Contents lists available at ScienceDirect

Journal of Biomechanics

journal homepage: www.elsevier.com/locate/jbiomech
www.JBiomech.com

Short communication

An open source software tool to assign the material properties of bone for ABAQUS finite element simulations

Elise C. Pegg*, Harinderjit S. Gill

Department of Mechanical Engineering, University of Bath, Bath, UK

ARTICLE INFO

Article history:
Accepted 28 July 2016

Keywords:
Material properties
Bone
Finite element
Python

ABSTRACT

A new software tool to assign the material properties of bone to an ABAQUS finite element mesh was created and compared with Bonemat, a similar tool originally designed to work with Ansys finite element models. Our software tool (py_bonemat_abaqus) was written in Python, which is the chosen scripting language for ABAQUS. The purpose of this study was to compare the software packages in terms of the material assignment calculation and processing speed. Three element types were compared (linear hexahedral (C3D8), linear tetrahedral (C3D4) and quadratic tetrahedral elements (C3D10)), both individually and as part of a mesh.

Comparisons were made using a CT scan of a hemi-pelvis as a test case. A small difference, of −0.05 kPa on average, was found between Bonemat version 3.1 (the current version) and our Python package. Errors were found in the previous release of Bonemat (version 3.0 downloaded from www.biomedtown.org) during calculation of the quadratic tetrahedron Jacobian, and conversion of the apparent density to modulus when integrating over the Young's modulus field. These issues caused up to 2 GPa error in the modulus assignment. For these reasons, we recommend users upgrade to the most recent release of Bonemat.

Processing speeds were assessed for the three different element types. Our Python package took significantly longer (110s on average) to perform the calculations compared with the Bonemat software (10s). Nevertheless, the workflow advantages of the package and added functionality makes 'py_bonemat_abaqus' a useful tool for ABAQUS users.

1. Introduction

Bone has heterogeneous material properties, and so in order to create representative finite element models it is essential that the properties are correctly described in order to draw useful results. It is possible to calculate the apparent density of bone from a Computed Tomography (CT) scan. Once the density is known, it can be used to calculate the Young's modulus value, typically using a power equation. Opinion varies as to which mathematical relationships are the most appropriate to use. Helgason et al. have published a comprehensive review on the area (Helgason et al., 2008) and it is not the purpose of this study to go into further detail on the subject.

What we are concerned with is the practical implementation of such mathematical relationships when using the finite element software ABAQUS (Simulia, Dassault Systèmes, Paris, France). It is possible to make such material assignments for an ABAQUS input file using commercial segmentation software, such as Simpleware (Simpleware Ltd., Exeter, UK) or Mimics (Materialise, Leuven, Belgium), but the cost of such software can be prohibitive. Researchers at the Istituto Ortopedico Rizzoli in Bologna, Italy, have created a software program called Bonemat to help with accurate material assignment for a finite element mesh. Bonemat is publically available and can be freely downloaded, making it a useful research tool. However, it has three main limitations:

1. Bonemat will only work on Microsoft Windows, which can be a problem for users of other operating systems.
2. Bonemat is not currently compatible with ABAQUS models, so users have to convert their finite element meshes into a compatible format, using an intermediate software such as Hypermesh (Altair Hyperworks, MI, USA) or custom scripting, before material properties can be assigned.
3. All model information (aside from the mesh) is lost after material assignment; such as, boundary conditions, element sets, or contact definitions. Redefining the element sets and model parameters can add significant time to a project involving multiple models.

* Corresponding author.
E-mail address: e.c.pegg@bath.ac.uk (E.C. Pegg).

http://dx.doi.org/10.1016/j.jbiomech.2016.07.037
0021-9290/© 2016 Published by Elsevier Ltd.

Table 1
Parameters used when assessing equivalence between material assignments of the software packages.

Parameter	Description	Set value
Gap Value	Modulus interval used when grouping the modulus values	0.01 GPa
Integration order	Order of the numerical integration used across the element	4
CT calibration coefficients [$\rho_{app}=a+b$ HU]	Calibration parameters a and b of the equation used to calculate the apparent density of bone (ρ_{app}) from the CT Hounsfield Unit (HU)	$a=-0.021075$, $b=0.000786$
Calibration correction	Option which allows correction of the calibration with up to three linear correlation.	Not applied
Modulus calculation parameters [$E=a+b\ \rho\ _{app}{}^{c}$]	Parameters a, b and c of the power equation used to convert the apparent density of bone to Young's modulus	$a=0$, $b=2.0173$, $c=2.46$
Minimum modulus value	Any modulus values below the minimum are changed to the minimum value	0.000001

The algorithm used by Bonemat to calculate the material properties of bone from a CT scan has evolved over the years. The first iteration (V1) calculated the average Hounsfield Unit (HU) value of all voxels found within an element; these were then converted to the modulus of bone to use for the element material (Zannoni et al., 1999). The second iteration (V2) used trilinear interpolation of the CT scan to more accurately estimate the HU value for each element node, and then used numerical integration to compute the HU for the element volume, which was then converted to modulus (Taddei et al., 2004). The third iteration (V3) first converted the CT scan voxels to modulus values, and then performed the linear interpolation and numerical integration to find the modulus for each element volume. Taddei et al. demonstrated V2 and V3 algorithms do result in different finite element model results. Strain calculated from models of a femur processed with V2 and V3 algorithms were compared to strains measured experimentally. V3 results had a higher regression coefficient with the experimentally measured strains (0.79) compared with V2 results (0.69) (Taddei et al., 2007).

We have created an open-source software which has been written in Python (Python Software Foundation, NH, USA) and aims to address the limitations of Bonemat for ABAQUS users. Python was an obvious choice for the program as it is the scripting language used by ABAQUS; furthermore Python works with all operating systems, so the package is multi-platform. It applies the same calculations as Bonemat, allows the same user options, but the mesh data input format is an ABAQUS input file. Furthermore, the program does not remove or change any model parameters already defined in the input file, such as boundary conditions, element sets, or contact definitions. The program also has the added functionality that it can cope with multiple parts, and can ‘ignore’ parts which are not bone and so do not need material assignment.

The aims of this study were to: (1) assess equivalence between material assignment for individual elements using Bonemat3.1 software (current version, Crimi, 2015), Bonemat3.0 software (previous version, Chiarini, 2006) and our Python package ('py_bonemat_abaqus') (2) assess equivalence for the modulus grouping and element assignment for a whole bone, and (3) compare the speed of the two software packages.

2. Methodology

Material assignment is applied to a finite element mesh in two stages: (1) the software calculates the modulus for each individual element, then (2) the modulus values for the entire mesh are grouped into bins to reduce computation time for the finite element model. Equivalence was assessed separately for these two stages of the material assignment. First, the numerical integration algorithm was checked on individual elements of controlled mesh quality. Second, modulus values were calculated for a CT scan of a hemi-pelvis using the different software packages; this enabled the grouping algorithms to be compared, and was representative of a typical study case. Lastly, the time taken for each software package to perform the calculations on the hemi-pelvis meshes was assessed.

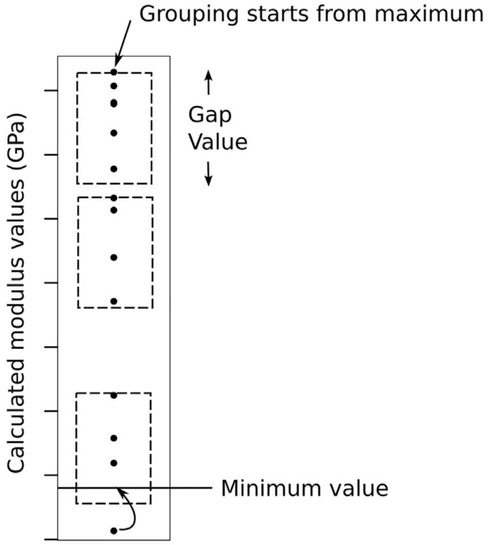

Fig. 1. Schematic illustration of the parameters detailed in Table 1 and how they influence the grouping of the modulus results.

2.1. Creation of the 'py_bonemat_abaqus' Python package

All scripts were written to be compatible with Python version 2.6 and 2.7 (Python Software Foundation, www.python.org) and were dependent on two other open-source packages: 'numpy' and 'pydicom'. The package can be installed from the Python Package Index using 'pip' or 'easy-install', or can be downloaded and manually installed using the setup.py script (Pegg, 2016). Once installed, the user can either run the script from the command line, or include the statement 'import py_bonemat_abaqus' in their python scripts, to process their ABAQUS input files.

2.2. Numerical integration accuracy and modulus assignment

The CT scan used was that of a pelvis downloaded from the VAKHUM database, created as part of a project funded by the European Commission under the Information Society Technologies Programme. The dataset was provided by the Laboratory of Human Anatomy and Embryology, University of Brussles (UBL), Belgium (Jan, 2005).

Ten single element input files were created with linear tetrahedral (C3D4), quadratic tetrahedral (C3D10), and linear hexahedral (C3D8) elements, and randomly assigned nodal co-ordinates within the CT scan volume. All elements had a Jacobian determinant greater than 0.2; this limit was recommended by Burkhart et al. (2013) for biomechanical studies of bone. The parameters used for the equivalence tests are summarised in Table 1, and described schematically in Fig. 1. The parameters used to convert HU to apparent density (ρ_{app}) (Eq. (1)) and from ρ_{app} to elastic modulus (Eq. (2)) were based on reported values for the pelvis (Anderson et al., 2005). Numerical integration across both the HU field (V2) and the modulus field (V3) were compared.

$$\rho_{app}=-0.021075+0.000786\ \text{HU} \tag{1}$$

$$E=2.0173\rho_{app}{}^{2.46} \tag{2}$$

The single element meshes were analysed using our ‘py_bonemat_abaqus’ Python package, Bonemat3.0 downloaded from the BiomedTown website (Chiarini, 2006), and Bonemat3.1 from www.bonemat.org (Crimi, 2015).