening in (d), there is not

much difference between the numerical analysis and experimental data as the sample is strained to a relatively lesser displacement. This can be noticed by comparing the curves (d) and (f). Considering the fact that (f) is loaded to 5% strain, the material model appears not to predict its behaviour better than (d), which is loaded to 3.5% strain. Aggregately, the material model predicts the overall behaviour of the curve better in the rest when compared to (d), (e), (f) and (h). In (i), though there is softening, the post-yield behaviour looks more promising than the elastic behaviour. There seems to be an error in the elastic region in the experimental data of (j), as the curve in the proportionality limit doesn't appear to be straight, which is the reason for the mismatched curve. Overall, before the implementation of damage, the material model is able to predict the pre-yield behaviour and the post-yield behaviour until the sample softens beyond the yield point. In the next section, the response of the material model when damage is implemented will be discussed.
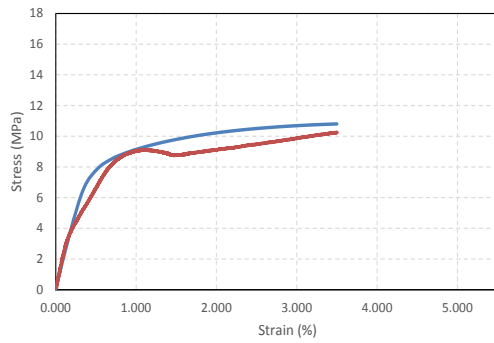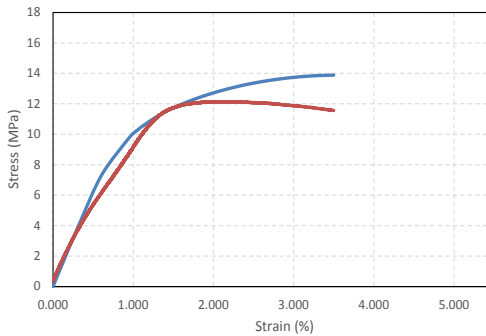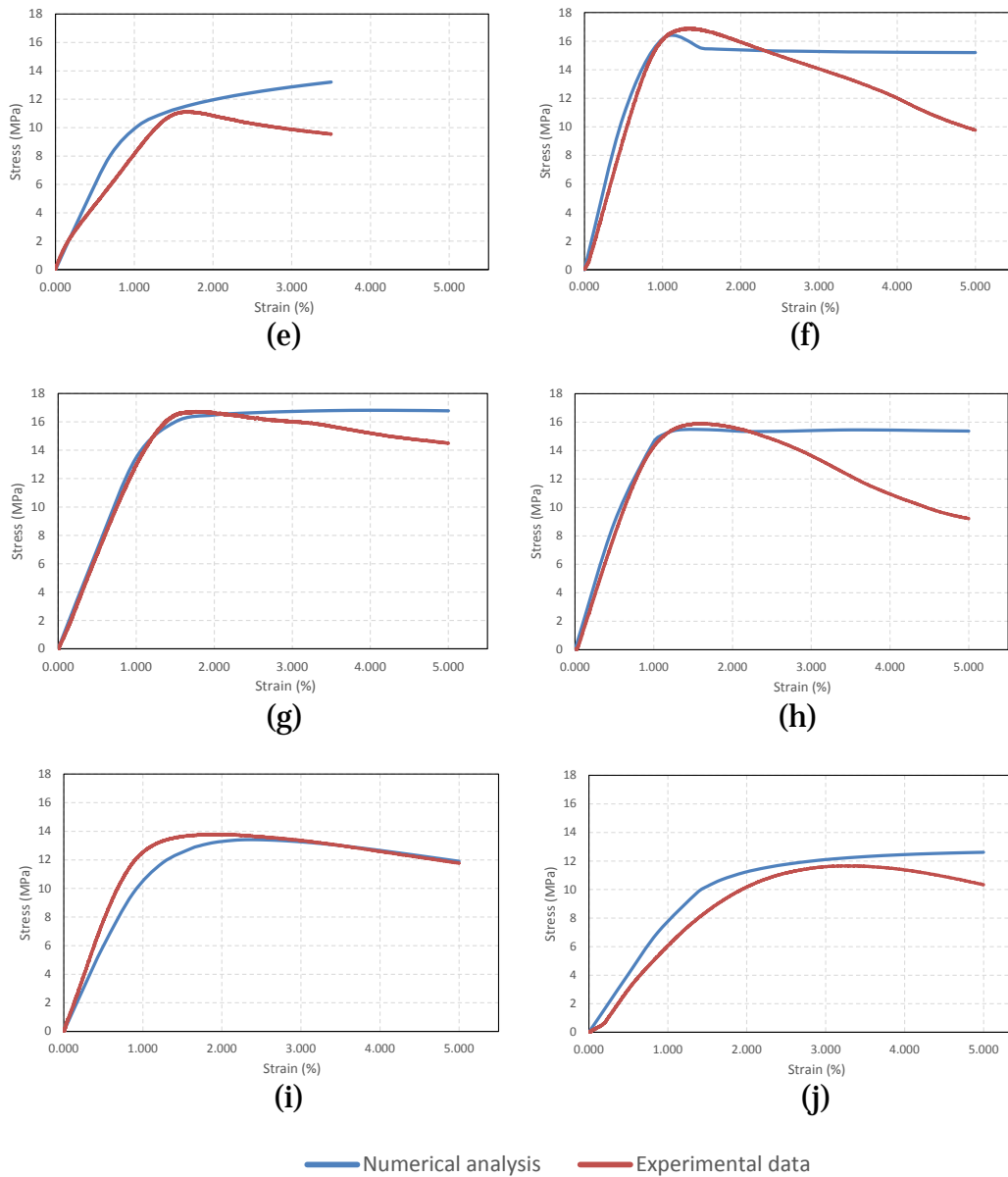

(a)
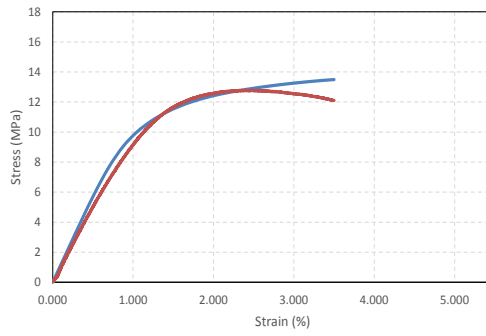
(b)
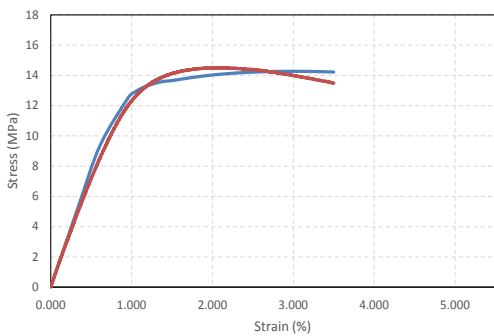
(c)

(d)

Figure 4.1 – Results comparing Numerical analysis to experimental data before implementing damage in the material model: (left to right) S1_G3.5%, S2_G3.5%, S3_G3.5%, S4_G3.5%, S6_G3.5%, S1_G5%, S2_G5%, S3_G5%, S4_G5%, S5_G5%
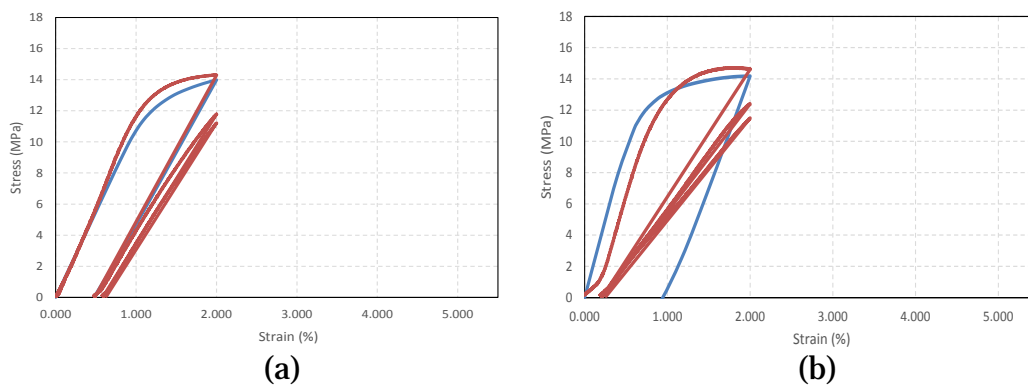
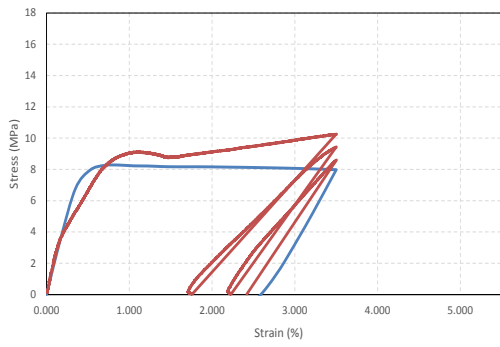## 4.1.1 Results after implementing damage

Damage was then implemented by introducing damage parameters and their respective inelastic strains in the CDP model, as explained in Chapter 3. The

results obtained are shown in Figure 4.2. In total, 12 samples were analysed, two from G2% and five each from G3.5% and G5%. During the experiments, the sample was unloaded and loaded twice to get rid of residual stresses, which can be seen in the curve. The post-damage E is calculated after the third unloading.

The material model was able to predict damage of the bone in few samples while wasn't very accurate in few others. The damage predictions look accurate enough in (a) and (c). Taking a closer look, in (a) and (c), there is hardening in the curve. Though curves (b) and (d) show hardening behaviour too, the damage predictions weren't too accurate. The model appears to predict the damage well in (h) and (k) than in (i) and (j), to prove the same point as above. Overall, the damage model is able to capture maximum stresses, while after yielding, it is not powerful enough in predicting the behaviour.

The damage obtained numerically and induced experimentally are referred to as numerical damage and experimental damage and the results are henceforth shown in Figure 4.2 and Table 4.1.
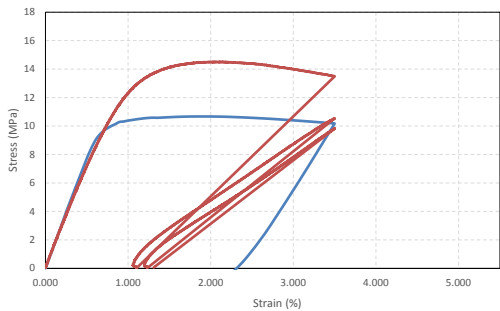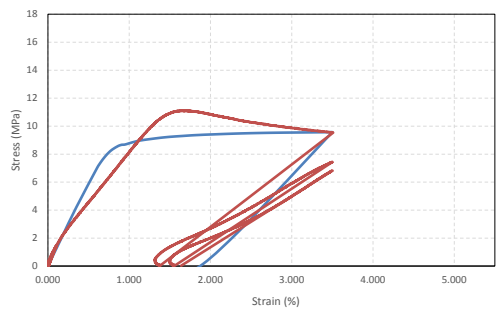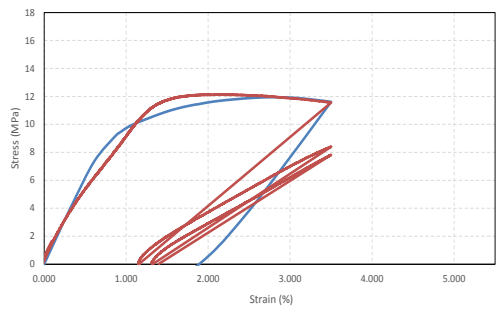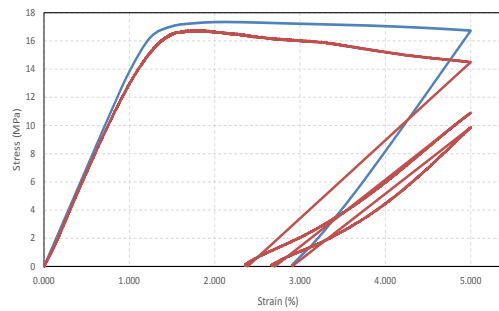

(a)   (b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

(j)

(k)   (l)

Numerical analysis   Experimental data
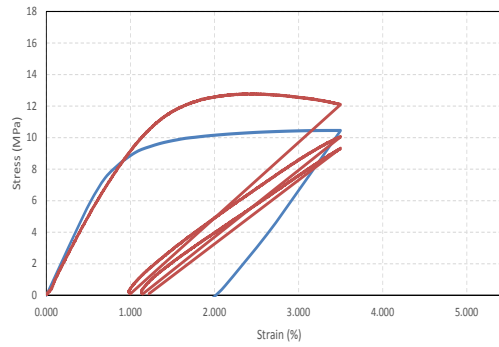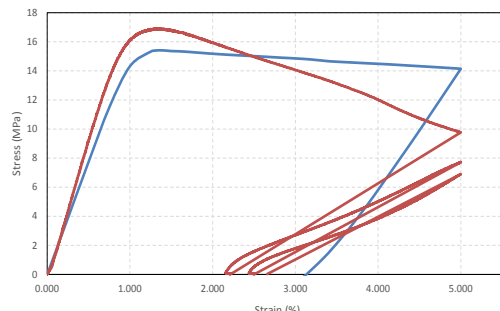
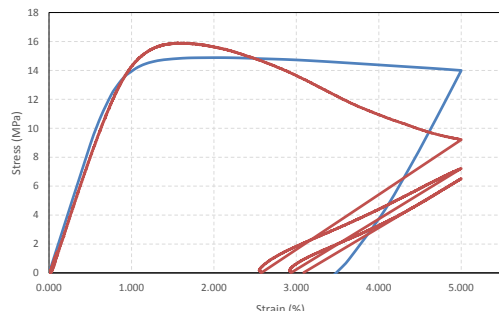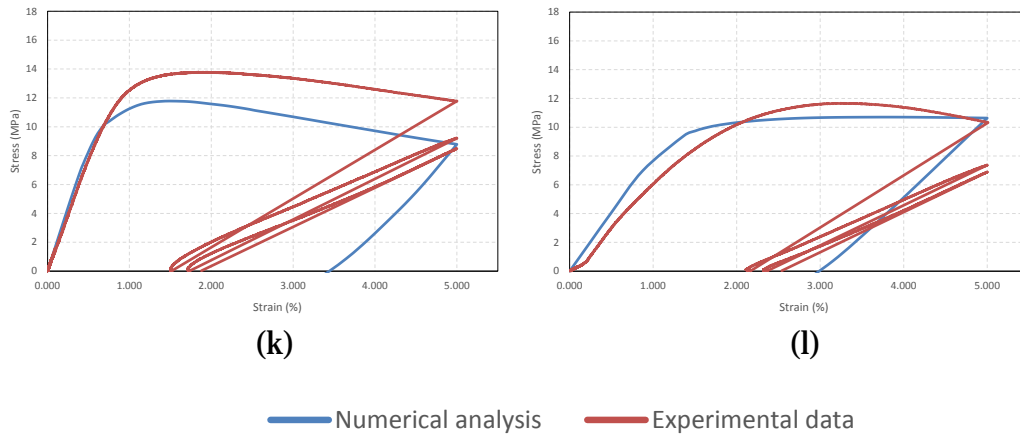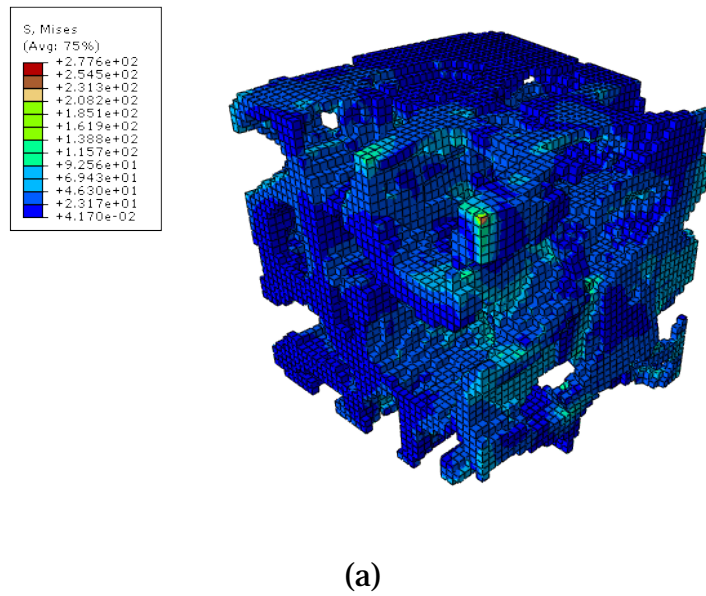*Figure 4.2 – Results comparing Numerical analysis to experimental data after implementing damage in the material model: (left to right) S1_G2%, S2_G2%, S1_G3.5%, S2_G3.5%, S3_G3.5%, S4_G3.5%, S6_G3.5%, S1_G5%, S2_G5%, S3_G5%, S4_G5%, S5_G5%*
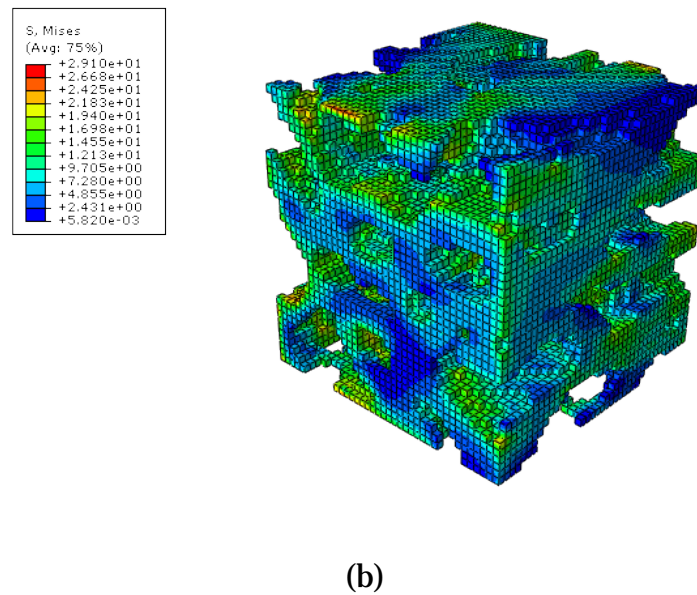

(a)

(b)

Figure 4.3 - Stress concentrations post-damage in S1_G3.5% and S2_G5%

The stresses in Figure 4.3 (b) is relatively more concentrated in few weak areas when compared to that of Figure 4.3 (a). Neglecting the magnitude of the stresses, it appears that the concentration is more towards the corners of the sample. Both the models appear to bend rather than buckle at the points with higher stress concentrations. The damage obtained numerically is closer to experimental damage for smaller strains rather than larger ones. Among the samples loaded to smaller strains, specifically, the samples undergoing hardening were approximated well with the numerical model. Even in the samples belonging to larger strains, the curves where no softening is observed are predicted well like in Figure 4.2 (h). As in Table 4.1, except for samples 10 and 12, the rest of the samples have less than 20% difference between numerical and experimental damage. As illustrated in error (%) column in Table 4.1, while the numerical damage of 7 out of 12 samples falls within 15% with the experimental damage, the variance for 3 samples is hardly more than 10%. Observing the trend lines of experimental data and numerical analysis in Figure 4.4, it is very clear that the error increases with increase in strain percentage to which the samples are loaded, proving that this model isn't very accurate for samples loaded with higher strains.

S.No	Sample	Experimental damage	Numerical damage	Error (%)
1	S1_G2%	0.320	0.392	-7.2
2	S2_G2%	0.390	0.277	11.3
3	S1_G3.5%	0.582	0.544	3.8
4	S2_G3.5%	0.501	0.403	9.8
5	S3_G3.5%	0.575	0.439	13.6
6	S4_G3.5%	0.643	0.483	16
7	S6_G3.5%	0.532	0.376	15.6
8	S1_G5%	0.584	0.448	13.6
9	S2_G5%	0.787	0.588	19.9
10	S3_G5%	0.774	0.435	33.9
11	S4_G5%	0.753	0.631	12.2
12	S5_G5%	0.650	0.322	32.8

Table 4.1 - Experimental and numerical damage parameters of samples

Overall, there are definitely few limitations using these damage and plasticity models for prediction of damage. There were few assumptions made during this analysis. Firstly, the model was assumed to be isotropic while it is orthotropic. The viscoelasticity was considered to be zero and the material was assumed to be elastic-perfectly plastic. Also, choosing a small representative cube of side 2 mm while the original sample is 30 mm long could have also played a role in the damage limitations. Just like a sample can have different local and global Young's moduli, a smaller value of microscopic damage could also amount

to a larger macroscopic damage. Pertaining to all of the above factors, the results obtained are definitely satisfactory.
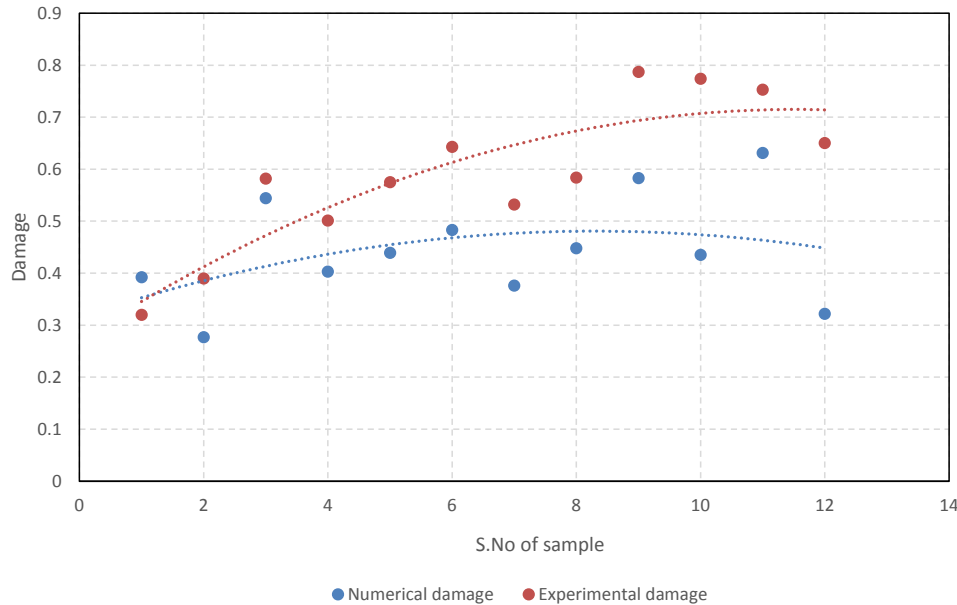

Figure 4.4 – Numerical damage vs Experimental damage of 12 samples

4.2   Considering common properties

In the previous section, the behaviour of the samples when loaded to the variety of displacements with their respective material properties has been looked at. But considering the fact that all the samples belong to a single material, a trabecular bone, it would be interesting to observe the behaviour of the samples when a common material model defines them all. This is one of the extraordinary upsides of Finite Element Modelling. To do this, mean values of all the required material properties were calculated from the experimental data as shown in Table 4.2. Further, these properties are used as input parameters in the material model of ABAQUS wherever necessary.

Group	E	Damage	Plastic Strain	Yield strain	Yield stress
G3.5%	1549.3	0.602	0.0158	0.96	11.6
G5%	1549.3	0.745	0.0267	1.08	13.4

Table 4.2 – Material properties (mean values) considered for analysis

A total of four samples, two samples from G3.5% and G5% were considered for this analysis. The Young's modulus has been taken the same for both the groups as it is a property of microstructure. The damage, plastic strain, yield strain and the yield stress properties were considered as an average of all the samples of their respective groups.

(a)   (b)

(c)   (b)

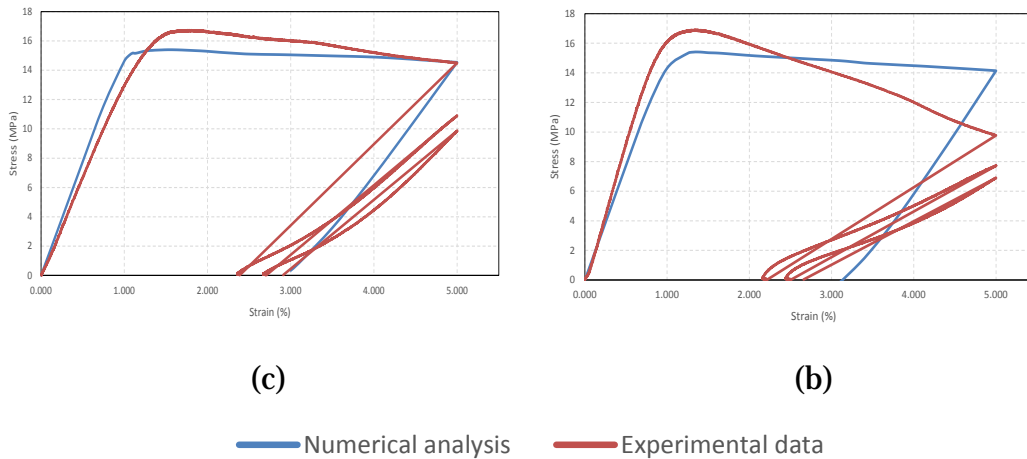━━Numerical analysis   ━━Experimental data
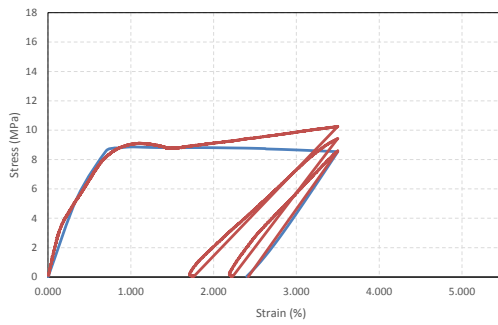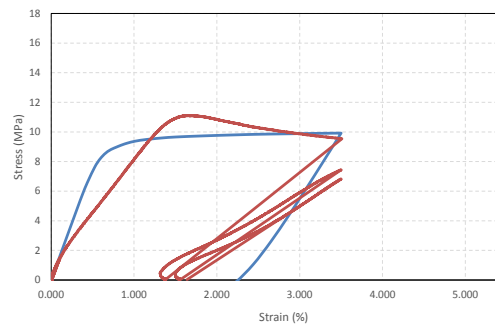
Figure 4.5 - Results comparing Numerical analysis to experimental data after implementing damage and considering average properties: (left to right) S1_G3.5%, S5_G3.5%, S2_G5% and S1_G5%

Owing to the difference between the value of average E and the values of E of each sample, there will be a deviation in the elastic region of the curve unless E of the respective sample matches with mean E. The same theory applies to Yield stresses as well. Interestingly, in Figure 4.5 (a), for S1_G3.5%, the mean property material model was able to predict its post-yield behaviour more accurately than its own properties. Though the numerical damage is higher than the experimental damage induced, it is closer than the previous analysis. While S1_G5% and S2_G5% have almost the same damage values when mean values and their original properties were considered, the damage achieved for S4_G3.5% wasn't as good as considering its original values.

In Table 4.3, numerical damage using original values refer to the damage obtained in the previous section, where the unique material properties of each sample were used for analysis. Looking at the behaviour of the samples, clearly, the damage achieved depends on how close the mean values are to its original values. The only case that questions this behaviour is S1_G3.5% because the damage obtained in this case is a closer approximation to experimental results rather than the original results. Though the calibrated mean and original values

of E are 6243.67 MPa and 8507 MPa, the deviation in the elastic region of the experimental curve made the numerical analysis curve in Figure 4.5 (a) look closer. The damage parameters and inelastic strains used during the analysis considering mean values were higher and lower respectively when compared to its original values. It means a higher damage is induced in the model at a much lower inelastic strain which is the reason for the higher value of damage output. On the contrary, a lower damage was induced for the rest of the three samples analysed in this section which resulted in a lower value of damage output. So, the approximation of damage strictly depends on the amount of damage induced relative to their respective inelastic strains. Certainly, all the limitations of the concrete damage plasticity model like softening of the sample apply here too.

Sample	Experimental damage	Numerical damage using mean values	Numerical damage using original values	Error (%)
S1_G3.5%	0.582	0.599	0.544	-1.7
S4_G3.5%	0.643	0.312	0.483	33.1
S1_G5%	0.580	0.445	0.448	13.5
S2_G5%	0.790	0.583	0.588	20.7

Table 4.3 – Bone samples, their experimental and numerical damages

4.3   Calibration Analysis of Young's modulus

As discussed earlier, the calibration factor is the ratio between the local and global Young's moduli. The calibration factor inherently depends on the microstructure of the sample, in this case, the representative cube chosen. Since local cubes were chosen from different samples belonging to different microstructures, the standard deviation of calibration factors is shown in Table

4.1. Except for S6_G3.5%, all the other samples lie within 2.5 from the mean value. The peculiar factor of this sample could be due to many factors like porosity, local trabecular connections etc.

In this section, an in-depth calibration analysis has been performed initially considering eight samples of two from each strain group. The calibration factors of eight different samples pre and post damage are as shown in Figure 4.6.

Sample	C_f	Standard deviation
S2_G1%	3.06	1.52
S1_G1%	3.17	1.41
S2_G2%	4.34	0.24
S1_G2%	2.43	2.15
S1_G3.5%	4.04	0.54
S2_G3.5%	5.04	0.46
S3_G3.5%	4.32	0.26
S4_G3.5%	2.15	2.43
S6_G3.5%	13.41	8.83
S2_G5%	2.46	2.12
S4_G5%	5.99	1.41

Table 4.4 – Standard deviation of calibration factors of samples

The pre-damage C_f was calculated subjecting the sample to a displacement equivalent to 0.5% strain. An assumed value of E was used as an input and the sample was strained to 0.5%, further obtaining output E from the resulting stress-strain data. Similarly, the post-damage C_f was obtained after straining the samples to strains with respect to the group they belong to.
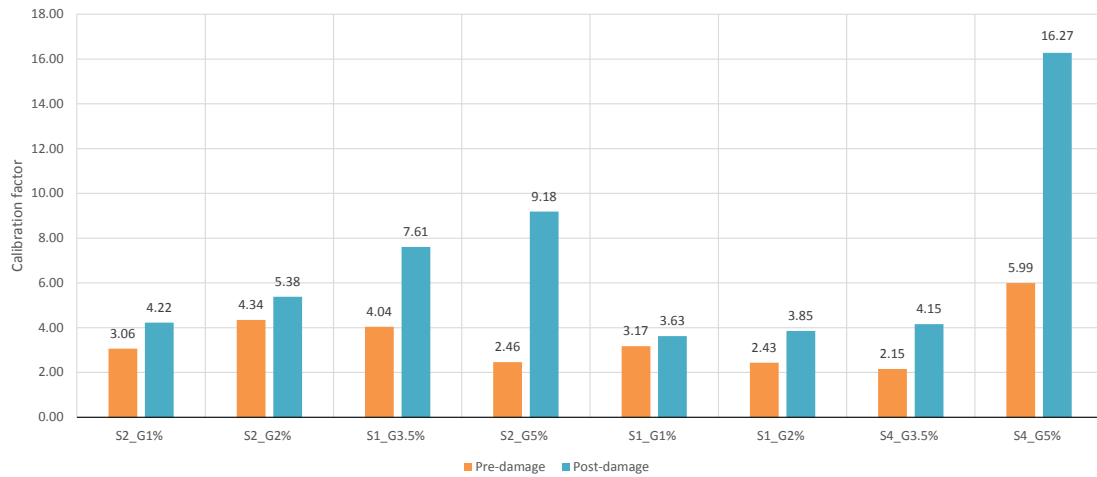
Figure 4.6 – Calibration factors of the samples S2_G1%, S1_G1%, S2_G2%, S1_G2%, S1_G3.5%, S4_G3.5%, S2_G5%, S4_G5% before and after damage

Though the pre-damage C_f doesn't lie in a specific range, it's clearly evident that the post-damage C_f keeps increasing with increase in the strain. The post-damage C_f of a sample that is loaded to 5% strain appears to be significantly higher than a sample loaded to 1% strain. But the pre-damage C_f per se depends on the microstructure of the sample, which is unique to each of them. Micro-structure includes factors like porosity, the connectivity of the elements in the voxel-based model etc. Since the sample considered in this study is a local representative cube of side 2.104 mm, similar cubes at different areas of a single sample could yield different calibration factors.

Since pre-damage C_f is constant and depends on the microstructure and the post-damage C_f looks like it depends on the displacement to which the sample is loaded and the pre-damage C_f, a factor was needed to nullify the effects of both the factors on this trend. It makes it easier to generalise the trend for all the samples with variant trabeculae.

$$C_f \, \text{ratio} = \frac{\text{Post-damage } C_f}{\text{Pre-damage } C_f}$$ *Eq. (1)*

C_f ratio depends on the change in the calibration factor rather than depending on the microstructure which is unique to a single sample. In Figure 4.6, though the sample *S2_G5%* and *S4_G5%* has a very divergent pre-damage and post-damage C_f, the ratio between their respective calibration factors for each sample tend to be almost the same, which is 2.4 for *S2_G5%* and 2.71 for *S4_G5%* (Figure 4.7). This explains the need for a parameter which doesn't depend on the microstructure of the samples to understand the factors that influence them.
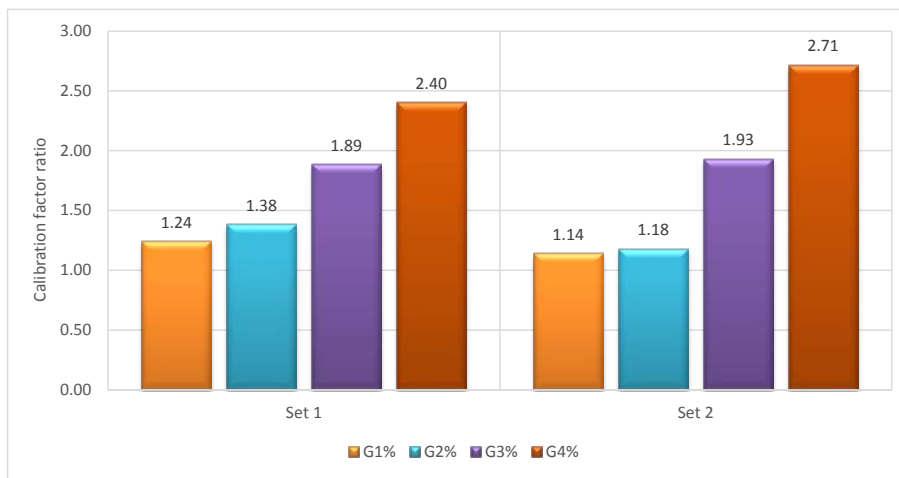

Figure 4.7 – Trend of C_f ratio of eight different samples

The eight samples considered in this analysis were divided into two sets, Set 1 and Set 2, each set comprising one sample from each strain group. In figure 4.7, it can be observed that the ratio keeps increasing with increase in the strain to which the sample is loaded. Since it appears like the C_f ratio depends on the strain to which the sample is loaded to, it would be more sensible to predict the trend and draw some conclusions when the samples belong to a single strain group. In this process, the calibration factors of five samples from group 3 were compared.
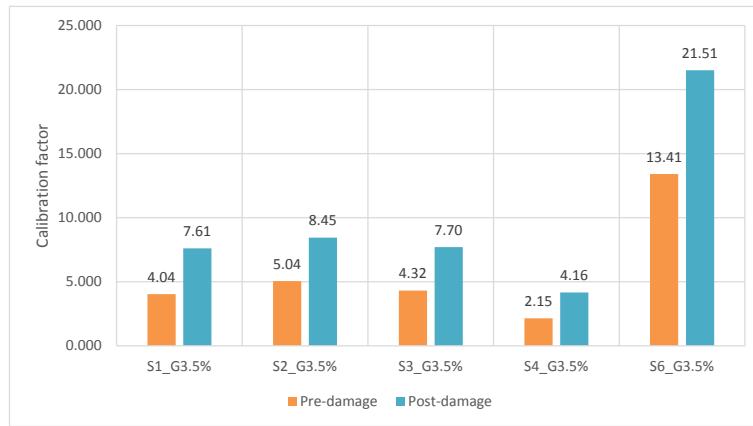
(a)
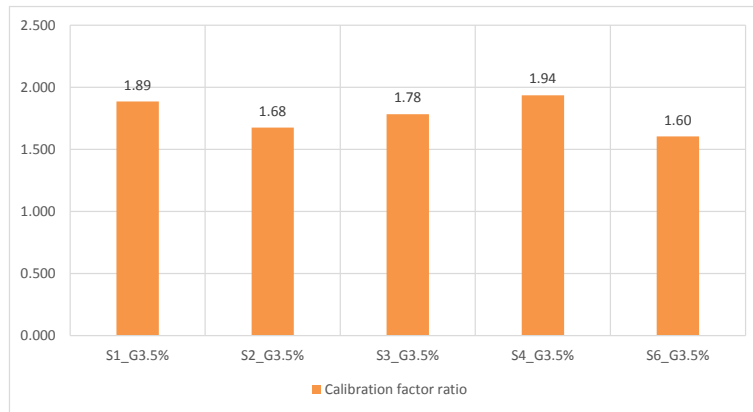
(b)

Figure 4.8 - Charts exhibiting the trend of (a) calibrations factors before and after damage and (b) calibration factor ratios of five samples from G3.5%

The pre-damage C_f of S4_G3.5% and S6_G3.5% are lower (2.15) and significantly higher (13.4) respectively when compared to the rest (Figure [4.2 a]). This could be influenced by several factors like local trabecular connections, porosity etc. Since porosity depends on the number of elements in a voxel-based model, different samples with different porosities were chosen to check if a higher number of elements leads to higher calibration factors. There wasn't a specific trend on which the calibration factors varied, which infers its multiple dependencies. Nevertheless, it is clear from Figure 4.8 (b) that the C_f ratio falls in the same range as the others demonstrating that it doesn't depend on the pre-

damage C_f. It implies that the C_f of any sample may vary in scale but not in the factor itself. This huge variation could also be the result of considering a local representative cube rather than the whole sample, entailing the weaker sections of the bone.

There are not many inferences that can be obtained from Figure 4.8 (a), considering the fact that there is no correlation between the calibration factors of all samples. Whereas, the C_f ratio appears to fall in the range of 1.6 to 2.2 for all the samples as illustrated in Figure 4.8 (b). Taking a closer look at the figures in Table 4.1, it gives a perception that damage could influence calibration factors considering the fact that both are directly dependent on E. Damage is obtained by subtracting the ratio of post-damage E to the pre-damage E from 1, in correlation with the calibration factors, which is the ratio of local and the global E. Considering pre-damage E to be constant throughout the analysis, the post-damage E depends on the amount of stiffness degradation or simple damage that a sample undergoes. The post-damage C_f, being the ratio of calibrated pre-damage E to the post-damage E, directly depends on the latter parameter since the former one is constant. But post-damage E depends subsequently on damage.

Sample	Pre-damage		Post-damage		C_f ratio	Damage (Numerical)
	E (MPa)	C_f	E (MPa)	C_f		
S1_G3.5%	247.82	4.035	112.96	8.853	2.194	0.544
S2_G3.5%	198.37	5.041	118.40	8.446	1.675	0.403
S3_G3.5%	231.65	4.317	129.92	7.697	1.783	0.439
S4_G3.5%	464.99	2.151	240.27	4.162	1.935	0.483
S6_G3.5%	74.58	13.408	46.49	21.510	1.604	0.376

Table 4.5 – Five samples and their respective calibration factors and numerical damage parameters

All the above factors contributed to the integration of damage parameter in Figure 4.9. The C_f ratios and damages of the samples ranged from 1.6 and 0.38 to 2.19 and 0.54 with a consistent increase in C_f ratio when there is an increase in

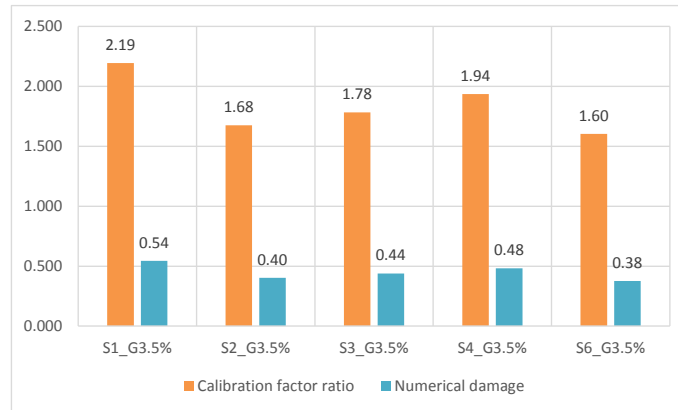damage. Looking at it, it's evident that the dependence on damage is highly probable.

Figure 4.9 – C$_f$ ratios and their respective numerically obtained damage parameters

Chapter 5

Conclusions and Scope of Improvements

This chapter aims to give a bigger perspective to this study, the results achieved and possible scope of improvements in this area.

To brief up, in this study, µCT images of trabecular bone samples were used as the input parameters to generate a voxel-based model for finite element modelling using a series of steps. Finite element analysis (FEA) is a 'non-destructive' simulation tool that can estimate mechanical properties of biomaterials when combined with 3D imaging modalities such as micro-computed tomography. FEA facilitates numerous studies considering various material models, subjecting a single sample to various strains, which is experimentally impossible.

Since a voxel-based model is readily extracted from each bone sample, numerous analyses can now be made on it including a fatigue test. Though a compression test is performed in this study, a fatigue test could rather replicate the real-life loads that a bone is subjected to.

A small representative cube has been chosen for analysis in this study. So all the results obtained should be noted as pertaining to local areas rather than being global. The local cubes were selected giving the trabecular structure more importance by ignoring the samples that looked more porous. Studies can also be made on the whole sample with a higher number of elements with better computing machines. That would indeed replicate the experimental results better.

Concrete damage plasticity model was chosen from three different material models to define plasticity. The damage predictions in this study using a concrete plasticity model are accurate for few samples, satisfactory for few more and unsatisfactory for the rest. Accurate results were limited to hardening behaviour of the curves and loads to lower strains. Based on this analysis, the damage mechanism can be understood well from various samples consisting of distinct porosities. The morphological and mechanical parameters of the samples post-damage could give an insight into its strength and thereby indicating the diseased bones. This analysis could be helpful in development of bioinspired materials in various fields as it gives a deeper understanding of the micromechanics.

Looking at a bigger perspective, though this study only deals with numerically estimating how close the analysis stands in contrast with experimental results, the long-term application of these analyses is particularly exciting because it provides valuable information for monitoring disease progress and treatment efficacy. Importantly, osteoporosis is a type of disease that effect's most of the post-menopausal woman and also found in men [18]. Though DXA is typically used for diagnosis of osteoporosis, since it doesn't provide assessment on bone microstructure, which is the key determinant of bone strength, High-resolution peripheral quantitative computed tomography (HR-pQCT) is used [22]. Owing to the same procedure as in this study, importing the HR-pQCT images of osteoporotic bones into an analysis software and comparing the research values with normal bones and also using them as a caution for osteopenic and normal bones could also be done. Numerous other studies are also being performed on three-dimensional computational models simulating the regeneration of bones, fracture healing processes etc. on the same path as this study.

List of Figures

List of Tables

Bibliography

[1]   Busscher I., Ploegmakers J.J., Verkerke G.J., Veldhuizen A.G., 2010. Comparative anatomical dimensions of the complete human and porcine spine. European Spine Journal 19, 1104–1114.

[2]   Paola Bin, Paola Delbon, Mauro Piras, Mariano Paternoster, Pierpaolo Di Lorenzo, and Adelaide Conti, 2016. Donation of the body for scientific purposes in Italy: ethical and medico-legal considerations. Open medicine journal 11, 316-320.

[3]   Caon M, 2004. Voxel-based computational models of real human anatomy: a review. Radiation and environmental biophysics 42, 229-235.

[4]   Lengsfeld M1, Schmitt J, Alter P, Kaminsky J, Leppek R., 1998. Comparison of geometry-based and CT voxel-based finite element modelling and experimental validation. Medical Engineering and Physics 20, 515-522.

[5]   Nobuyuki Otsu, 1979. A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9, 62-66.

[6]   https://it.mathworks.com/help/matlab/, MATLAB documentation.

[7]   T. Guillén, Q. H. Zhang, G. Tozzi, A. Ohrndorf, H. J. Christ, J. Tong, 2011. Compressive behaviour of bovine cancellous bone and bone analogous materials, micro CT characterisation and FE analysis. Journal of the Mechanical Behaviour of Biomedical Materials, 4, 1452-1461.

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

[9]   M. Kinzla, U. Wolfram, D. H. Pahr, 2013. Identification of a crushable foam material model and application to strength and damage prediction of human femur and vertebral body. Journal of the Mechanical Behaviour of Biomedical Materials, 26, 136-147.

[10]   http://ABAQUS.software.polimi.it/v6.14/books/usi/default.htm, ABAQUS CAE User's Guide.

[11]   Yujie Huang, Zhenjun Yang, Wenyuan Ren, Guohua Liu, Chuanzeng Zhang, 2015. 3D meso-scale fracture modelling and validation of concrete based on in-situ X-ray Computed Tomography images using damage plasticity model. International Journal of Solids and Structures, 67, 340-352.

[12]   E. Dall'Ara, D. Pahr, P. Varga, F. Kainberger, P. Zysset, 2011. QCT-based finite element models predict human vertebral strength in vitro significantly better than simulated DEXA. Osteoporosis International, 23, 563-572.

[13]   E. Dall'Ara, B. Luisier, R. Schmidt, F. Kainberger, P. Zysset, D. Pahr, 2013. A nonlinear QCT-based finite element model validation study for the human femur tested in two configurations in vitro. The bone Journal, 52, 27-28.

[14]   E. Dall'Ara, R. Schmidt, D. Pahr, P. Varga, Y. Chevalier, J. Patsch, F. Kainberger, P. Zysset, 2010. A nonlinear finite element model validation study based on a novel experimental technique for inducing anterior wedge-shape fractures in human vertebral bodies in vitro. Journal of Biomechanics, 43, 2374-2380.

[15]   A. Garo, P. J. Arnoux, E. Wagnac, C.E. Aubin, 2011. Calibration of the mechanical properties in a finite element model of a lumbar vertebra under dynamic compression up to failure. Medical & Biological Engineering & Computing, Volume 49, Issue 12, pg 1371–1379.

[16]   Philippe K Zysset, Enrico Dall'Ara, Peter Varga, and Dieter H Pahr, 2013. Finite element analysis for prediction of bone strength. International bone and mineral society, 2: 386.

[17]   S. Patil ; B. Ravi, 2006. Voxel-based representation, display and thickness analysis of intricate shapes. Ninth International Conference on Computer Aided Design and Computer Graphics (CAD-CG'05)

[18]   M Sahiti Lahari ; Vijay ; M Anburajan, 2011. Finite element analysis of femur in the evaluation of Osteoporosis. 2011 3rd International Conference on Electronics Computer Technology.

[19]   Davide Ferrario. Effect of morphological and clinical parameters on damage accumulation in porcine trabecular bone.2016.

[20]   Bart Clarke, 2008. Normal bone anatomy and physiology. Clinical journal of the American society of Nephrology. 3(Suppl 3): S131–S139.

[21]   Gulgun Kayalioglu. The Vertebral Column and Spinal Meninges, 2009. The spinal cord. Pages 17-36.

[22]   Kyle K. Nishiyama and Elizabeth Shane. Clinical Imaging of Bone Microarchitecture with HR-pQCT. Current osteoporosis reports, 2013, 147-155.

Appendix A

Automatizing MATLAB Code to generate '*.inp' file

The following code includes only the required code for automatizing the numerical analyses. Two functions are called in the primary code, which will be described chronologically.

A.1 MATLAB Code

A.1.1 Primary code:

```matlab
clear variables;
clc;

[,coordinate,nset_sup,m_set]=IncidCoord(s(1,1), 1, s(1,2), 1, s(1,3),
biggest_c+1, k*Risoluzione,cubo);

displ=((cubo/2)*52.6*0.00001);
disp('Scrittura del file .inp')
createf=fopen(['descrittori_CEL2_comp_','k_',num2str(k),'.inp'],'w+');
stampa(createf,coordinate,incidenze,nset_sup,m_set,displ);
fclose(createf);

end
```

A.1.2 External function IncidCoord:

```matlab
function [ref_points] = IncidCoord(cubo)

ref_points=struct('ref_point_1_x',[],'ref_point_1_y',[],'ref_point_1_z
',[],'ref_point_2_x',[],'ref_point_2_y',[],'ref_point_2_z',[]);

ref_points.ref_point_1_x=((cubo-2)/4)*0.0526;
ref_points.ref_point_1_y=((cubo-2)/4)*0.0526;
ref_points.ref_point_1_z=((-2)/4)*0.0526;
ref_points.ref_point_2_x=((cubo-2)/4)*0.0526;
ref_points.ref_point_2_y=((cubo-2)/4)*0.0526;
ref_points.ref_point_2_z=((cubo-1)/2)*0.0526;

end
```

A.1.3 External function stampa:

```matlab
function stampa (file,ref_points,displ)
```

```
fprintf(file, '** Section: Section-1\n*Solid Section, elset=Volume,
material="Trabecular bone"\n,\n*End Part\n');

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

fprintf(file, '*Node\n       1,    %f,    %f,
%f\n',ref_points.ref_point_1_x,ref_points.ref_point_1_y,ref_points.ref
_point_1_z);

fprintf(file, '*Node\n       2,    %f,    %f,
%f\n',ref_points.ref_point_2_x,ref_points.ref_point_2_y,ref_points.ref
_point_2_z);

fprintf(file, '*Nset, nset=Set-9\n 1,\n*Nset, nset=Set-10\n 2,\n');

%Here the all the nodes of the top and bottom surfaces in the Z
direction are defined under two different sets.

fprintf(file, '*Nset, nset=ref_point_1\n 1,\n*Nset, nset=ref_point_2\n
2,\n');

fprintf(file, '*Surface, type=NODE, name=z_bottom_CNS_,
internal\nz_bottom, 1.\n');

fprintf(file, '*Surface, type=NODE, name=z_top_CNS_, internal\nz_top,
1.\n');

fprintf(file, '** Constraint: Bottom surface\n*Coupling, constraint
name="Bottom surface", ref node=ref_point_1,
surface=z_bottom_CNS_\n');

fprintf(file, '*Kinematic\n** Constraint: Top surface\n*Coupling,
constraint name="Top surface", ref node=ref_point_2,
surface=z_top_CNS_\n*Kinematic\n*End Assembly\n** \n** MATERIALS\n');

fprintf(file, '** \n*Material, name="Trabecular
bone"\n*Elastic\n1629., 0.3\n** \n** BOUNDARY CONDITIONS\n** \n**
Name: Fixed bottom surface Type: Displacement/Rotation\n');

fprintf(file, '*Boundary\nSet-9, 1, 1\nSet-9, 2, 2\nSet-9, 3, 3\nSet-
9, 4, 4\nSet-9, 5, 5\nSet-9, 6, 6\n** ------------------------------
------------------------------\n** \n** STEP: Step-2\n** \n*Step,
name=Step-2, nlgeom=YES, inc=1000000\n');

fprintf(file, '*Static\n0.1, 1., 1e-15, 0.1\n** \n** BOUNDARY
CONDITIONS\n** \n** Name: Load Type:
Displacement/Rotation\n*Boundary\n');

fprintf(file, 'Set-10, 1, 1\nSet-10, 2, 2\nSet-10, 3, 3, %f\n** \n**
OUTPUT REQUESTS\n** \n*Restart, write, frequency=0\n** \n',displ);

fprintf(file, '** FIELD OUTPUT: F-Output-1\n** \n*Output, field,
variable=PRESELECT\n** \n** HISTORY OUTPUT: H-Output-1\n** \n*Output,
history, variable=PRESELECT\n*End Step\n');
```

A.2 Result in '*.inp' file

The input file starts with the definition of the part and all the nodes of the generated mesh. It is followed by defining the elements obtained by the connection of the nodes. The later part is as follows:

```
** Section: Section-1
*Solid Section, elset=Volume, material="Trabecular bone"
,
*End Part
**
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Instance-1, part=Part
*End Instance
**
*Node
     1,   1.288700,   1.288700,   -0.026300
*Node
     2,   1.288700,   1.288700,   2.603700
*Nset, nset=Set-9
 1,
*Nset, nset=Set-10
 2,

%now, all the nodes that fall into the top and bottom surfaces in Z
direction are defined, which isn't included in the code below.

*Nset, nset=ref_point_1
 1,
*Nset, nset=ref_point_2
 2,
*Surface, type=NODE, name=z_bottom_CNS_, internal
z_bottom, 1.
*Surface, type=NODE, name=z_top_CNS_, internal
z_top, 1.
** Constraint: Bottom surface
*Coupling, constraint name="Bottom surface", ref node=ref_point_1,
surface=z_bottom_CNS_
*Kinematic
** Constraint: Top surface
*Coupling, constraint name="Top surface", ref node=ref_point_2,
surface=z_top_CNS_
*Kinematic
*End Assembly
**
** MATERIALS
**
*Material, name="Trabecular bone"
*Elastic
1629, 0.3
**
** BOUNDARY CONDITIONS
```

```
**
** Name: Fixed bottom surface Type: Displacement/Rotation
*Boundary
Set-9, 1, 1
Set-9, 2, 2
Set-9, 3, 3
Set-9, 4, 4
Set-9, 5, 5
Set-9, 6, 6
** ----------------------------------------------------------------
**
** STEP: Step-2
**
*Step, name=Step-2, nlgeom=YES, inc=1000000
*Static
0.1, 1., 1e-15, 0.1
**
** BOUNDARY CONDITIONS
**
** Name: Load Type: Displacement/Rotation
*Boundary
Set-10, 1, 1
Set-10, 2, 2
Set-10, 3, 3, 0.026300
**
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
*End Step
```

Appendix B

Bone samples chosen for analysis

Sample	Group	E	Damaged E	Damage	Plastic strain	Yield strain	Yield stress	BV/TV	TbTh	BMD	TBS
S1_G1%	1	1375.70	811.11	-	-	-	-	0.497	0.386	0.479	0.730
S2_G1%	1	2006.60	1371.96		0.05	0.68	9.63	0.408	0.256	0.415	0.641
S1_G2%	2	1535.80	1042.40	0.321	0.56	1.20	15.40	0.442	0.248	0.419	0.481
S2_G2%	2	1990.90	744.30		0.11	0.90	13.90	0.367	0.235	0.343	0.367
S1_G3.5%	3	2108.30	881.30	0.582	2.43	0.55	7.40	0.416	0.251	0.437	0.405
S2_G3.5%	3	1158.00	578.10	0.501	1.20	1.10	10.40	0.416	0.267	0.325	0.332
S3_G3.5%	3	1568.80	666.60	0.575	1.32	1.05	13.30	0.421	0.249	0.447	0.563
S4_G3.5%	3	1203.50	429.10	0.643	1.66	0.76	6.70	0.430	0.243	0.349	0.479
S5_G3.5%	3	1477.87	565.61	0.617	1.67	1.18	14.48	0.419	0.251	0.362	0.554
S6_G3.5%	3	1260.32	589.51	0.532	1.43	0.90	8.822	0.357	0.265	0.300	0.351
S1_G5%	4	1558.00	647.98	0.584	2.89	1.31	17.30	0.485	0.257	0.455	0.517
S2_G5%	4	2172.03	462.00	0.787	2.63	1.03	18.03	0.435	0.224	0.388	0.353
S3_G5%	4	1810.02	409.15	0.774	3.06	1.06	15.57	0.429	0.245	0.424	0.644
S4_G5%	4	1768.38	435.93	0.753	1.87	0.98	13.79	0.477	0.316	0.417	0.658
S5_G5%	4	815.22	285.07	0.650	2.42	1.14	7.663	0.408	0.241	0.343	0.273

Appendix C

Variation of number of elements with mesh size

S.No	Length (mm)	Mesh Size	Number of elements	Compression factor
1	13.17	167x167x167	928,914	3
2	13.15	83x83x83	117,181	6
3	10.49	133x133x133	498,287	3
4	13.15	250x250x250	3,322,560	2
5	10.52	200x200x200	1,804,314	2
6	5.26	100x100x100	365,600	2
7	2.63	50x50x50	49,854	2
8	2.10	40x40x40	26,089	2
9	1.84	35x35x35	17,320	2

Appendix D

Plasticity models

1. Concrete damage plasticity model

1.1 Defining compressive behaviour

The stress-strain behaviour can be defined in uniaxial compression outside the elastic range. Compressive stress data are provided as a tabular function of inelastic (or crushing) strain, $\tilde{\varepsilon}_c^{in}$. The stress-strain curve can also be defined beyond the ultimate stress point, into the strain-softening regime.

The post-yield data are given in terms of an inelastic strain, $\tilde{\varepsilon}_c^{in}$, instead of plastic strain, $\tilde{\varepsilon}_c^{pl}$. The compressive inelastic strain is obtained by subtracting the elastic strain from the total strain, $\tilde{\varepsilon}_c^{in} = \varepsilon_c - \varepsilon_{0c}^{el}$, where $\varepsilon_{0c}^{el} = \sigma_c/E_0$, as illustrated in Figure A.
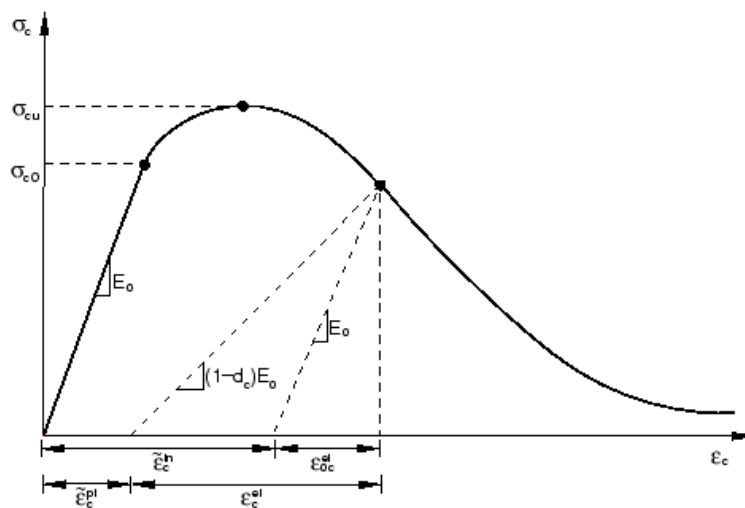

Figure A – Illustration of inelastic strain used for compression post-yield data

Unloading data are provided to Abaqus in terms of compressive damage curves, $d_c - \tilde{\varepsilon}_c^{in}$, as discussed below. The relationship between inelastic and plastic strain is given by,

$$\tilde{\varepsilon}_c^{pl} = \tilde{\varepsilon}_c^{in} - \frac{d_c}{(1 - d_c)} \frac{\sigma_c}{E_0}.$$

1.2 Defining damage

Damage d_c should be specified in tabular form. If damage isn't mentioned, the model behaves as a plasticity model. In Abaqus, the damage variables are treated as non-decreasing material point quantities. At a random increment during the analysis, updated value of each damage variable is obtained as the maximum between the value at the end of the previous increment and the value corresponding to the current state, which is obtained by interpolation of user-specified tabular data. It is given by

$$d_c|_{t+\Delta t} = \max \left\{ d_c|_t, d_c(\tilde{\varepsilon}_c^{pl}|_{t+\Delta t}, \theta|_{t+\Delta t}, f_i|_{t+\Delta t}) \right\}.$$

Where t and Δt are time and time increment respectively.

2. Crushable foam model

Firstly, volumetric hardening was chosen for this model owing to the fact that isotropic hardening isn't possible for a bone. This model uses a yield surface with an elliptical dependence of deviatory stress on pressure stress.

The yield surface for the volumetric hardening crushable foam model is defined as

$$F = \sqrt{q^2 + \alpha^2 (p - p_0)^2} - B = 0,$$

where

p is pressure stress,

q is Mises stress,

$S = \sigma + pI$ is the deviatory stress,

A is the size of the horizontal (p) axis of the yield ellipse,

$B = \alpha\ A = \alpha\ \frac{p_c + p_t}{2}$ is the size of the vertical (q) axis of the yield ellipse,

$\alpha = B/A$ is the shape factor of the yield ellipse that defines the relative magnitude of the axes,

$p_0 = \frac{p_c - p_t}{2}$ is the center of the yield ellipse on the p-axis,

p_t is the strength of the material in hydrostatic tension and
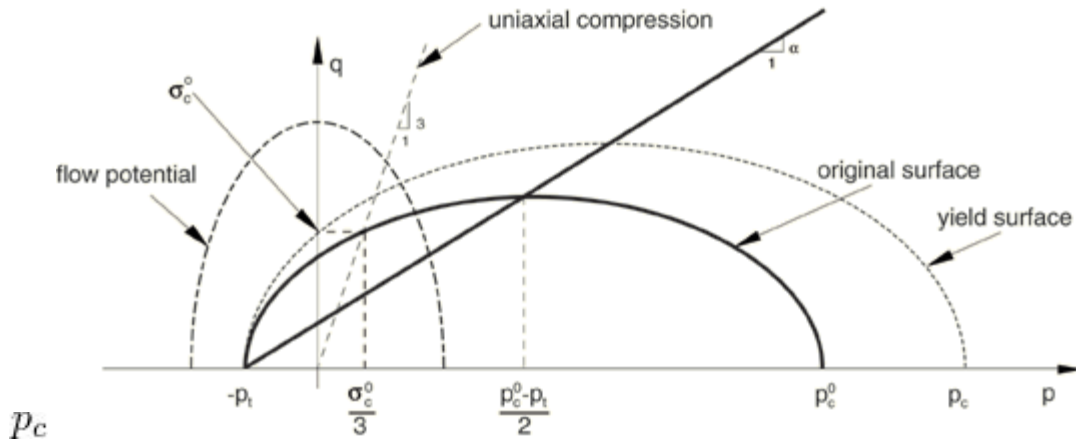
p_c is the yield stress in hydrostatic compression.

Figure B - Crushable foam model with volumetric hardening: yield surface and flow potential in the p–q stress plane.

The shape factor is calculated using the initial yield stress in uniaxial compression, σ_c^0, the initial yield stress in hydrostatic compression, p_c^0 (the initial value of p_c), and the yield strength in hydrostatic tension, p_t as shown below:

$$\alpha = \frac{3k}{\sqrt{(3k_t + k)(3 - k)}} \qquad \text{with} \qquad k = \frac{\sigma_c^0}{p_c^0} \quad \text{and} \quad k_t = \frac{p_t}{p_c^0}.$$

For a valid yield surface the choice of strength ratios must be such that $0 < k < 3$ and $k_t \geq 0$.

3. Classical plasticity model

Kinematic hardening was chosen to model the compressive loading with a constant rate of hardening. The evolution law of this model consists of a linear kinematic hardening component that describes the translation of the yield surface in stress space through the back stress, $\boldsymbol{\alpha}$. When temperature dependence is omitted, this evolution law is the linear Ziegler hardening law

$$\dot{\boldsymbol{\alpha}} = C \frac{1}{\sigma^0} (\boldsymbol{\sigma} - \boldsymbol{\alpha}) \dot{\bar{\varepsilon}}^{pl},$$

where $\dot{\bar{\varepsilon}}^{pl}$ is the equivalent plastic strain rate and C is the kinematic hardening modulus. In this model the equivalent stress defining the size of the yield surface, σ^0, remains constant, $\sigma^0 = \sigma|_0$, where $\sigma|_0$ is the equivalent stress defining the size of the yield surface at zero plastic strain.