



# Bachelorarbeit

---

## Implementierung verschiedener Algorithmen zur automatisierten Berechnung und Zuweisung von Materialgesetzen von CT-Daten auf FE-Netze

---

vorgelegt von: Amadeus Cavalcanti Salvador de Alcântara  
Studiengang: Maschinenbau / **Mechatronik** / Physiktechnik  
Matrikelnummer: 21053  
Erstgutachter: Prof. Dr. -Ing. Wolf-Dietrich Knoll  
Zweitgutachter: M. Eng. Konrad Mehle

© 2017

Dieses Werk einschließlich seiner Teile ist **urheberrechtlich geschützt**. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Autors unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen sowie die Einspeicherung und Verarbeitung in elektronischen Systemen.

---

## Kurzfassung

Die Finite-Elemente-Methode (FEM) hat sich bereits seit langem im Bereich der Medizintechnik bzw. Biomechanik stark etabliert. Eine vollständige Beschreibung von Knocheneigenschaften und Spannungsfeldern in Knochenstrukturen kann mittels dreidimensionaler Simulationen durch FEM-Anwendung geliefert werden. Die Computertomographie (CT) gibt Auskunft über die Geometrie des Knochen. Aus der CT können auch Materialeigenschaften von Knochen ermittelt werden, die in computerunterstützte FE-Simulationen implementiert werden können. Wenn CT-Daten und FE-Netze präzise miteinander korrelieren, können numerisch-patientenspezifische Modelle erstellt werden, welche den Chirurgen eine Entscheidungshilfe bei der Wahl der idealen Rekonstruktionstechnik sein können. Das Ziel dieser Arbeit besteht darin, MATLAB-Algorithmen zu entwickeln, welche CT-Bilder und FE-Netze einlesen und sie miteinander in Verbindung setzen, sodass jedem Element des Netzes daraus abgeleitete Materialeigenschaften (Elastizitätsmodul, Dichte) anhand seiner jeweiligen Position zugewiesen werden; sprich Algorithmen, die eine Materialzuordnung bzw. Material-Mapping durchführen. Hierfür wurde eine grafische Benutzeroberfläche erstellt, die unterschiedlichen Benutzern (Medizinern, Ingenieuren, Studenten usw.) einen leichteren Umgang mit der MATLAB-Routine ermöglicht. Darüber hinaus wird der Einfluss der in dieser Arbeit vorgestellten Material-Mapping-Methoden untersucht. Zwei verschiedene Varianten werden hier präsentiert. Die erste Variante weist jedem Element die Materialeigenschaften der Voxeln (Volumenelement der CT) zu, deren Schwerpunkte innerhalb seiner Geometrie liegen. Die zweite Strategie weist jedem Element die aus der am nächsten am Element liegenden Voxel-Schwerpunkte abgeleiteten Materialeigenschaften zu. Beide in dieser Arbeit verwirklichten Strategien wurden miteinander und mit der Public-Domain-Software Bonemat verglichen. Die hier entwickelte MATLAB-Routine erzeugt eine APDL (Ansys Parametric Design Language) basierte Input-Datei, die sowohl in ANSYS-Classic als auch in ANSYS-Workbench bearbeitet werden kann. Somit können Knochen in Simulationen analysiert werden, was z.B. Prognosen und Evaluationen von Frakturrisiken erleichtert.

---

## Abstract

The Finite Element Method (FEM) has long been a key procedure in orthopedic biomechanics. Since 1972, this method has been applied with increasing frequency for stress analysis of bone and bone-prosthesis structures, fracture fixation devices, and various kinds of tissues other than bone. Computed Tomography (CT) data provide three dimensional bone information and are thus commonly used to generate FE models of bone. When properly calibrated, it is possible to calculate a bone material density from CT images. Hence, an accurate correlation of CT data and FE mesh is capable of providing meaningful and trustful models that could give assistance to surgeons in choosing the ideal reconstruction techniques for each specific patient case. The aim of the present work is to develop a © MATLAB program able to read a CT data set as well as a Finite Element (FE) mesh generated from it in © ANSYS Workbench, and to assign to each element of the mesh material properties, more specifically a Young's Modulus, based on the bone material density at the element position. This work also aims at the creation of a Graphical User Interface (GUI), which allows users from different fields (radiology, medicine, engineering, students, etc.) to easily interact with the program and to understand what is going on in the program. Furthermore, the influence of the method used to assign a Young's Modulus to an element has been evaluated. Two different methods were proposed. Method one assigns each element the material properties of the voxel (volume element of CT data) centroids lying directly inside the element. The second proposal or method two assigns an element the material properties of the closest to its centroid and to its vertices voxel centroids. Both methods have been tested, analysed and compared. Besides, a comparison with the software available in public domain © Bonemat has been run. The developed MATLAB program generates an APDL (Ansys Parametric Design Language) based data providing all material informations, that can be opened in ANSYS Classic for further simulations and medical analysis.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>III</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Zielstellung . . . . .	2
1.2. Aufbau der Arbeit . . . . .	2
<b>2. Grundlagen</b>	<b>4</b>
2.1. Computertomographie . . . . .	4
2.2. Schwächung von Röntgenstrahlen . . . . .	5
2.3. Hounsfield-Skala . . . . .	8
2.4. DICOM . . . . .	9
2.5. Knochendichtemessung . . . . .	12
2.6. Finite-Elemente-Methode . . . . .	13
<b>3. Die MATLAB-Routine</b>	<b>15</b>
3.1. Importieren . . . . .	17
3.1.1. Einlesen der CT-Daten . . . . .	17
3.1.2. Einlesen des FE-Netzes . . . . .	20
3.2. Kalibrierung der CT-Daten . . . . .	21
3.3. Material-Mapping . . . . .	26
3.3.1. Variante 1 - Convex Hull . . . . .	26
3.3.2. Variante 2 - Nearest Point Search . . . . .	34
3.4. Ableitung der elastischen Materialeigenschaften anhand der Knochen- dichte . . . . .	35
3.5. Komponentenerzeugung . . . . .	36
3.6. Exportieren . . . . .	39
<b>4. Material und Methoden</b>	<b>44</b>
<b>5. Ergebnisse und Diskussion</b>	<b>47</b>
5.1. Ergebnisse . . . . .	47
5.1.1. Os ilium links . . . . .	48
5.1.2. Os ilium rechts . . . . .	50
5.1.3. Os sacrum . . . . .	52
5.1.4. L5 . . . . .	54
5.2. Diskussion . . . . .	56
5.2.1. Betrachtung der Rechenzeit . . . . .	60
<b>6. Zusammenfassung und Ausblick</b>	<b>63</b>

<b>7. Danksagung</b>	<b>65</b>
<b>Literaturverzeichnis</b>	<b>67</b>
<b>Eidesstattliche Erklärung</b>	<b>71</b>
<b>A. Anhang</b>	<b>i</b>
A.1. Anleitung zur Erstellung und Speicherung eines FE-Netzes . . . . .	<b>i</b>

# Abbildungsverzeichnis

1.1. Prinzipielles Vorgehen beim CT-basierten Material-Mapping . . . . .	3
2.1. Grundprinzip der Computertomographie. Eine komplette Rotation des Röntgenstrahlers um den Körper entspricht einer Pixel-Matrix auf dieser Ebene [1]. . . . .	4
2.2. Absorber mit großer Kernladungszahl . . . . .	6
2.3. Absorber mit niedriger Kernladungszahl . . . . .	6
2.4. Absorber mit großer Dichte . . . . .	6
2.5. Allgemeines Schwächungsgesetz [2]. Eine Strahlung der Intensität $I$ trifft auf eine Materialschicht der Dicke $dx$ . Die Intensität nimmt um $dI$ ab. . . . .	7
2.6. Exponentielle Abschwächung von Strahlung in Abhängigkeit des linearen Schwächungskoeffizienten . . . . .	8
2.7. Hounsfield-Skala für verschiedene Gewebetypen. © Institut für Anatomie, Universität Bern . . . . .	9
2.8. Gruppierung von DICOM Objekten . . . . .	11
3.1. Die grafische Benutzeroberfläche von <i>MatMap</i> . . . . .	16
3.2. DICOMDIR-Informationsliste . . . . .	17
3.3. Bild Pixel-Matrix im MATLAB . . . . .	18
3.4. Darstellung des Pixelabstandes und der Pixel-Matrixkoordinate . . . . .	19
3.5. Darstellung eines CT-Bildstapels . . . . .	19
3.6. Berechnung der Pixelkoordinaten [3] . . . . .	20
3.7. Solid186 (© ANSYS-HELP) . . . . .	21
3.8. Beispiel eines Phantoms . . . . .	22
3.9. Beispiel eines Kreises um den Rand des Knochenmineralstandards . . . . .	23
3.10. maskierte Matrix . . . . .	24
3.11. Bestimmung des mittleren Pixels . . . . .	24
3.12. Erzeugung eines Rechtecks um das mittlere Pixel herum . . . . .	25
3.13. ermitteltes Rechteck in der nicht maskierten Bild-Matrix . . . . .	25
3.14. ermittelte lineare Kurvenanpassung des $HU-\rho_{QCT}$ -Zusammenhangs . . . . .	26
3.15. (a) endliche Menge von Punkten. (b) konvexe Hülle der Punktmenge . . . . .	27
3.16. Mit der Funktion <i>inhull</i> wird überprüft, ob sich ein Punkt innerhalb oder außerhalb einer konvexen Hülle befindet . . . . .	28
3.17. Betrachtung eines Tetraeders im 2D-Raum . . . . .	28
3.18. Rahmen der Begrenzung, wobei $s$ und $r$ die Hälfte der PixelSpacing jeweils für die $x$ und $y$ Richtung sind. Bei Voxel-Schwerpunkten, die sich außerhalb der $xyz$ -Begrenzung befinden, wird nicht überprüft, ob sie sich innerhalb der konvexen Hülle der Elementgeometrie befinden . . . . .	29

3.19. Elemente und die entsprechenden, innerhalb der xyz-Begrenzung liegenden Voxel-Schwerpunkte . . . . .	29
3.20. Beispiel eines Elements mit und ohne Voxel-Schwerpunkte innerhalb seiner konvexen Hülle (links bzw. rechts) . . . . .	32
3.21. Aufteilung der 1. und 2. Ordnung eines Voxels . . . . .	32
3.22. Vorgehensweise der Variante 1 . . . . .	33
3.23. Element am Rand des Knochens liegt zwischen großen und niedrigen HU-Werte . . . . .	34
3.24. V1 vs V2 - Unterschied der betrachteten Voxel-Schwerpunkte . . . . .	35
3.25. Erzeugung von Knoten in ANSYS-Mechanical-APDL . . . . .	40
3.26. Erzeugung von Elementen in ANSYS-Mechanical-APDL . . . . .	40
3.27. Definition von Materialeigenschaften in ANSYS-Mechanical-APDL . . . . .	41
3.28. Zuweisung von Elementen zu seiner Komponente . . . . .	41
3.29. Zuweisung von Materialeigenschaften zur Komponente . . . . .	42
3.30. <i>MatMaps</i> Ablaufplan . . . . .	43
4.1. Beckenknochen [4] . . . . .	44
4.2. Beispiel einer ELIST . . . . .	45
4.3. Beispiel einer MPLIST . . . . .	46
5.1. E-Modul vs Komponente-ID - Os ilium links . . . . .	48
5.2. Elementanzahl vs Komponente-ID - Os ilium links . . . . .	49
5.3. E-Modul vs Element-ID für die ersten 15 Elemente- Os ilium links . . . . .	49
5.4. E-Modul vs Komponente-ID - Os ilium rechts . . . . .	50
5.5. Elementanzahl vs Komponente-ID - Os ilium rechts . . . . .	51
5.6. E-Modul vs Element-ID für die ersten 15 Elemente- Os ilium rechts . . . . .	51
5.7. E-Modul vs Komponente-ID - Os sacrum . . . . .	52
5.8. Elementanzahl vs Komponente-ID - Os sacrum . . . . .	53
5.9. E-Modul vs Element-ID für die ersten 15 Elemente- Os sacrum . . . . .	53
5.10. E-Modul vs Komponente-ID - L5 . . . . .	54
5.11. Elementanzahl vs Komponente-ID - L5 . . . . .	55
5.12. E-Modul vs Element-ID für die ersten 15 Elemente- L5 . . . . .	55
5.13. Importieren des kompletten Beckens und des Wirbelkörpers in ANSYS-Classic . . . . .	58
5.14. Becken in ANSYS-Workbench . . . . .	59
5.15. Materialzuordnung in ANSYS-Workbench . . . . .	59
5.16. Trennung der Knochenmaterialien über die Funktion PNUMMAT in ANSYS-Workbench . . . . .	60
A.1. ANSYS-Workbench Modell . . . . .	i
A.2. Speicherung einer cdb-Datei bzw. E- und NLIST . . . . .	ii

# 1. Einleitung

Seit vielen Jahren steigt die durchschnittliche Lebenserwartung sowohl in den Industrieländern als auch in den Entwicklungsländern in einer bemerkenswerten Größenordnung [5]. Doch bedeutet ein längeres Leben nicht unbedingt ein gesundes Leben. Neben vielen anderen Erkrankungen nimmt die Anzahl der Fälle von Osteoporose bei der älteren Bevölkerung deutlich zu [6].

Osteoporose ist die häufigste generalisierte altersabhängige Knochenerkrankung, die durch eine niedrige Knochenmasse und eine mikroarchitektonische Verschlechterung des Knochengewebes charakterisiert wird, was zu einer Steigerung des Frakturrisikos und der Knochenfragilität führt [7; 8]. Dies ist eine wachsende gesundheitsassoziierte und sozioökonomische Gefahr und ist auch die Hauptbehinderungs- und -todesursache unter älteren Frauen weißer Hautfarbe, bei denen sie am häufigsten auftritt [9]. Aufgrund steigender Lebenserwartung lässt sich vermuten, dass die Fallzahlen von Knochenfrakturen in den kommenden Jahren zunehmen werden. Um diese Prognose zu vermeiden und um den Menschen ein gesundes Leben im höheren Alter zu gewährleisten, wird eine immer engere Kooperation zwischen Medizinern und Ingenieuren erforderlich [10].

Bildgebende Verfahren der Medizin, wie z.B. die Computertomographie (CT), ermöglichen Medizinern und Wissenschaftlern eine Untersuchung der inneren Strukturen des Körpers, ohne diesen dabei aufschneiden zu müssen. Dies kann zum Beispiel die Arbeit eines orthopädischen Chirurgen erleichtern und ihm eine Entscheidungshilfe bei der Wahl der idealen Rekonstruktionstechnik geben. Durch die CT können z.B. auch 3D-Informationen und Materialeigenschaften von Knochen ermittelt werden, die in computerunterstützte FE-Simulationen implementiert werden können.

Die Finite-Elemente-Methode (FEM) hat sich in den letzten Jahren im Bereich der Medizintechnik, insbesondere bei der Analyse des mechanischen Verhaltens von biologischen Strukturen d.h. Biomechanik, stark etabliert. Eine vollständige Beschreibung von Knocheneigenschaften und Spannungsfeldern in Knochenstrukturen kann mittels dreidimensionaler Simulationen durch FEM-Anwendung geliefert werden. Solche Informationen sind ein entscheidender Faktor für das Verständnis des Verhaltens der Knochenstruktur in der Forschung und der klinischen Anwendung [11; 12].

## 1.1. Zielstellung

Die vorliegende Arbeit dient der Erstellung numerischer-patientenspezifischer Modelle, welche dem Operateur eine Entscheidungshilfe bei der Wahl der idealen Rekonstruktionstechnik geben können.

Das konkrete Ziel dieser Arbeit war es, Matlab-Algorithmen zu entwickeln, welche CT-Bilder und FE-Netze einlesen und sie miteinander in Verbindung setzen. Dies ermöglicht eine Materialzuweisung bzw. -zuordnung. Sprich, es wird jedem Element des FE-Netzes ein HU-Wert bzw. daraus abgeleitete Materialeigenschaften (Elastizitätsmodul, Knochendichte) anhand seiner jeweiligen Position zugewiesen. Nach der Materialzuordnung erzeugt das Programm eine APDL (Ansys Parametric Design Language) basierte Input-Datei, die in ANSYS geöffnet werden kann. Somit können Knochen in Simulationen analysiert werden, was z.B. Prognosen und Evaluationen von Frakturrisiken erleichtert. Die Aufgabenstellung dieser Arbeit gliedert sich dabei in folgende Schwerpunkte:

1. Aufbau einer GUI-basierten Matlab-Routine zum Einlesen und Analysieren von CT- Daten (DICOM-Files) und FE- Netzen (Ansys-Files)

Das Ziel der grafischen Benutzeroberfläche (GUI) besteht darin, dem Benutzer (egal ob Operateur, Arzt oder Ingenieur) den Umgang mit dem Programm zu erleichtern.

2. Recherche zu Korrelationen zwischen Röntgenabschwächung und elastischen Konstanten von Knochen

3. Implementierung einer Mapping-Strategie für die Zuordnung der Voxel-HU-Werte auf ein FE-Mesh und Vergleich der Ergebnisse mit herkömmlichen Softwarelösungen (Bonemat)

Der Vergleich der Ergebnisse wurden anhand Bonemat [13], einer am Istituto Ortopedico Rizzoli in Bologna entwickelten Software, durchgeführt.

4. Export einer Mapped- Mesh- Datei auf APDL-Basis

## 1.2. Aufbau der Arbeit

Die Ausgangssituation bzw. -informationen des im Rahmen dieser Arbeit entwickelten Programms waren CT- Daten und Step-Dateien des entsprechenden CT Bildsatzes. CT-Daten wurden im BG Klinikum Bergmannstrost Halle aufgenommen. Die Step-Dateien sind aus diesen CT-Daten mit Hilfe der Software AMIRA

und CATIA erstellt worden. Da deren Erstellung nicht Teil dieser Arbeit war, wird sie hier nicht weiter beschrieben. Die Step-Dateien (.stp; also eine 3D Geometrie) wurden in ANSYS-Workbench importiert, woraus sich mit verschiedenen Strategien (Größe, Elementtyp) FE-Vernetzungen erstellen ließen. Das FE-Netz wurde in ANSYS-APDL konvertiert und die Informationen (Knotenkoordinaten und Elementzuordnung) anhand von ELIST und NLIST abgespeichert.

Das für diese Arbeit erstellte Programm bzw. die MATLAB-Routine soll CT-Daten und das zugehörige FE-Netz importieren und daraus eine auf ANSYS-APDL basierte Datei erzeugen, die eine Materialzuordnung der Elemente beinhaltet und für spätere Simulationen angewendet werden kann.

Abb. 1.1 stellt schematisch den Prozess der Materialzuordnung (Material-Mapping) von der CT-Aufnahme im Klinikum bis zur Erstellung der Input-Datei für ANSYS-APDL dar.

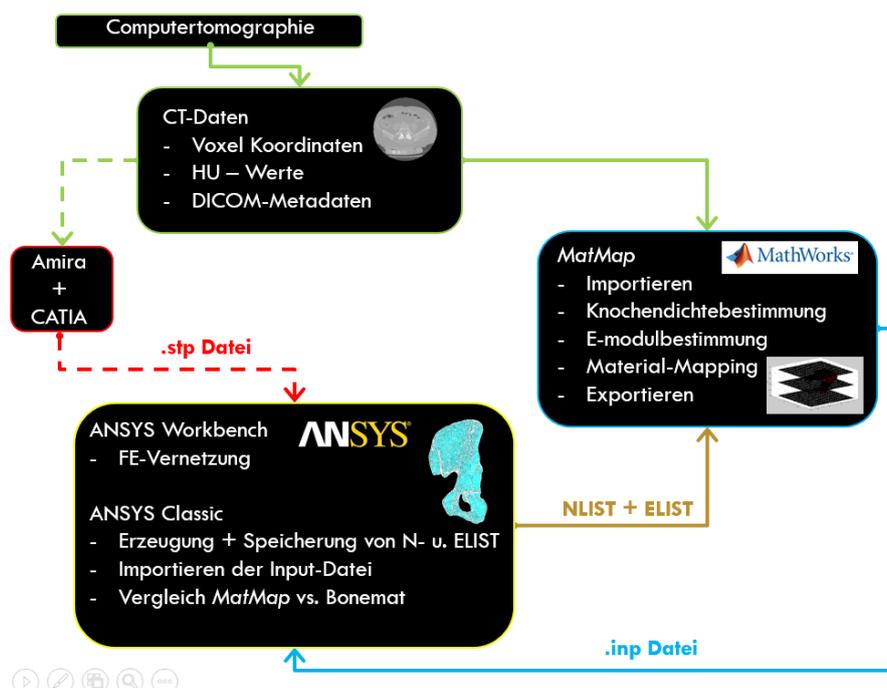


Abbildung 1.1.: Prinzipielles Vorgehen beim CT-basierten Material-Mapping

Die gestrichelten Linien symbolisieren die Schritte, die nicht von dem Autor gemacht wurden. Sie waren nicht Thema dieser Arbeit. Für diese Arbeit wurden folgende Softwarepakete genutzt:

- MATLAB Simulink 2014a
- ANSYS-Workbench und ANSYS-APDL 17.0
- Bonemat v3.2

## 2. Grundlagen

Zum besseren Verständnis dieser Arbeit werden hier theoretische Grundlagen und häufig vorkommende Begriffe, Verfahren und Informationen näher erläutert und erklärt.

### 2.1. Computertomographie

Die Computertomografie (CT) ist eines der wichtigsten bildgebenden Verfahren zur Darstellung von inneren Körperstrukturen in der Medizin. Mittels Röntgenstrahlung (siehe Unterabschnitt 2.2) ermöglicht die CT eine Untersuchung der inneren Struktur eines Körpers, ohne diesen dabei aufschneiden zu müssen [14]. Das von der Computertomographie gelieferte Bild (auch Röntgenbild genannt) ist im Prinzip die Projektion von einem dreidimensionalen Gebilde auf eine Ebene. Wenn mehrere solcher Bilder aufeinander gelegt werden, kann ein drei dimensionales Model des ursprünglichen Objekts erstellt werden ( z.B. eines Knochens).

Das bildgebende Gerät der Computertomographie heißt Computertomograph. Er besteht aus Röntgenröhre und Empfangsdetektoren, welche sich sehr schnell um den Körper des Patienten drehen, während dieser auf einer Liege langsam durch das ringförmige Gehäuse des Gerätes geschoben wird (siehe Abb. 2.1). Die Strahlung durchdringt die Materie, in dem Fall den Körper des Patienten, und wird entweder ganz oder teilweise absorbiert.

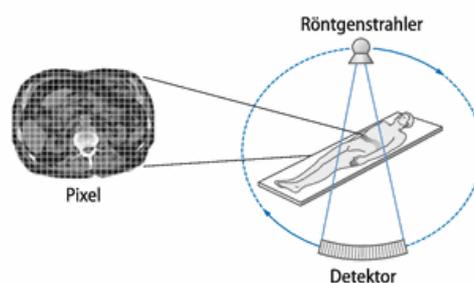


Abbildung 2.1.: Grundprinzip der Computertomographie. Eine komplette Rotation des Röntgenstrahlers um den Körper entspricht einer Pixel-Matrix auf dieser Ebene [1].

Die Empfangssensoren gegenüber dem Röntgenstrahler empfangen die unterschiedlich starken Signale und leiten sie an einen Computer weiter, der aus den empfangenen Daten Berechnungen durchführt und Schichtbilder des Körpers zusammensetzt. Diese Schichtbilder sind eigentlich Röntgen-Schwächungskoeffizienten, die innerhalb

einer durchstrahlten Schicht des Patienten aufgenommen und in einer Matrix abgebildet werden. Jedem Volumenelement (dem sog. Voxel) einer solchen Matrix wird ein über sich selbst gemittelter linearer Schwächungskoeffizient in Hounsfield-Einheit zugeordnet. Dieser Koeffizient wird dann in einen Grauwert übersetzt, der im betreffenden Pixel (Bildelement, ein zweidimensionales Voxel) des zwei-dimensionalen CT-Bildes dargestellt wird, ein CT-Bild besteht normalerweise aus  $512 \times 512$  gleich großen Pixeln. Je dichter ein Material ist, desto mehr Strahlung wird absorbiert und desto weißer wird das zugehörige Röntgenbild bzw. der Pixel. Da es einen großen Unterschied zwischen der Röntgenabwächung von Knochen und den daneben liegenden Geweben bzw. Organen gibt, können Knochen in Röntgenbildern sehr gut erkannt werden. Herz, Lunge, Gehirn und viele weitere Strukturen lassen sich auch sehr detailliert abbilden. Deshalb und aus anderen Gründen findet die Computertomographie heute in vielen Bereichen der Medizin eine breite Anwendung und ist aus der klinischen Routine nicht mehr wegzudenken. Sie gibt es bereits seit Mitte der 1970er Jahre und sie hat seitdem die bildgebenden Verfahren gewissermaßen revolutioniert. Die vielfältigen, von der CT gelieferten Informationen sind sehr hilfreich um Mehrdeutigkeiten auszuschließen und sind auch essentiell für die Therapieentscheidung.

### 2.2. Schwächung von Röntgenstrahlen

Wenn ein Röntgenstrahl versucht durch ein Material hindurchzutreten, verliert er ganz oder teilweise seine Intensität bzw. Energie, er wird also entweder ganz oder teilweise absorbiert [1]. Es gibt hauptsächlich drei Größen, die Einfluss auf diesen Verlust haben: Die Kernladungszahl, die Dichte und die Dicke des Materials.

Atome mit größerer Kernladungszahl sind für die Strahlung einfacher zu „treffen“ als Atome mit einer niedrigeren Kernladungszahl und absorbieren daher mehr Energie. In Analogie kann man sich eine Strömung von verschmutztem Wasser durch einen Filter vorstellen, wobei das verschmutzte Wasser die Strahlung und die Feinheit des Filters die Größe der Kernladungszahl darstellt. Atome mit größerer Kernladungszahl verhalten sich wie ein feiner Filter und lassen kaum Verschmutzungen bzw. Strahlung hindurch (siehe Abb. 2.2).

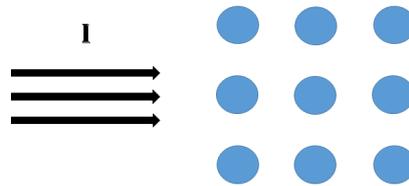


Abbildung 2.2.: Absorber mit großer Kernladungszahl

Atome mit einer niedrigeren Kernladungszahl hingegen nehmen im Raum weniger Platz ein und lassen somit die Strahlung leichter durchgehen. Genauso wie ein grober Filter (siehe Abb. 2.3).

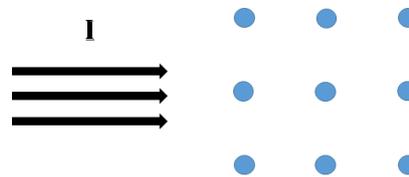


Abbildung 2.3.: Absorber mit niedriger Kernladungszahl

Die selbe Analogie kann auch für die Dichte angewendet werden, wobei die Dichte der Größe der Löcher des Filters entspricht. Je dichter das Material, desto kleinere Löcher besitzt der Filter und umso schwieriger ist es für die Strahlung bzw. Verschmutzung durchzuströmen (siehe Abb. 2.4).

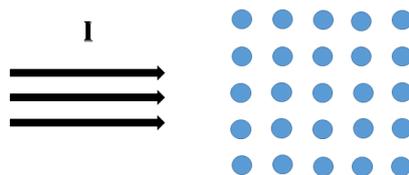


Abbildung 2.4.: Absorber mit großer Dichte

Die gleiche Analogie gilt auch für die Dicke des Materials: Je dicker das Material desto schwächer wird die Intensität der Strahlung am Ende. Nachfolgend wird das mathematische Modell eines Absorptionsexperiments, indem nur die Dicke des Absorbers als Variable betrachtet wird, dargestellt.

Abb. 2.5 zeigt einen parallelen Röntgenstrahl der Intensität  $I$ , der durch eine Folie der Dicke  $dx$  eines Materials hindurch tritt.

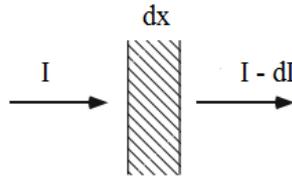


Abbildung 2.5.: Allgemeines Schwächungsgesetz [2]. Eine Strahlung der Intensität  $I$  trifft auf eine Materialschicht der Dicke  $dx$ . Die Intensität nimmt um  $dI$  ab.

Die verlorene Intensität bzw. Energie  $-dI$  ist proportional zur Dicke der Materialschicht und zur eingestrahnten Intensität:

$$-dI \propto I \cdot dx \quad (2.1)$$

Das Minuszeichen deutet an, dass die Intensität durch den Absorber verringert wird. Wird die proportionale Beziehung in eine Gleichung umgewandelt, ergibt sich:

$$-dI = \mu \cdot I \cdot dx \quad (2.2)$$

,wobei die materialabhängige Proportionalitätskonstante  $\mu$  als linearer Schwächungskoeffizient bezeichnet wird (auf englisch *attenuation coefficient*) und abhängig von der Dichte und von der Kernladungszahl des Absorbers ist. Diese Gleichung beschreibt den Effekt für kleine Änderungen in der Absorberdicke  $dx$ . Um herauszufinden, was sich für die gesamte Länge des Absorbers ergibt, müssen die Effekte für alle kleinen Teilstücke zusammengerechnet werden: Sprich die Gleichung muss integriert werden. Wird dies formaler ausgedrückt, so lässt es sich feststellen, dass die Intensität der Strahlung von der Dicke  $x=0$  bis zu einer endlichen Dicke  $x$  von  $I_0$  auf  $I$  abnehmen wird, so dass gilt:

$$-\int_{I_0}^I \frac{dI}{I} = \int_0^x \mu \cdot dx \quad (2.3)$$

$$\log I - \log I_0 = \log\left(\frac{I}{I_0}\right) = -\mu \cdot x \quad (2.4)$$

$$I = I_0 \cdot e^{-\mu \cdot x} \quad (2.5)$$

Diese letzte Gleichung verdeutlicht, dass die Intensität der Strahlung exponentiell mit der Dicke der getroffenen Materialschicht abnimmt, wobei die Abnahmerate durch den linearen Schwächungskoeffizienten gegeben ist. Abb. 2.6 stellt sowohl diese Beziehung als auch den Einfluss des linearen Schwächungskoeffizienten graphisch dar. Es ist die Intensität  $I$  gegenüber der Dicke des Absorbers  $x$  aufgetragen.

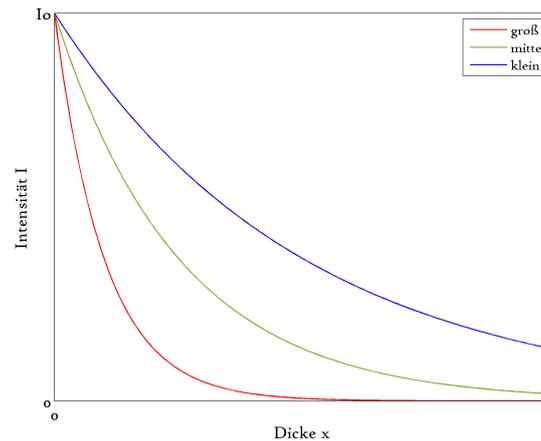


Abbildung 2.6.: Exponentielle Abschwächung von Strahlung in Abhängigkeit des linearen Schwächungskoeffizienten

Die Intensität vom Wert  $I_0$  bei  $x=0$  nimmt zunächst sehr schnell und später immer langsamer in klassischer exponentieller Form ab. Alle drei Kurven verlaufen exponentiell, nur die linearen Schwächungskoeffizienten sind verschieden. Bei einem kleinen linearen Schwächungskoeffizienten sinkt die Kurve relativ langsam und bei einem großen linearen Schwächungskoeffizienten sinkt sie ziemlich schnell.

Die Energie der Gamma-Strahlung kann die Abschwächung von einer Strahlung auch beeinflussen. Strahlungen mit höherer Energie werden weniger stark abgeschwächt. Es ist auch hier wichtig zu bemerken, dass die obige Ableitung nur für dünne Strahlenbündel strikt gültig ist. Bei ausgedehnten Strahlen müssen weitere Faktoren berücksichtigt werden.

### 2.3. Hounsfield-Skala

Wie schon vorher erwähnt, wird in einem CT-Bild der lineare Schwächungskoeffizient gemittelt über jedes Volumenelement in Hounsfield-Einheiten (HE) (auf englisch Hounsfield units (HU)) dargestellt. Die Hounsfield-Einheit ist im Grunde genommen eine Umrechnung des ermittelten Röntgen-Schwächungskoeffizienten die beim Treffen von Wahrscheinlichkeitsaussagen von angezeigten CT-Zahlen auf einem CT-Bild

## 2. Grundlagen

helfen kann, Gewebe zu identifizieren. Mit dem Schwächungskoeffizienten des betrachteten Gewebes  $\mu_{\text{Gewebe}}$  und von Wasser  $\mu_{\text{Wasser}}$  wird der CT-Wert definiert als:

$$[CT - Wert](\mu_{\text{GEWEBE}}) = \frac{\mu_{\text{GEWEBE}} - \mu_{\text{WASSER}}}{\mu_{\text{WASSER}}} \cdot 1000HU \quad (2.6)$$

In der Praxis hat sich der Werte-Bereich der Hounsfield-Skala von -1024 bis +3071 HU durchgesetzt (entspricht 4096 Graustufen). Da die Luft nahezu keine Röntgenstrahlung absorbiert und definitionsgemäß einen HU-Wert von -1000 besitzt, erreicht die Hounsfield-Skala keinen viel kleineren Wert. Nach oben hin ist die Skala theoretisch offen. Metalle z.B. können eine Absorption bis hin zur Totalabsorption (die sich nicht mehr darstellen lässt) bewirken. Verschiedene Gewebetypen weisen unterschiedliche HU-Werte auf (siehe Abb. 2.7).

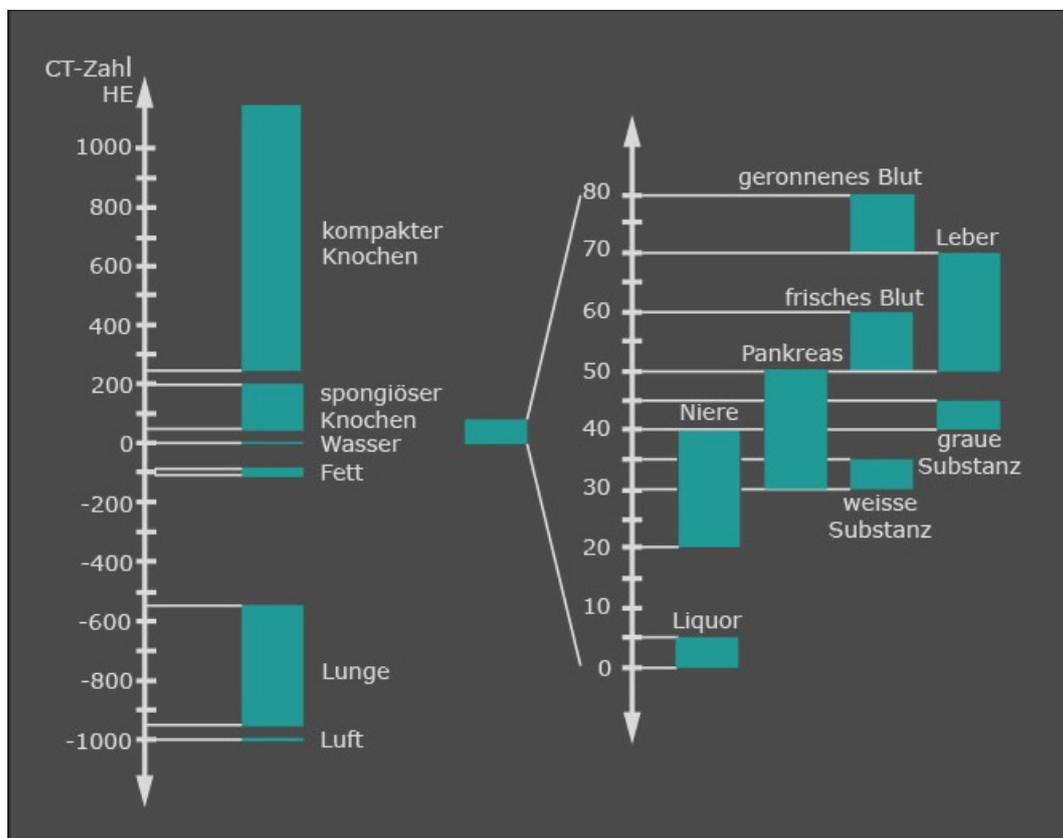


Abbildung 2.7.: Hounsfield-Skala für verschiedene Gewebetypen. © Institut für Anatomie, Universität Bern

## 2.4. DICOM

DICOM (Digital Imaging and Communications in Medicine; deutsch Digitale Bildgebung und -kommunikation in der Medizin) ist ein weltweiter offener Standard zum

Austausch von digitalen Bildern in der Medizin. DICOM standardisiert sowohl das Format zur Speicherung von Bilddaten, als auch das Kommunikationsprotokoll zum Austausch der Bilder.

DICOM ist ein offener Standard, der die Interoperabilität von medizinischen Systemen herstellerübergreifend gewährleistet und wird gerne in Krankenhäusern verwendet. Alle DICOM konformen Medizinprodukte müssen ein Conformance-Statement nachweisen, das sicherstellt, dass alle DICOM Produkte untereinander kompatibel sind. DICOM ist somit die Schnittstelle zwischen den bildgebenden Geräten (CT, Röntgen, MRT, usw.) und den anderen Objekten die mit dem DICOM-Netzwerk verbunden sind (Befundungsarbeitsplatz, Drucker, PACS usw.). Das Krankenhauspersonal ist somit in der Lage, von verschiedenen Orten (z.B. Behandlungszimmer, Station oder Büro) aus über verschiedenste DICOM-fähige Geräte auf die gewünschten Daten zuzugreifen.

Der Kern von DICOM ist ein Dateiformat- und ein Kommunikationsprotokoll. Sie werden im Folgenden näher beschrieben.

**DICOM File Format (Dateiformat)** – Alle medizinischen Bilder werden im DICOM Format gespeichert. Ärzte und Mediziner benutzen DICOM Viewer und Computersoftware-Anwendungen, um die DICOM Bilder zu visualisieren bzw. zur Erstellung von Diagnosen. DICOM Dateien beinhalten viel mehr als nur Bilder. Jede DICOM Datei enthält Patienteninformationen (z.B. Name, Geschlecht, Geburtsdatum, Versichertennummer, Status, usw.), wichtige Daten über die Aufnahme der Bilder (z.B. Geräteart und -einstellungen) und den Kontext der Studie zur Tomographie bzw. den bildgebenden Untersuchungen, der einen Zusammenhang zwischen dem Bild und seiner entsprechenden medizinischen Behandlung setzt. Diese Informationen werden in einer bestimmten Hierarchie abgelegt: Patient, Studie, Serie, Bild. Abb. 2.8 verdeutlicht diesen Zusammenhang.

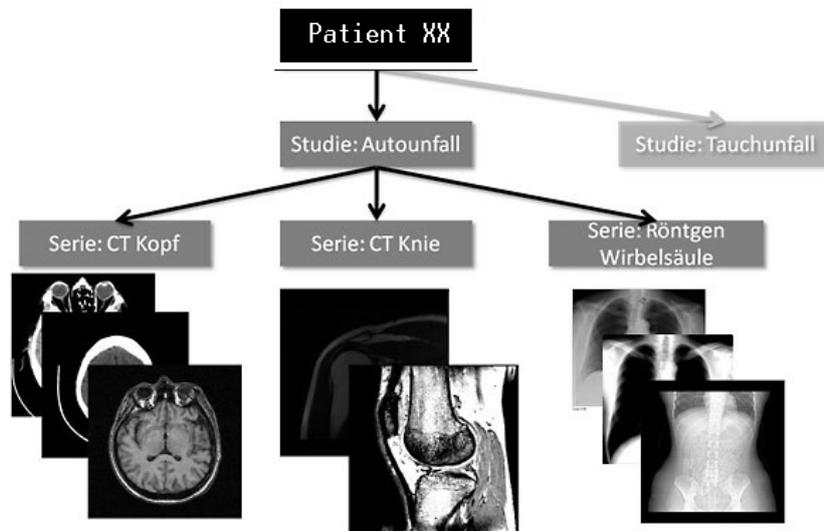


Abbildung 2.8.: Gruppierung von DICOM Objekten

Der Patient ist die Basis der Hierarchie und nimmt meist an einer Studie teil, z.B. im Kontext Autounfall. Die Studie enthält oft eine ID und einen Namen (hier beispielsweise „Autounfall“). Wenn er dann noch einmal eingeliefert wird, kann eine neue Studie angelegt werden. In der Praxis wird häufig allerdings bei jedem Fall ein neuer Patient mit eigener Studie angelegt. In einer Studie gibt es mehrere Geräte mit denen der Patienten untersucht wird. Die Daten der Geräte werden in sogenannten Serien abgelegt wie z.B. die Serie eines CTs, oder Ultraschalls. Neue Serien werden begonnen bei:

- jeder Modalität (Ultraschall, CT, usw.)
- jeder Aufnahmeposition: Körperstelle, Gantry-Winkel bei CT, usw.
- unterschiedlichen Aufnahmeparametern: mit/ohne Kontrastmittel, T1 oder T2 Gewichtung bei Kernspin, usw.
- verschiedenen Zeitpunkten\*

\*In diesen Serien werden Informationen wie z.B. überprüfte Körperregionen, Datum und Uhrzeit, Hersteller und Typ des Gerätes und eine fortlaufende Nummer gespeichert.

In einer solchen Serien befinden sich die Bilder der entsprechenden Geräte. Die Bilder werden ebenfalls mit Metadaten versehen, die die Bildgröße, Fensterbreite und -größe sowie verschiedene gerätespezifische Daten enthalten.

**DICOM Network Protocol (Kommunikationsprotokoll)** – Alle vorhandenen medizinischen Bildgebungsanwendungen im Netzwerk eines Krankenhauses verwenden das DICOM Protokoll zum Informationsaustausch, vor allem von Bildern aber auch von Patienteninformationen und Auskünften über Verfahren. Das DICOM Kommunikationsprotokoll wird eingesetzt, um im Archiv Bildgebungsstudien zu durchsuchen und Bildgebungsstudien in jedem beliebigen Arbeitsplatz des Krankenhauses anzeigen zu lassen.

DICOM Dateien enthalten auch Informationen über die Positionierung des Patienten bei der Bildaufnahme und somit Pixel- bzw. Voxelinformationen, die vital für die Bildrekonstruktion bzw. die Erstellung eines 3D-Modells sind. Für diese Arbeit sind sie die wichtigsten Informationen. Sie können unter dem Modul „Image Plane Module“ einer DICOM Datei gefunden werden.

## 2.5. Knochendichtemessung

Der Begriff bzw. die Definition von Dichte ist relativ eindeutig, doch diese im Zusammenhang mit Knochen zu messen ist eine überaus komplexe Angelegenheit, denn Knochen bestehen aus verschiedenen Geweben (Kortikalis und Spongiosa) und sind teilweise porös [15]. Dennoch kann eine akkurate Schätzung der Knochendichte die Genauigkeit von fachspezifischen FE-Modellen erhöhen [16]. Die Literatur [17; 18; 19] unterscheidet verschiedene Begriffe bzw. Methoden zur Messung der Knochendichte:

**Real density (echte Dichte):** nasses Gewicht durch tatsächliches Volumen des Knochengewebes (real volume) geteilt.

**Apparent density (scheinbare Dichte;  $(\rho_{APP})$ ):** nasses Gewicht durch das gesamte Volumen des Knochen plus Porenraum (total sample volume) geteilt.

**Ash density (Aschendichte;  $(\rho_{ASH})$ ):** Aschengewicht durch das tatsächliche Volumen geteilt.

Doch keines dieser Verfahren ist in vivo (im lebenden Organismus) realisierbar.

Wie im Unterabschnitt 2.2 demonstriert, sind Schwächungskoeffizienten von der Dichte eines Materials abhängig. Das bedeutet, dass eine Umrechnung von dem linearen Schwächungskoeffizient bzw. HU-Wert in eine Materialdichte nachvollziehbar und plausible ist. Sprich es ist möglich, die Knochendichte anhand CT-Daten in vivo abzuschätzen bzw. zu bestimmen [19]. Doch es ist unklar, was genau für eine Knochendichte (ob echte, scheinbare oder Asche-Dichte) aus der Computertomographie geliefert wird [15]. Die aus der CT ermittelte Dichte wird in der Literatur als

radiologische Dichte (radiological density [16]), quantitative CT-Dichte (quantitative equivalent CT density ( $\rho_{\text{QCT}}$ ) [19]) oder auch als Knochenmineraldichte (englisch Bone Mineral/Material Density (BMD) [19; 20]) bezeichnet.

Zusätzlich wurden in [21; 16; 22] Beziehungen zwischen Knochenmineraldichte und Aschendichte gesucht und ermittelt.

Die Knochendichtemessung umfasst verschiedene, nicht-invasive Verfahren zur Bestimmung physikalischer Eigenschaften des Knochens. Das Standard-Diagnoseverfahren bzw. Messverfahren zur Bestimmung der Knochenmineraldichte ist die Dual-Röntgen-Absorptiometrie (DEXA; englisch Dual energy x-rays absorptiometry (DXA)), darüber hinaus gibt es aber weitere Verfahren wie die quantitative Computertomographie (QCT; englisch quantitative computed tomography) [23; 24; 20] oder eine standardisierte Ultraschalluntersuchung. Diese Techniken nutzen die Knochenmineraldichte zur Diagnose und Kontrolle von einer Osteoporose und anderer Knochenstoffwechselstörungen mit erhöhtem Risiko eines Knochenbruchs (für die Osteoporosediagnostik sind vor allem der Mineralgehalt im Knochen und die geometrische Anordnung bzw. Struktur der Knochen trabekel von Bedeutung).

Wichtig für diese Arbeit ist aber die quantitative Computertomographie, da sie mittels Computertomographie die physikalische Dichte ( $\text{kg}/\text{m}^3$ ) jedes Volumenelements ermittelt (DEXA-Verfahren messen hingegen eine flächenprojizierte Masse ( $\text{kg}/\text{m}^2$ )) [25] und da die in dieser Arbeit entwickelte MATLAB-Routine anhand von CT-Daten eine Materialzuordnung durchführt (siehe Kapitel 3).

Es wird bei der Aufnahme einer QCT ein Knochenmineralstandard, ein sogenanntes Kalibrations- bzw. Dichtephantom zwischen Tischoberfläche und Patient gebracht und mit aufgenommen [25; 26]. Da die Dichte von dem Knochenmineralstandard bekannt ist, ermöglicht die Kalibrierung mit diesem Phantom eine spätere Umrechnung der CT-gemessenen Hounsfield- in eine gültige Knochenmineraldichte ( $\rho_{\text{QCT}}$ ). Diese kann dann später in ( $\rho_{\text{ASH}}$ ) umgerechnet werden und aus dieser lassen sich weitere Materialeigenschaften wie z.B. der E-Modul ableiten (siehe Abschnitt 3.4).

## 2.6. Finite-Elemente-Methode

Zahlreiche ingenieurwissenschaftliche Probleme insbesondere im Bereich der Festigkeitslehre werden mit Hilfe von Differentialgleichungen (DGL) gelöst. Doch gibt es nur für wenige dieser Probleme geschlossene Lösungen. In der Realität sowie bei komplexen Aufgaben, bei denen die Geometrien typischerweise kompliziert sind, können

Differentialgleichungen nicht mehr analytisch gelöst werden. Um solche Problematik zu beheben, werden Näherungsverfahren eingesetzt, die eine hohe Genauigkeit ermöglichen und die möglichst einfach auf dem Computer umsetzbar sind [27; 28].

Die Finite-Elemente-Methode (FEM) ist das am weitesten verbreitete Näherungsverfahren zur Lösung von gewöhnlichen und partiellen DGL im Bereich der Festigkeitslehre, denn sie kann auch bei komplizierten Körpern effektiv eingesetzt werden. Zahlreiche kommerzielle Computerprogramme zur Lösung von Aufgaben der Mechanik (Konstruktionslehre, Fluidtechnik, Thermodynamik, etc.), arbeiten auf der Basis der Finite-Elemente-Methode.

Das Grundprinzip der FEM besteht darin, das zu untersuchende Gebiet in viele einfache endliche Teilgebiete (daher Finite-Elemente) zu zerlegen.

Trotz mangelnder Forschungserkenntnisse in den ersten Jahren ihrer Anwendung, stellten in den letzten Jahren viele Forschungsergebnisse das Potential und die Beschränkungen bzw. Grenzen der FEM dar. Die Finite-Elemente-Methode setzte sich im Bereich der Medizintechnik durch und ist heutzutage maßgeblich für Forschungsanwendung und Designanalysen innerhalb der Biomechanik [12]. Die Anzahl der Veröffentlichungen, in denen sie verwendet wird, stieg und steigt immer noch sehr stark weiter [11]. Aufgrund der sich entwickelnden Technologien und der erhöhten Computerleistung lässt sich in Zukunft mit aufregenden neuen Möglichkeiten rechnen, wodurch z.B. die Abhängigkeit von Tierversuchen verringert wird. Im Allgemeinen sind biologische Strukturen und klinische Fragestellungen sehr komplex. Wissenschaftlicher Fortschritt im Bereich der Biomechanik benötigt einerseits ein solides Verständnis der Festigkeitslehre bzw. der Technischen Mechanik allgemein, andererseits eine tiefe Kenntnis der komplexen Realität des menschlichen Körpers bzw. der Natur.

## 3. Die MATLAB-Routine

Das Ziel von *MatMap*, die hier in dieser Arbeit entwickelten MATLAB-Routine, besteht darin, den Elementen eines FE-Netzes Materialeigenschaften anhand ihrer Positionen zuzuweisen. Die Materialeigenschaften von Knochen (Knochendichte und Elastizitätsmodul) sind durch mathematische Umrechnungen der CT-Bilder entnehmbar. Das FE-Netz wird aus dem selben CT-Datensatz generiert.

Die Vorgehensweise von *MatMap* kann grundsätzlich in folgende Schritte gegliedert werden:

1 - Importieren

1.1 - Einlesen der CT-Daten

1.2 - Einlesen des FE-Netzes

2 - Kalibrierung der CT-Daten\*

3 - Material-Mapping

3.1 - Variante 1 - Convex Hull

3.2 - Variante 2 - Nearest Point Search

4 - Ableitung der elastischen Materialeigenschaften anhand der Knochendichte

5 - Komponentenerzeugung

6 - Exportieren

\* Dieser Schritt ist nicht zwingend notwendig und lässt sich nur durchführen, wenn bei der CT-Erstellung ein Phantom mit aufgenommen wurde. *MatMap* bietet die Möglichkeit an, die Korrelation zwischen HU-Wert und Knochendichte über eine lineare Näherung zu bestimmen, indem die Parameter der linearen Gleichung vom Benutzer direkt eingegeben werden. Gegebenenfalls ermöglicht *MatMap* die Bestimmung der Parameter anhand eines in der CT mit aufgenommen Dichtephantoms.

Für *MatMap* wurde eine grafische Benutzeroberfläche (englisch: Graphical User Interface - GUI) erstellt (siehe Abb. 3.1). Das Ziel des GUIs ist es, dem Benutzer einen leichteren Umgang mit *MatMap* zu ermöglichen. Zur Bedienung von *MatMap* sind keine Vorkenntnisse bezüglich MATLAB-Programmierung erforderlich.

### 3. Die MATLAB-Routine

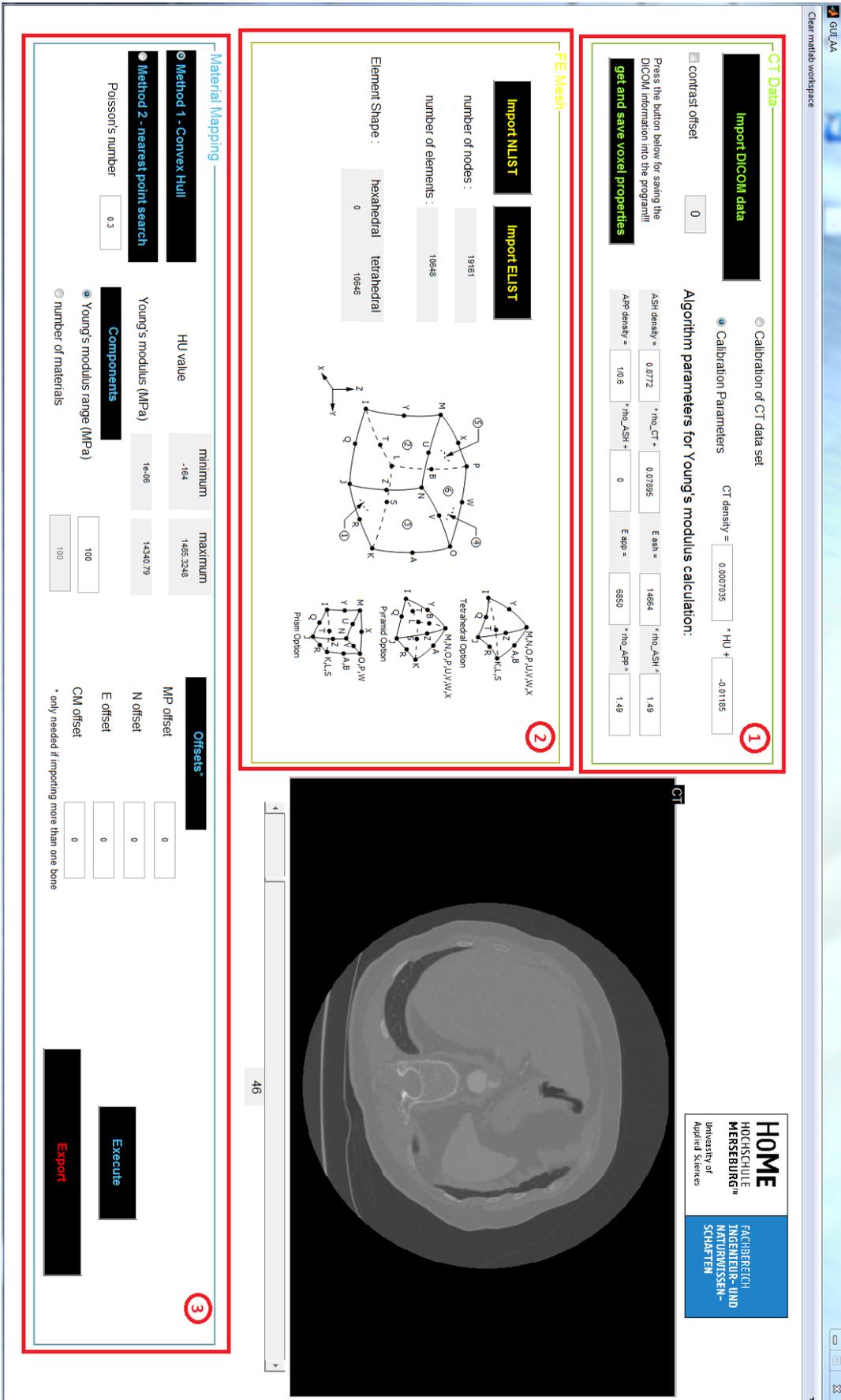


Abbildung 3.1.: Die grafische Benutzeroberfläche von MatMap

## 3.1. Importieren

Um *MatMap* anwenden zu können, sind im Wesentlichen zwei Eingaben erforderlich: Die CT-Daten und ein daraus erzeugtes FE-Netz. Die CT-Daten wurden im BG Klinikum Bergmannstrost Halle aufgenommen. Die FE-Netze wurden in ANSYS-Workbench generiert und in ANSYS-Classic als NLIST und ELIST abgespeichert (siehe Anhang A.1).

### 3.1.1. Einlesen der CT-Daten

Das Einlesen der DICOM-Daten erfolgt durch Anklicken des Buttons „Import DICOM data“ am GUI (im Bedienfeld 1 Abb. 3.1). Wenn dies geschieht, wird ein neues Fenster vom Microsoft Explorer geöffnet. Der Ordner mit den gewünschten DICOM-Bildern (das sogenannte DICOMDIR; wobei „DIR“ für directory (deutsch: Dateiverzeichnis) steht) soll vom Benutzer gesucht und ausgewählt werden. Das Programm liest das komplette DICOMDIR ein und zeigt anschließend eine Liste mit allen Informationen wie z.B. Studien und Serien an (siehe Bild. 3.2). Die Bilder von der ausgewählten Serie werden im rechten Rahmen im GUI angezeigt.

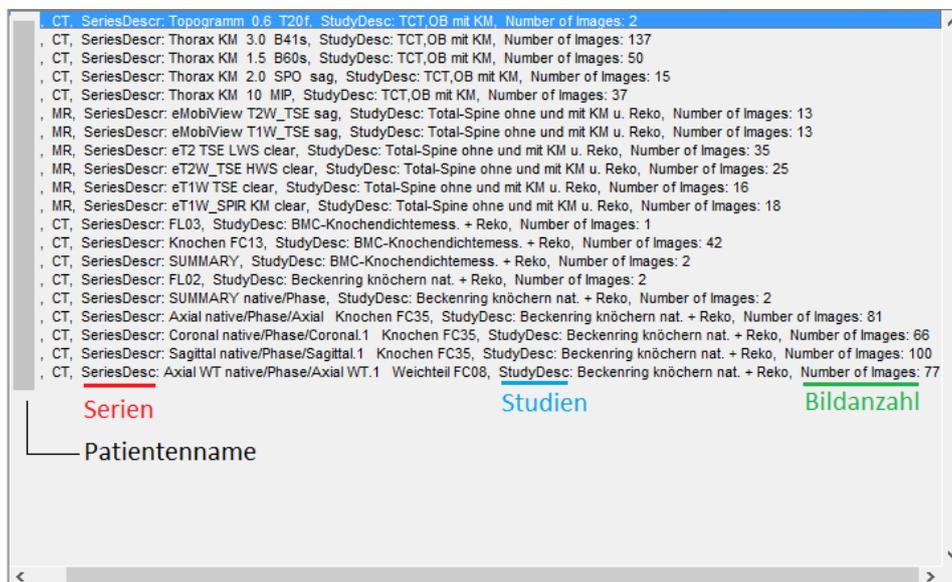


Abbildung 3.2.: DICOMDIR-Informationsliste

Die Begriffe Voxel und Pixel werden hier oftmals verwendet und können zur Verwirrung führen. Um dies zu vermeiden, werden hier Begriffe und ihre Unterschiede näher gebracht. Ein Pixel (picture element) ist ein Bildelement. Ein Bild ist eine Matrix von Pixeln bzw. Pixelmatrix (siehe Abb. 3.3). Pixel sind allgemein gleichgroß

und haben die Form eines Quadrats (siehe Abb. 3.4). Ein Voxel (englisch volume element) ist nichts anderes als ein Pixel in einem dreidimensionalen Raum. Also ein Volumenelement, wie ein Würfel. CT-Bilder bestehen aus Pixeln, die einen HU-Wert besitzen. Pixel mit großen HU-Werten (sprich Materialien mit großen Dichten, in der Regel Knochen) werden auf dem Bild weiß dargestellt. Pixel mit niedrigen HU-Werten werden schwarz abgebildet. Für alle andere HU-Werte wird dem Pixel ein entsprechender Grauwert zugeordnet.

Die Metadaten des ausgewählten DICOM Objekts werden durch Anklicken des Buttons „Import DICOM data“ im MATLAB-Workspace gespeichert. Nachfolgend werden ihre wichtigsten Informationen vorgestellt:

- **Pixelmatrix:** *MatMap* speichert alle Pixel in einer 512 x 512 x n dreidimensionalen Matrix, wobei n der Anzahl von Bildern der ausgewählten Serie entspricht (Pixel bzw. Voxel(2,3,4) der MATLAB-Routine bezieht sich auf das Voxel der dritten Spalte der zweiten Reihe der vierten Schicht bzw. des vierten CT-Bildes). Abb. 3.3 zeigt ein CT-Bild auf der linken Seite und die zur ROI (Region of Interest; deutsch Bereich von Interesse) gehörenden Pixel und ihre jeweiligen HU-Werte auf der rechten Seite.

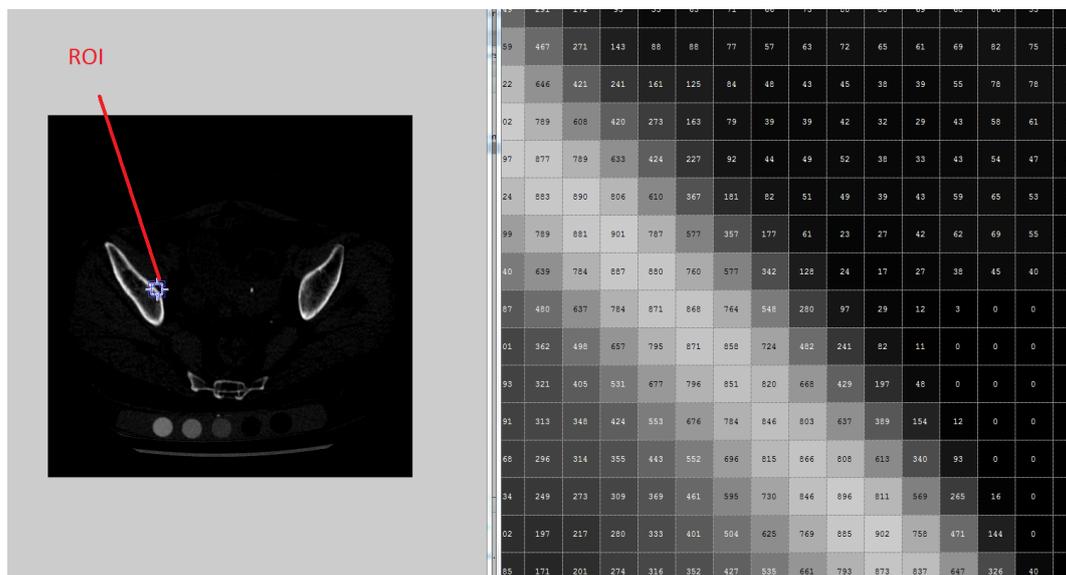


Abbildung 3.3.: Bild Pixel-Matrix im MATLAB

- **PixelSpacing:** X und Y (Linie und Spalte) Abstand zwischen zwei nebeneinander liegenden Pixel-Schwerpunkten des selben Bildes (bzw. auf der selben Schicht). Mit anderen Worten: Breite und Länge des Pixels. Abb. 3.4 verdeutlicht den Pixelabstand (deutsche Übersetzung für Pixel spacing).

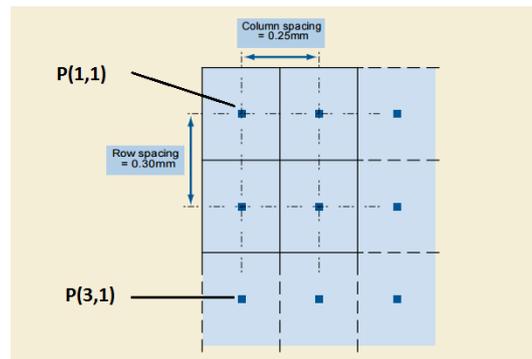


Abbildung 3.4.: Darstellung des Pixelabstandes und der Pixel-Matrixkoordinate

- **ImageOrientationPatient:** Der Kosinus des Winkels zwischen der ersten Linie (der Pixelmatrix) und der ersten Spalte bezüglich des Patienten. Dies gibt die Orientierung des Bildes an. Im Matlab wird dies als 6x1 Orientierungsmatrix dargestellt.
- **ImagePositionPatient:** X, Y und Z Koordinaten der linken oberen Ecke des Bildes (Schwerpunkt des Pixels  $P(1,1,n)$  der Bild-Matrix) in mm. In Abb. 3.4 als  $P(1,1)$  und im Abb. 3.6 als IPP dargestellt.
- **SliceThickness:** Abstand zwischen zwei benachbarten Schichten (siehe Abb. 3.5). Die Schichtdicke (deutsche Übersetzung für slice thickness) entspricht der Höhe eines Voxels.
- **InstanceNumber:** Gibt die Position eines Bildes in dem Bildstapel an. Dies ist wichtig, um die Bilder in der richtigen Reihenfolge darzustellen, da die CT-Bilder sich nicht immer in der richtigen Reihenordnung im Ordner befinden.

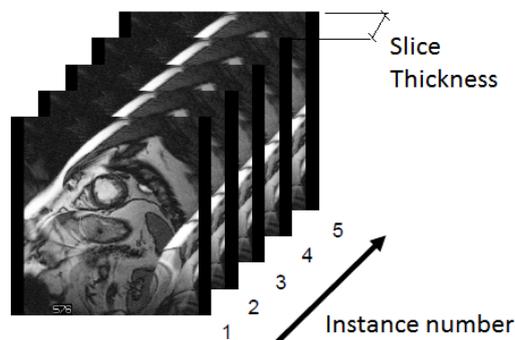


Abbildung 3.5.: Darstellung eines CT-Bildstapels

- **RescaleIntercept** und **RescaleSlope**: Wenn ein CT-Bild aus irgendeinem Grund vorher umskaliert wurde (mit einem Offset/Faktor), werden hier die Skalierungsparameter gespeichert.

Der Großteil dieser Informationen ist unter dem Image Plane Modul (dicom.nema.org) des DICOM-Standards zu finden. Die MATLAB-Funktionen *dicomread* und *dicominfo* geben Zugriff auf diese Informationen.

Der Button „get and save voxel properties“ speichert die Koordinaten der Voxel-Schwerpunkte (Voxel-SP) gemäß Gleichung 3.1 (siehe auch Abb. 3.6),

Allgemein, werden die Koordinaten des Pixels der Position (L,S,SC), also das Pixel in der Linie L der Spalte P der Schicht SC, wie folgt bestimmt:

$$\begin{vmatrix} x_{P(L,S,SC)} \\ y_{P(L,S,SC)} \\ z_{P(L,S,SC)} \end{vmatrix} = \begin{vmatrix} x_{P(1,1,SC)} \\ y_{P(1,1,SC)} \\ z_{P(1,1,SC)} \end{vmatrix} + (S-1) \cdot PS(2) * \begin{vmatrix} Xx \\ Xy \\ Xz \end{vmatrix} + (L-1) \cdot PS(1) * \begin{vmatrix} Yx \\ Yy \\ Zz \end{vmatrix} \quad (3.1)$$

wobei PS für PixelSpacing steht und Xx die Projection von X in der x-Achse ist, Xy die Projection (oder Kosinus) von X in der y-Achse ist usw..

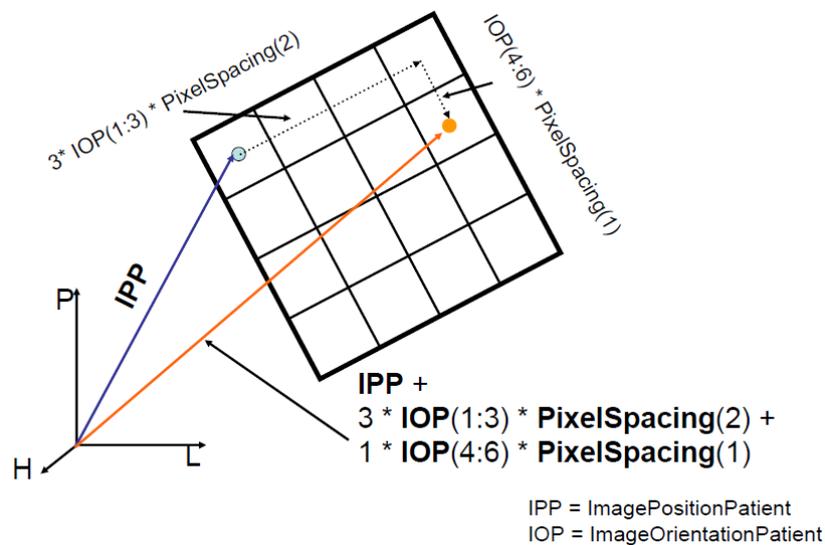


Abbildung 3.6.: Berechnung der Pixelkoordinaten [3]

### 3.1.2. Einlesen des FE-Netzes

Das Importieren des Netzes in MATLAB erfolgt durch Einlesen von zwei Dateien: eine Elemente-Liste und eine Knoten-Liste. Sie werden im Folgenden als ELIST und

NLIST bezeichnet, denn sie werden so in ANSYS-Classic abgespeichert. Die Buttons „Import NLIST“ und „Import ELIST“ (siehe Abb. 3.1) sind für das Einlesen der beiden Dateien verantwortlich.

Eine NLIST liefert *MatMap* die Knoten-ID und ihre Koordinaten. Eine ELIST gibt die Elementanzahl an und aus welchen Knoten ein Element besteht (siehe Abb. 4.2). *MatMap* kann nur Tetraeder und Hexaeder vom Typ *SOLID186* der ANSYS-Bibliothek einlesen, da sie bei der Erzeugung von Finite-Elemente-Netzen in ANSYS am häufigsten vorkommen und für ein genaueres Ergebnis bzw. für die Erzeugung eines feinen Netzes genügen. Ein Element vom Typ *SOLID186* besteht aus 20 Knoten, die in ANSYS als I, J, K, L, M, N, O, P, Q, R, S, T, U, W, X, Y, Z, A und B bezeichnet werden (siehe Abb. 3.7). Im Fall eines Tetraeders beinhaltet das Element nur die Knoten I, J, K, M, Q, R, T, Y, Z und A (siehe Abb. 3.7).

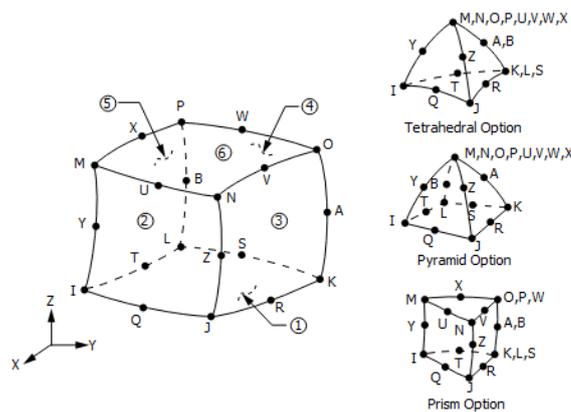


Abbildung 3.7.: Solid186 (© ANSYS-HELP)

Nach dem Einlesen von N- und ELIST werden in *MatMap* zwei Matrizen erzeugt, die die Eckknoten und die Schwerpunkte der Elemente beinhalten. Die Berechnung des Schwerpunktes erfolgt durch das arithmetische Mittel der Koordinaten, da alle Eckknoten die selbe Gewichtung haben. Es wurden hier nur die Eckknoten des Elements betrachtet, da für die in dieser Arbeit vorgestellten Material-Mapping-Methoden die Mittenknoten nicht notwendig sind (siehe Abschnitt 3.3.1).

## 3.2. Kalibrierung der CT-Daten

Wenn bei der Aufnahme einer CT ein Phantom zur Kalibrierung mit aufgenommen wird, kann eine mathematische Beziehung zwischen dem HU-Wert und der Knochenmineraldichte ermittelt werden. Diese Beziehung wird üblicherweise durch

einen linearen Zusammenhang beschrieben. Auch die  $\rho_{\text{QCT}}\text{-}\rho_{\text{ASH}}$ - und  $\rho_{\text{ASH}}\text{-}\rho_{\text{APP}}$ -Korrelationen werden meistens linear ermittelt [26; 22; 29; 16; 25].

*MatMap* bietet dem Benutzer die Möglichkeit, die Parameter dieser Zusammenhänge direkt einzugeben. Falls jedoch ein Phantom vorhanden ist, ist es möglich, die Gleichungsparameter durch eine Kalibrierung zu bestimmen. Abb. 3.8 zeigt ein Beispiel eines CT-Bildes, in dem ein Phantom mit fünf verschiedenen Dichten mit aufgenommen wurde.

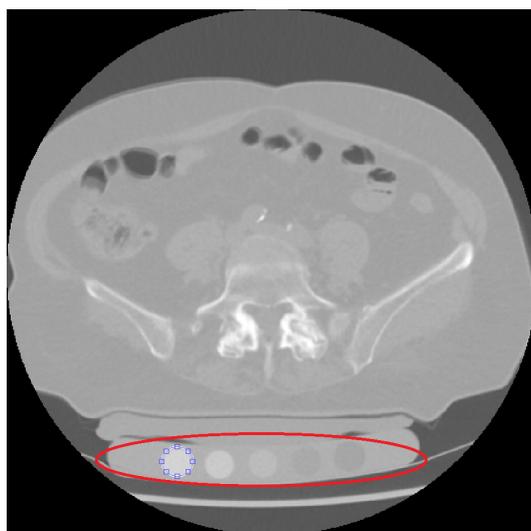


Abbildung 3.8.: Beispiel eines Phantoms

Um die Parameter der linearen Beziehung zu bestimmen, sind im Grunde zwei Angaben notwendig: Die Knochenmineraldichten des Phantoms und ihre entsprechenden HU-Werte. Es gibt verschiedenen Wege, den HU-Wert eines bestimmten Kalibrationsphantoms zu bestimmen. *MatMap* macht dies jedoch nicht automatisch, da sie vorher nicht weiß, ob es ein Phantom im Bild gibt und wo es sich im Bild befindet. Das Ziel von *MatMap* hierbei ist es, eine CT-Kalibrierung mit reproduzierbaren Ergebnissen zu schaffen. Um die Kalibrierung in *MatMap* durchzuführen, soll der Schaltknopf „calibration of CT data set“ (siehe Bedienfeld 1 in Abb. 3.1) betätigt und folgende Schritte abgearbeitet werden:

1 - Eingabe der ID bzw. Instance number der zu untersuchenden Bilder:

Vor Beginn der Kalibrierung sollen die CT-Bilder im GUI angeschaut werden, um die Bilder, in welchen es ein Phantom gibt, identifizieren zu können. Wenn zum Beispiel ein CT-Bildsatz von 100 Bildern ein Phantom zwischen dem 10. und 90. Bild hat, könnten die Bilder zwischen 10 und 90 ausgewählt werden. Es wird jedoch empfohlen, einen Abstand von dem Rand des Phantoms zu halten, da diese durch das Material des Behälters bzw. der Decke beeinflusst werden könnten (z.B. Bilder

15. bis 85 auswählen). *MatMap* fragt nach dem ersten Bild und nach der Anzahl der zur Kalibrierung gehörenden Bilder. Im hier betrachteten Beispiel sollte das Bild mit der Bild-ID 15 und die Anzahl der zur kalibrierenden Bilder 70 (85-15) eingegeben werden.

2 - Eingabe der Anzahl der Knochenmineralstandards des Phantoms:

Anschließend soll die Anzahl der Knochenmineralstandards des Phantoms angegeben werden. Abb. 3.8 hat z.B. 5 Knochenmineralstandards. Doch wenn nur 4 oder weniger betrachtet werden sollten, kann auch dies eingestellt werden.

3 - Position des Phantoms:

Damit *MatMap* die Position des Knochenmineralstandards erfassen kann, soll als nächstes für jeden Knochenmineralstandard des Phantoms ein Kreis bzw. eine Ellipse möglichst genau um seinen Rand herum aufgezogen und der entsprechende Wert für seine Knochenmineraldichte eingegeben werden (siehe Abb. 3.9). Die Skizze wird in dem ersten Bild der Kalibrierung gezeichnet (in unserem Beispiel im Bild mit ID 15)

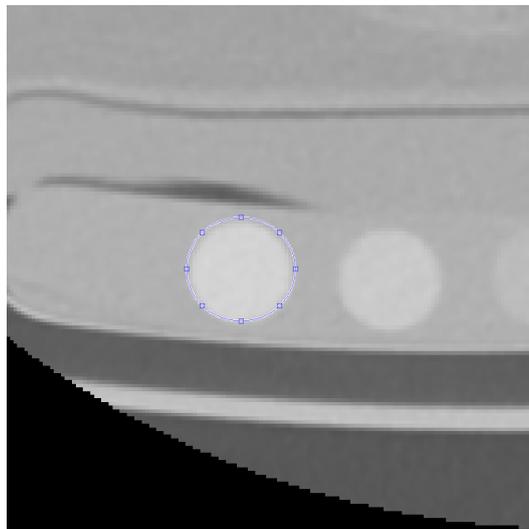


Abbildung 3.9.: Beispiel eines Kreises um den Rand des Knochenmineralstandards

Wenn ein Kreis bzw. eine Ellipse im Bild erzeugt wird, speichert *MatMap* intern eine maskierte Bild-Matrix, in der aller Pixeln außerhalb der aufgezogenen Form der Wert 0 und allen Pixeln innerhalb der Wert 1 zugewiesen wird (siehe Abb. 3.10).

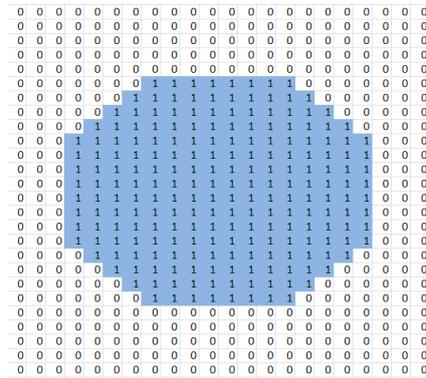


Abbildung 3.10.: maskierte Matrix

*MatMap* erfasst dann die Breite und die Höhe (Einheit in Pixelanzahl) und bestimmt den Mittelpunkt bzw. das mittlere Pixel der Geometrie (Im Fall einer Breite bzw. Höhe mit gerader Anzahl von Pixeln (z.B. Breite = 16 Pixel) setzt *MatMap* das linke bzw. obere Pixel als Mittelpunkt). Abb. 3.11 zeigt ein Beispiel, in dem Breite und Höhe (gelbe Linien) eine gerade Pixelanzahl aufweisen. Das ausgewählte mittlere Pixel (rot) befindet sich nicht genau in der Mitte, sondern ist etwas nach links und nach oben verschoben.

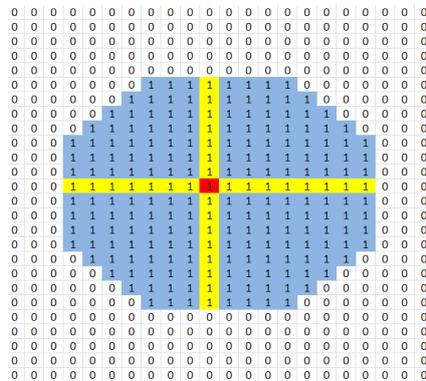


Abbildung 3.11.: Bestimmung des mittleren Pixels

Anschließend erzeugt *MatMap* ein Rechteck, dessen Mittelpunkt das ermittelte mittlere Pixel ist. Höhe und Breite des Rechtecks sind jeweils die Hälfte der Breite und der Höhe der Ellipse (siehe Abb. 3.12). Soll einer von denen keine ganze Zahl sein (z.B. Durchmesser gleich 21 und Radius gleich 10,5), übernimmt die Routine die kleinste ganze Zahl (10).

### 3. Die MATLAB-Routine

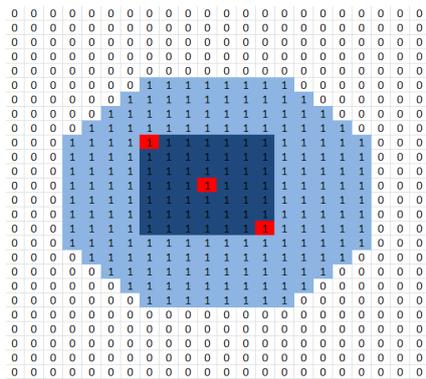


Abbildung 3.12.: Erzeugung eines Rechtecks um das mittlere Pixel herum

*MatMap* greift dann auf die originale Bildmatrix (also nicht maskiert) zu und berechnet den arithmetischen HU-Mittelwert der Pixel innerhalb des Rechtecks (3.13).

450	460	477	475	480	484	479	492	489	465	492	514	496	448	447	451	425
482	488	493	487	491	488	476	492	489	472	478	491	474	443	446	464	470
496	494	493	493	485	474	479	497	494	483	486	491	475	453	453	472	485
497	491	496	499	483	471	489	502	496	483	476	472	466	458	463	480	490
491	480	490	501	492	477	487	485	478	476	467	455	454	453	465	485	484
488	474	482	496	489	479	476	461	468	476	474	474	463	453	462	476	466
480	473	479	484	476	469	458	442	456	468	474	485	472	459	463	474	466
461	463	472	473	467	464	455	441	448	464	478	492	483	472	477	487	484
459	467	477	473	466	468	471	460	457	472	487	496	491	483	481	489	486
477	500	504	486	464	458	466	467	467	479	491	498	496	480	472	481	483
491	517	519	496	460	448	450	463	478	496	500	501	499	473	459	475	488
493	508	506	474	450	447	449	463	484	511	518	511	501	465	446	467	486
489	495	486	461	445	458	466	470	477	498	500	501	482	460	456	478	489
483	487	486	475	462	479	495	482	472	475	473	477	467	465	493	503	501
482	476	480	487	491	500	512	497	478	482	480	483	479	476	494	504	497
480	464	468	499	515	507	495	481	473	485	483	476	484	493	490	488	476
471	455	461	504	517	486	461	455	464	478	462	452	464	492	487	473	451

Abbildung 3.13.: ermitteltes Rechteck in der nicht maskierten Bild-Matrix

Mit der Vermutung, dass das Phantom in allen Bildern die gleiche Position hat, wird dieser Vorgang für alle weiteren Bilder durchgeführt (zurück zum Beispiel: es wird ein Rechteck in den Bildern von ID 15 bis 85 erzeugt und der entsprechende HU-Mittelwert ermittelt. Die Ellipse muss nur im ersten Bild der Kalibrierung skizziert werden).

Nachdem alle Kreise skizziert und alle Knochenichten des entsprechenden Knochenmineralstandards eingegeben wurden, berechnet das Programm anhand der fünf aufgenommenen Punkte einen linearen Zusammenhang zur Berechnung der Knochenmineraldichte (siehe Abb. 3.14).

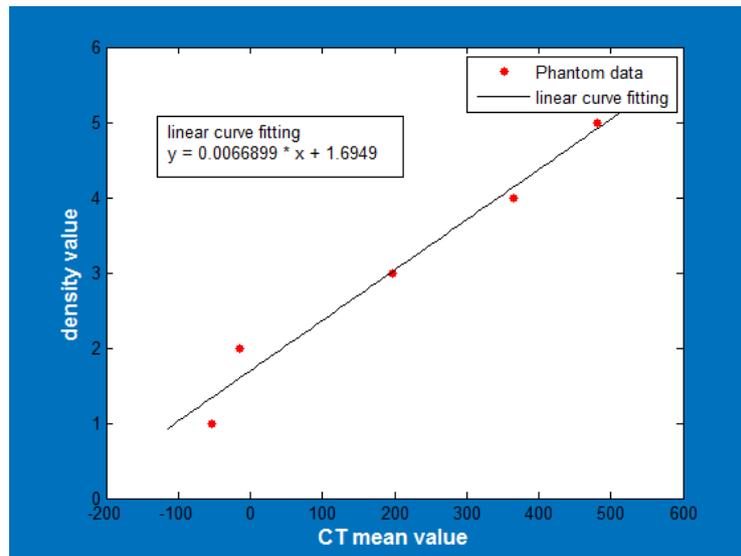


Abbildung 3.14.: ermittelte lineare Kurvenanpassung des HU- $\rho_{QCT}$ -Zusammenhangs

Bei wiederholt durchgeführten Experimenten ließ sich feststellen, dass die in dieser Arbeit verwendete Kalibrierung zur Ermittlung der Parametern zur HU- $\rho_{QCT}$ -Korrelation reproduzierbar ist (wenn der Kreis bzw. die Ellipse möglichst genau um den Rand des Knochenmineralstandards skizziert wird).

### 3.3. Material-Mapping

Das Ziel des Material-Mappings besteht darin, für jedes Element des FE-Netzes einen HU-Wert anhand der CT-Daten zu berechnen. Zwei verschiedene Ansätze zum Material-Mapping werden in dieser Arbeit behandelt: Die erste Variante weist jedem Element den arithmetischen HU-Mittelwert der Voxeln, deren Schwerpunkte innerhalb seiner Geometrie liegen, zu. Die zweite Variante weist jedem Element den arithmetischen HU-Mittelwert der am nächsten am Element liegenden Voxel-Schwerpunkte zu. Der Einfluss der Material-Mapping Strategie auf die Materialzuweisung wurde untersucht. Die in dieser Arbeit angegangenen Strategien wurden untereinander und mit der Public-Domain-Software Bonemat verglichen (siehe Abschnitt 5).

#### 3.3.1. Variante 1 - Convex Hull

Das Ziel der Variante 1 besteht darin, dem Element nur die HU-Werte der Voxel-SP zuzuweisen, die sich innerhalb seiner Geometrie bzw. seines Volumens befinden.

Zur Abstrahierung des Volumen eines Elements wird hier seine konvexe Hülle ermittelt.

Die konvexe Hülle (deutsche Übersetzung für convex hull) einer endlichen Menge von Punkten ist die kleinste konvexe Menge, die sie umfasst. Mathematisch ausgedrückt [30]:

Gegeben sei eine Menge  $M$  von Punkten in der Ebene. Die konvexe Hülle von  $M$  ist die kleinste konvexe Menge, in der  $M$  enthalten ist.

Satz: Wenn  $M$  eine endliche Menge ist, so ist die konvexe Hülle von  $M$  ein konvexes Polygon oder, wenn die Punkte von  $M$  alle auf einer Linie liegen, ein Liniensegment.

Abb. 3.15<sup>1</sup> verdeutlicht die Definition der konvexen Hülle.

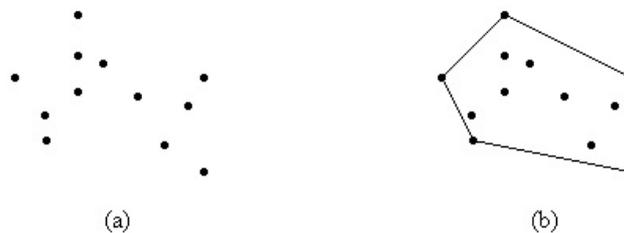


Abbildung 3.15.: (a) endliche Menge von Punkten. (b) konvexe Hülle der Punktmenge

Wie bereits erwähnt, sucht die erste Variante die Voxel-Schwerpunkte, die innerhalb eines Elements liegen und weist diesem den HU-Mittelwert dieser Voxeln zu. Die Variante 1 basiert auf der Berechnung der konvexen Hüllen jedes einzelnen Elementes.

Da die Koordinaten der Kanten eines Elementes bekannt sind, abgesehen davon, ob es ein Tetraeder oder ein Hexaeder ist, lässt sich leicht eine konvexe Hülle mit MATLAB erzeugen (mit der Funktion `convhull`<sup>2</sup>) und überprüfen, ob sich ein beliebiger Punkt innerhalb dieser befindet (mit der Funktion `inhull`<sup>3</sup>, siehe Abb. 3.16). Wenn viele Elemente und Bilder berücksichtigt werden müssen, kann eine Überprüfung, ob sich ein Punkt innerhalb oder außerhalb einer konvexen Hülle befindet, sehr hohen Rechenaufwand mit sich bringen. Z.B. eine in dieser Arbeit evaluierte Studie beinhaltet 391 Bilder (sprich  $512 \times 512 \times 391 = 102.498.304$  Voxel) und ihr entsprechendes Netz besitzt über 200.000 Elemente. Wenn man überprüfen würde, ob sich jeder Voxel-SP innerhalb eines Elements befindet, würde MATLAB in einer Schleife 102.498.304 Berechnungen durchführen. Wird das für jedes Element gemacht, wird dieser Wert noch mit 200.000 multipliziert.

<sup>1</sup><http://www.inf.fh-flensburg.de/lang/algorithmen/geo/convex.htm>

<sup>2</sup><https://de.mathworks.com/help/matlab/math/computing-the-convex-hull.html>

<sup>3</sup><https://de.mathworks.com/matlabcentral/fileexchange/10226-inhull>

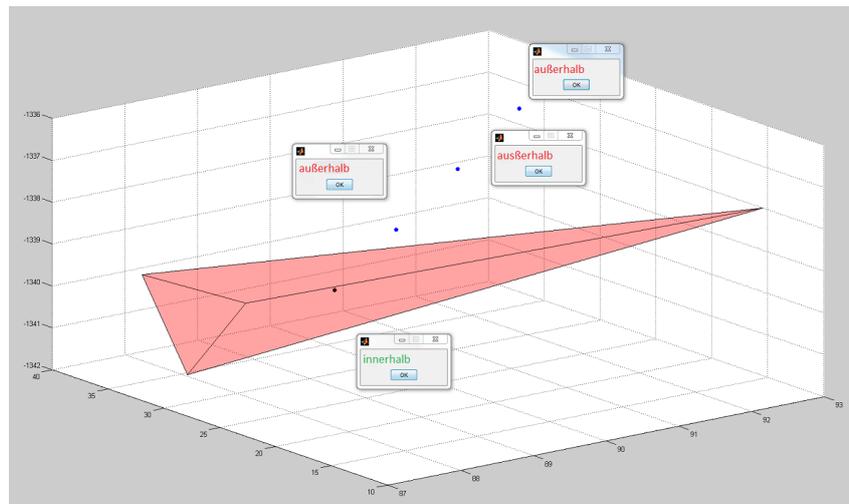


Abbildung 3.16.: Mit der Funktion *inhull* wird überprüft, ob sich ein Punkt innerhalb oder außerhalb einer konvexen Hülle befindet

Um den Rechenaufwand von *MatMap* zu minimieren, wurde eine Begrenzung der zu untersuchenden Voxel festgelegt. Ziel dieser Begrenzung ist es, ausschließlich die in der Nähe des Elements liegenden Voxel-Schwerpunkte zu betrachten, um eine schnellere Berechnung realisieren zu können.

Die xyz-Begrenzung, wie sie hier weiterhin genannt wird, sucht zunächst die kleinsten und die größten Koordinaten der Eckknoten des Elements. Im Abb. 3.18 sind z.B.  $x_1, y_1$  die kleinsten und  $x_3, y_2$  die größten Knotenkoordinaten.

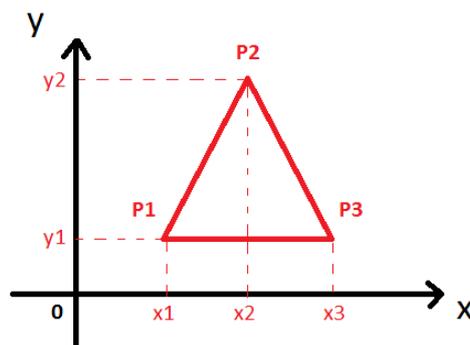


Abbildung 3.17.: Betrachtung eines Tetraeders im 2D-Raum

Nachdem die extremen Knotenkoordinaten identifiziert wurden, wird ein Offset dazu addiert. Ein Offset-Wert in Höhe des Wertes des Halb-PixelSpacings wurde hier genommen, da er (sogar für den unmöglichen Fall, wenn die größte und kleinste Koordinate identisch sind (In Abb. 3.18 wenn  $x_1 = x_3$ )) sicher stellt, dass es mindestens ein Voxel-SP zwischen beiden äußeren Rahmen der Begrenzung gibt.

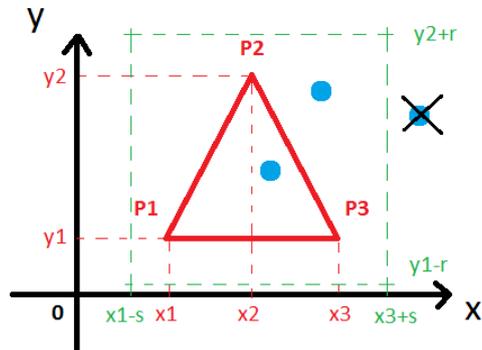


Abbildung 3.18.: Rahmen der Begrenzung, wobei  $s$  und  $r$  die Hälfte der PixelSpacing jeweils für die  $x$  und  $y$  Richtung sind. Bei Voxel-Schwerpunkten, die sich außerhalb der  $xyz$ -Begrenzung befinden, wird nicht überprüft, ob sie sich innerhalb der konvexen Hülle der Elementgeometrie befinden

Nur die Punkte innerhalb dieser Begrenzung werden in der Berechnung der Variante 1 betrachtet. Abb. 3.19 verdeutlicht die Funktionsweise der  $xyz$ -Begrenzung.

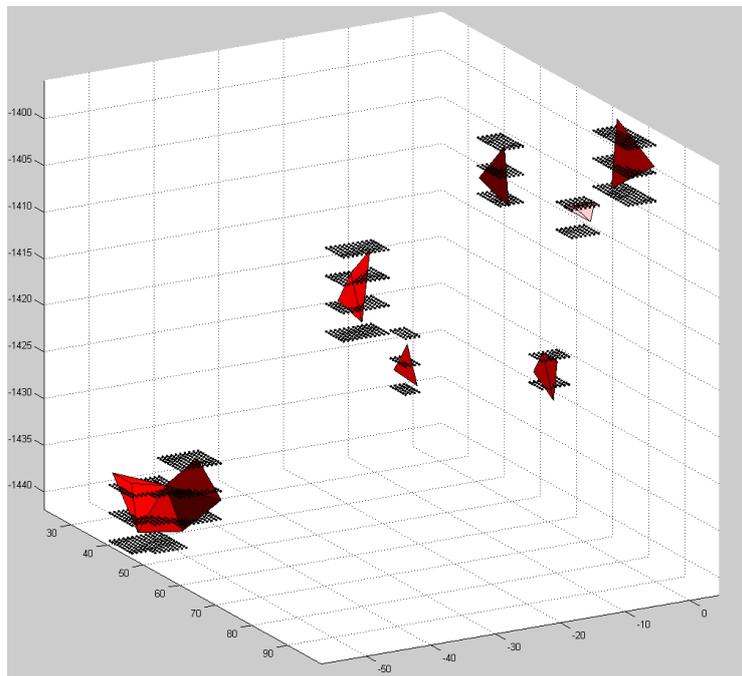


Abbildung 3.19.: Elemente und die entsprechenden, innerhalb der  $xyz$ -Begrenzung liegenden Voxel-Schwerpunkte

Selbst wenn viele Berechnungen zur Bestimmung der  $xyz$ -Begrenzung notwendig sind, wurde empirisch festgestellt, dass die gesamte Rechenzeit des Material-Mapping

reduziert wird. Um wie viel Prozent oder wie stark sie die Berechnung beschleunigt, konnte aus Zeitgründen in dieser Arbeit nicht untersucht bzw. festgelegt werden.

Im unteren Feld ist der Algorithmus der xyz-Begrenzung dargestellt. Die Berechnung wird für jedes einzelne Element in der For-Schleife durchgeführt.

Code 3.1: Algorithmus der xyz-Begrenzung

```
1 %\caption={Zeilen Code der xyz-Begrenzung} \label{lst:v1},captionpos=t
2 for ell = 1:num_ele
3 % restrict search
4 my_element = element_vertices(:, :, ell);
5 max_my_element = max(my_element);
6 max_x = max_my_element(1) + (voxel_size(1)/2); % x max + half ...
   voxel x dimension
7 max_y = max_my_element(2) + (voxel_size(2)/2); % y max + half ...
   voxel y dimension
8 max_z = max_my_element(3) + (voxel_size(3)/2); % y max + half ...
   voxel z dimension
9
10 min_my_element = min(my_element);
11 min_x = min_my_element(1) - (voxel_size(1)/2); % x min - half ...
   voxel x dimension
12 min_y = min_my_element(2) - (voxel_size(2)/2); % y min - half ...
   voxel y dimension
13 min_z = min_my_element(3) - (voxel_size(3)/2); % y min - half ...
   voxel z dimension
14
15 bnum = 1;
16
17 % find layers surrounding the element (z-Begrenzung)
18 for schicht = 1:tot_num_layers
19     if ((voxel_sp_complete(1,3,schicht) >= min_z) &&...
20         (voxel_sp_complete(1,3,schicht) <= max_z))
21         border_layer(bnum,1) = schicht;
22         border_layer(bnum,2) = voxel_sp_complete(1,3,schicht);
23         bnum = bnum + 1;
24     end
25 end
26 ...
27 ...
28 %     Add X, Y and Z borders
29 %     create matrix with voxel that are on the layers nearby my ...
   element
30 lnum = 1;
31 % for each surrounding layer (z-Begrenzung)
32 for num_layers = border_layer(1,1): 1 : border_layer(end,1)
33     for id_voxel = 1:length(voxel_sp_complete) % = 512*512
```

```
34     if ((voxel_sp_complete(id_voxel,1,num_layers) ≥ min_x) &&...
35         (voxel_sp_complete(id_voxel,1,num_layers) ≤ max_x)) && ...
36         ((voxel_sp_complete(id_voxel,2,num_layers) ≥ min_y) &&...
37         (voxel_sp_complete(id_voxel,2,num_layers) ≤ max_y))
38
39         voxel_sp_lim_xyz(lnum,1:3,ell) = ...
40             voxel_sp_complete(id_voxel,1:3,num_layers);
41         voxel_id_xyz(lnum,1,ell) = id_voxel;
42         voxel_id_xyz(lnum,2,ell) = num_layers;
43         lnum = lnum + 1;
44     end
45 end
46 ...
```

Für die Ermittlung der konvexen Hülle wurden in *MatMap* nur die Eckknoten der Elemente betrachtet (I, J, K und M im Fall eines Tetraeders; siehe Abb. 3.7). Üblicherweise befinden sich die anderen Knoten eines Elements exakt in der Mitte zwischen zwei benachbarten Eckknoten oder von dort nicht weit entfernt. Das bedeutet, dass sie keinen großen Einfluss auf die konvexe Hülle haben. Nur wenn sie etwas nach außen gewölbt sind, wäre die konvexe Hülle größer. Da eine Implementierung nur mit Eckknoten weniger Programmieraufwand benötigt und da angenommen wurde, dass die Betrachtungen der anderen Knoten keinen großen Einfluss auf das Ergebnis hat, verwendet *MatMap* nur die Eckknoten zur Ausbildung einer konvexen Hülle.

Wenn sich nur ein Voxel-SP innerhalb der konvexen Hülle eines Elements befindet, wird diesem Element der HU-Wert dieses Voxels zugewiesen. Wenn es mehrere Voxel-Schwerpunkte innerhalb eines Elements gibt, wird, wie bereits gesagt, der arithmetische Mittelwert deren HU-Werte gebildet. Wenn sich beispielsweise ein Voxel-SP mit HU-Wert von 300 und ein Voxel-SP mit HU-Wert von 400 innerhalb des Elementes befinden, wird diesem Element ein HU-Wert von 350 zugewiesen.

Doch der Fall, dass kein Voxel-SP innerhalb der konvexen Hülle des Elements zu finden ist, wurde noch nicht betrachtet. Es ist nicht unüblich, dass es kein Voxel-SP innerhalb einer Elementgeometrie gibt. *MatMap* zeigte, dass dieser Fall häufig eintritt. Dies geschieht z.B. bei einer sehr feinen Vernetzung, in der die Elementgröße kleiner als der kleinste Abstand zwischen zwei nebenliegenden Voxel-SP (also PixelSpacing) ist. Eine weitere Ursache dafür sind flache bzw. abgeflachte Elemente. Abb. 3.20 stellt links eine konvexe Hülle dar, die Voxel-SP beinhaltet und rechts eine leere konvexe Hülle bzw. ein leeres Element.

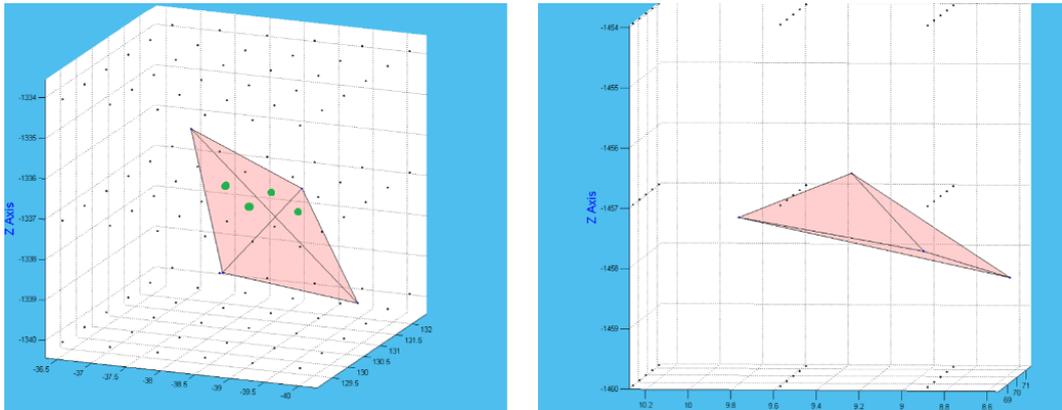


Abbildung 3.20.: Beispiel eines Elements mit und ohne Voxel-Schwerpunkte innerhalb seiner konvexen Hülle (links bzw. rechts)

Den leeren Elementen wird kein HU-Wert und somit keine Materialeigenschaften zugewiesen, was ungewollt ist und zu Problemen beim Import in ANSYS-Classic führt. *MatMap* betrachtete bisher nur den Schwerpunkt des Voxels. Das bedeutet, dass das ganze Volumen des Voxels in der Realität nicht betrachtet wird. Um mehr vom Voxel-Volumen auszunutzen und um das Leer-Element-Problem zu beheben, wurde vorgeschlagen, das Voxel aufzuteilen. Um ein Voxel bzw. Würfel in mehrere gleichgroße Würfel aufzuteilen, sind mindestens drei Schnitte notwendig (Halbierung der Breite, Länge und Höhe). Die einfachste hier gewählte Aufteilung generiert 8 neue Voxel, eine Aufteilung der 2. Ordnung generiert 27 gleichgroße Voxel. Abb. 3.21 stellt die Aufteilung eines Würfels bzw. Voxels dar.

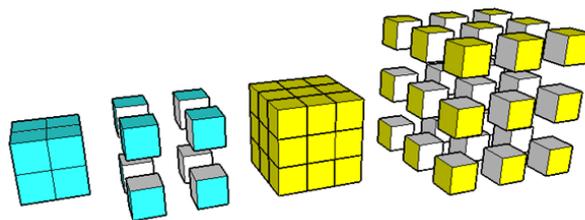


Abbildung 3.21.: Aufteilung der 1. und 2. Ordnung eines Voxels

Eine Aufteilung der  $n$ -ten Ordnung generiert  $(n+1)^3$  Voxel bzw. Voxel-Schwerpunkte. Je mehr Voxel es gibt, desto wahrscheinlicher ist es, dass sich ein Voxel-SP innerhalb der konvexen Hülle des Elements befindet. Genau deshalb teilt die Variante 1 die Voxel auf und überprüft, ob die neuen Voxel-SP innerhalb der konvexen Hülle liegen. Wenn ein Element nach der 1. Ordnung Aufteilung noch immer leer ist, führt *MatMap* eine Aufteilung 2. Ordnung für dieses spezifisches Element durch usw. bis zur dritten Ordnung (64 Voxel-Schwerpunkte). Gibt es immer noch leere Elemente bzw. wurde noch kein Voxel-SP innerhalb der konvexen Hülle dieser Elementen gefunden,

wird die Variante 2 für diese Elemente eingesetzt (siehe 3.3.2). Die Variante 2 sucht die bezüglich der Elementeckenknoten nächstgelegenen Voxel-Schwerpunkte. Diese Variante ist zuverlässig und weist den Elementen immer einen HU-Wert zu, weil sie unabhängig davon ist, ob die Voxel innerhalb oder außerhalb der Elementgeometrie liegen. Das untere Flussdiagramm verdeutlicht die Vorgehensweise der Variante 1.



Abbildung 3.22.: Vorgehensweise der Variante 1

### 3.3.2. Variante 2 - Nearest Point Search

Im Unterschied zur Variante 1 gibt es nach der Durchführung der Variante 2 kein leeres Element und sie braucht daher keine Voxel-Aufteilung durchzuführen. Jedem Element wird gleich nach der ersten Interaktion ein HU-Wert zugeordnet. Das ist ein großer Vorteil bezüglich des Rechenaufwandes. Hier werden weniger Iterationen bzw. Berechnungsschritte benötigt. Die zweite Variante basiert auf einem komplett anderem Verfahren als die erste. Sie basiert auf der Suche nach dem nächstgelegenen Punkt. Diese Variante sucht die am nächsten liegenden Voxel-Schwerpunkte der Eckknoten und des Element-Schwerpunkts. Das bedeutet, dass sie 5 Voxel-Schwerpunkte für einen Tetraeder und 9 für einen Hexaeder sucht. Ob der Voxel-SP innerhalb, außerhalb, weit entfernt oder nah am Element ist, wird hier nicht berücksichtigt.

Dennoch hat die Variante 2 einen Nachteil gegenüber der Variante 1. Sie kann zu niedrige HU-Werte für die Elemente am Rand des Knochens liefern. Wenn ein Element am Rand des Knochens positioniert ist, kann der nächste liegende Voxel-SP einer seiner Eckknoten (oder seines Schwerpunkts) außerhalb der Knochengeometrie liegen. Knochen besitzen ein viel größeren HU-Wert als die neben umliegenden Gewebe und somit die Voxel-SP, die sich innerhalb deren Geometrie befinden. Abb 3.23 zeigt ein Beispiel dazu. Die weiße Linie stellt den Knochenrand und das gelbe Dreieck ein Element dar. Es ist deutlich zu erkennen, dass es einen großen Unterschied zwischen den HU-Wert innerhalb (blaue Kreise) und außerhalb (rote Kreise) der Knochengeometrie gibt.

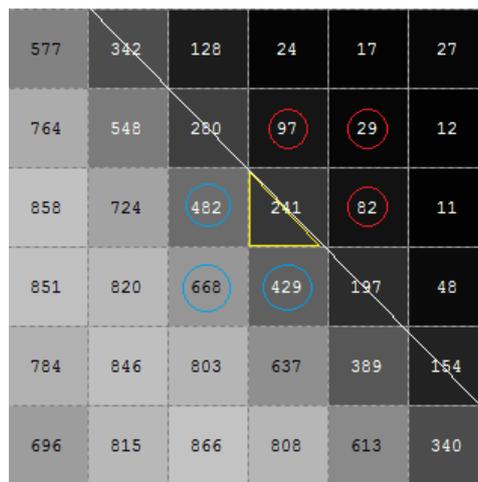


Abbildung 3.23.: Element am Rand des Knochens liegt zwischen großen und niedrigen HU-Werte

Abb. 3.24 zeigt den Unterschied zwischen der Variante „Convex Hull“ und der Variante „Nearest Point Search“ grafisch dar. Die von der zweiten Variante betrachteten

Voxel-Schwerpunkte werden grün dargestellt. Die blaue Voxel-SP wurden von der ersten Variante ermittelt.

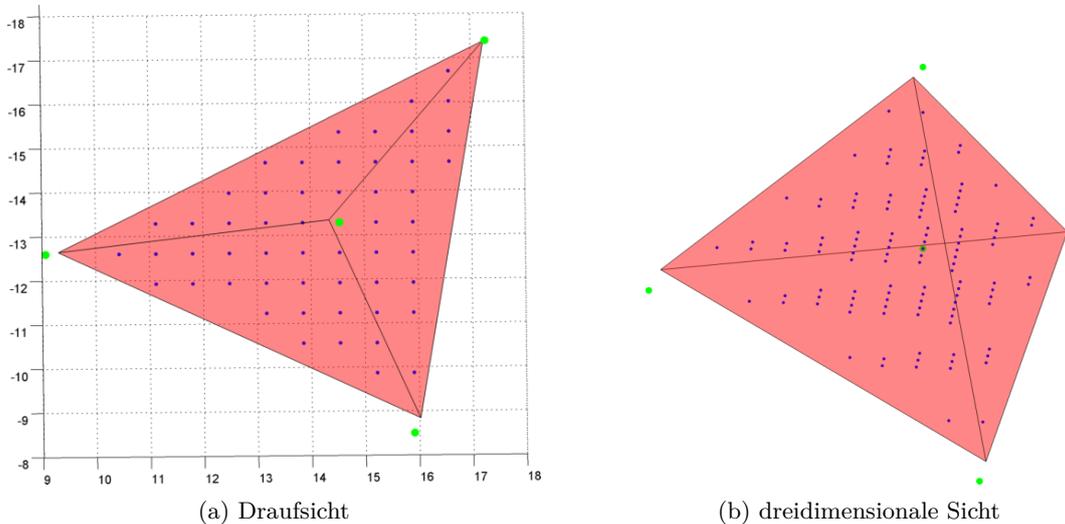


Abbildung 3.24.: V1 vs V2 - Unterschied der betrachteten Voxel-Schwerpunkte

### 3.4. Ableitung der elastischen Materialeigenschaften anhand der Knochendichte

Die Bestimmung der elastischen Materialeigenschaften von Knochen ist entscheidend für die Evaluierung des Frakturrisikos und der Knochenfragilität. Damit vertrauenswürdige Prognosen vorgegeben und das Treffen einer Entscheidung bei der Auswahl einer Rekonstruktionstechnik vereinfacht werden können, müssen von Simulationen zuverlässige Ergebnisse geliefert werden. Dies ist nur möglich, wenn realistische Materialeigenschaften verwendet werden. Jedoch ist die Korrelation zwischen Knochendichte und Materialeigenschaften von Knochen nicht so trivial wie die Korrelation zwischen HU-Wert (Abschwächungskoeffizient) und Knochendichte. Viele Veröffentlichung versuchen elastische Eigenschaften von Knochen und Knochendichte experimentell zu bestimmen und daraus einen mathematischen Zusammenhang abzuleiten. Die meisten der empfohlenen bzw. vorgeschlagenen Zusammenhänge unterscheiden sich untereinander (manchmal sogar sehr stark) und trotz Verfolgung verschiedener methodischer Herangehensweisen, kann der Grund der Abweichung nicht festgestellt werden [31].

Die Ableitung des Elastizitätsmoduls anhand der Knochendichte ist nicht Ziel bzw. Teil der vorliegenden Arbeit. Sie jedoch zu verstehen und ihre Relevanz zu kennen ist von großer Bedeutung für den Programmierer (der Autor), denn es ist wesentlich für die Auswertung und Zuverlässigkeit des Programms (*MatMap*).

Darüber hinaus hat die Bestimmung der elastischen Materialeigenschaften von Knochen keinen direkten Einfluss auf die Evaluierung und Bewertung der in dieser Arbeit entwickelten Algorithmen. Die Bewertung dieser Arbeit hängt direkt davon ab, ob *MatMap* ähnliche Ergebnisse wie Bonemat liefert. Und dafür reicht es, wenn die selben Parameter zur Ableitung des Elastizitätsmoduls in beiden Programmen verwendet werden. Im Folgenden werden die in dieser Arbeit verwendeten Beziehungen zur Ermittlung der elastischen Eigenschaften aus dem HU-Wert präsentiert.

Die Knochenmineraldichte anhand des HU-Werts wurde wie in [32] ermittelt:

$$\rho_{\text{QCT}} = 0.0007035 \cdot \text{HU} - 0.01185 \quad (3.2)$$

Die scheinbare Dichte von Knochen wurde, unter der Annahme der Folgenden aus adulten Knochen ermittelten Zusammenhänge, aus der Aschendichte abgeleitet [16]:

$$\rho_{\text{ASH}} = 0.8772 \cdot \rho_{\text{QCT}} + 0.07895 \quad (3.3)$$

$$\rho_{\text{APP}} = \frac{1}{0.6} \cdot \rho_{\text{ASH}} \quad (3.4)$$

Der Elastizitätsmodul wurde anschließend auf die in [33] vorgeschlagenen Potenzgesetz-Regression angesetzt:

$$E = 14664 \cdot \rho_{\text{ASH}}^{1.49} = 6850 \cdot \rho_{\text{APP}}^{1.49} \quad (3.5)$$

### 3.5. Komponentenerzeugung

Nach den zwei vorher beschriebenen Schritten besitzt jedes Element des Netzes seine eigenen Materialeigenschaften. Diese werden später in einer Simulations-Software (ANSYS) zur entsprechenden Festigkeitsberechnung genutzt. Doch wäre es ungünstig, alle diese Eigenschaften in der Software einzeln zu definieren. Bei einem feinen Netz würden über 100.000 neue Materialeigenschaften definiert werden müssen. Das wäre unpraktisch, rechenaufwändig und für das Ergebnis nicht entscheidend. Darüber hinaus weisen große Teile der Elemente ähnliche Materialeigenschaften auf. Wenn z.B. zwei Elemente ein E-Modul von 10 bzw. 10,1 MPa haben, reicht es für die Simulation, wenn beiden ein E-Modul von 10 MPa zugewiesen wird. Genau deswegen findet eine Aufteilung der Materialeigenschaften in sogenannte Komponenten

statt. Eine Komponente ist eine Gruppierung von quasi gleichwertigen Materialeigenschaften. Alle Elastizitätsmoduli die innerhalb eines bestimmten Bereiches liegen, gehören zu der Komponente dieses Bereichs.

Die Komponentenerzeugung kann entweder über eine vorgegebene Komponentenanzahl oder über einen vordefinierten E-Modul-Bereich ( $\Delta E$ ) durchgeführt werden. Das wird dem Benutzer überlassen.

Wenn eine Komponentenanzahl vorgegeben wird, sucht das Programm den niedrigsten und den größten E-Modul von Elementen auf und teilt den gesamten Bereich (durch  $E_{\text{MAX}} - E_{\text{MIN}}$  bestimmt) durch die Komponentenanzahl. Daraus wird der E-Modul-Bereich ermittelt. Wenn z.B.  $E_{\text{MAX}} = 100$  MPa,  $E_{\text{MIN}} = 10$  MPa und die gewünschte Komponentenanzahl von 18 eingegeben wurde, wird  $\Delta E = (100-10)/18 = 5$  MPa. Das bedeutet, dass aller 5 MPa eine Komponente erstellt wird (Komponente 18,  $10 < K18 \leq 15$ ; Komponente 17,  $15 < K17 \leq 20$  usw.). Die Erzeugung der Komponenten erfolgt von dem größten bis hin zu dem kleinsten E-Modul (Die erste Komponente wäre hier im Beispiel,  $95 < K1 \leq 100$  MPa).

Wenn ein bestimmter gewünschter E-Modul-Bereich eingegeben wird, teilt *MatMap* den gesamten Bereich diesmal durch diesen ( $\Delta E$ ) und ermittelt daraus die Komponentenanzahl. Z.B. wenn  $E_{\text{MAX}} = 100$  MPa,  $E_{\text{MIN}} = 10$  MPa und  $\Delta E = 9$  MPa, wird  $K_{\text{ANZAHL}} = (100-10)/9 = 10$ .

Unabhängig von der ausgewählten Methode, wird als nächstes im Prinzip die selbe Vorgehensweise realisiert. *MatMap* ordnet jedes Element einer Komponente zu. Jedes Element gehört zur Komponente, dessen  $\Delta E$  das E-Modul des Elements beinhaltet. Z.B. Komponente 17 hat einen E-Modul-Bereich von  $15 < K17 \leq 20$  MPa und Element X hat ein E-Modul von 16 MPa, dann gehört das Element X zur Komponente 3. Dann ermittelt *MatMap* den arithmetischen Mittelwert der E-Moduli innerhalb jeder Komponente und weist den zu dieser Komponente gehörenden Elementen den gemittelten E-Modul zu. Sprich alle Elemente einer Komponente besitzen die gleiche Materialeigenschaft.

Liste 3.2 und Liste 3.3 zeigen wie der Algorithmus der Komponentenerzeugung nach Komponentenanzahl bzw. E-Modul-Bereich funktioniert.

Diese Aufteilung der Materialeigenschaften in Komponenten kann theoretisch entweder vor oder nach der Konvertierung vom HU-Wert in E-Modul statt finden, wenn ein linearer Zusammenhang zwischen Dichte und Materialeigenschaften verwendet wird. Man könnte dementsprechend die Elemente auch in HU-Wert-Bereiche aufteilen, anstatt in E-Modul-Bereiche. Jedoch, wenn eine Potenzgesetz-Regression (Gl. 3.5) zum Einsatz kommt, macht dies einen Unterschied [13]. Es wurde in [13] festgestellt, dass eine Abschätzung des E-Moduls nach der Komponentenerzeugung (also

eine Aufteilung der HU-Werte) zu verschiedenen Ergebnissen führt und dass eine Erzeugung von Komponenten nach der Abschätzung des E-Moduls einige Vorteile schafft.

Code 3.2: Komponentenerzeugung nach Komponentenanzahl

```
1 % Ermittlung des gesamten Bereichs
2 max_E = max(E_app(:,2));
3 min_E = min(E_app(:,2));
4 range = single(max_E - min_E);
5
6 num_of_mat = material_number; % vorgegebene Komponentenanzahl
7 gap = range/num_of_mat; % ΔE
8
9 Material = zeros(num_of_mat,1);
10 num_ele_in_component = zeros(num_of_mat,1);
11
12 for mat = 1:num_of_mat % all material groups
13     pr = 1;
14     for id_mittelwert = 1: num_of_ele % all Mittelwert values
15         if (E_app(id_mittelwert,2) ≤ max_E - gap*(mat-1)) && ...
16             (E_app(id_mittelwert,2) > max_E - gap*mat)
17             mat_temp(mat,pr) = E_app(id_mittelwert,2);
18             ele_in_comp(mat,pr) = id_mittelwert;
19             %to later on know which element is within witch component
20             pr = pr+1;
21
22             elseif E_app(id_mittelwert,2) == min_E
23                 %% special case for min value add it to the
24                 min_v = id_mittelwert;
25             end
26     end
27
28     mat_temp(num_of_mat,nnz(mat_temp(num_of_mat,:))+1) = E_app(min_v,2);
29     ele_in_comp(num_of_mat,nnz(ele_in_comp(num_of_mat,:))+1) = min_v;
30     %to later on know which element is within witch component
31
32     for mat_2 = 1:num_of_mat
33         num_ele_in_component(mat_2,1) = nnz(mat_temp(mat_2,:));
34         Material(mat_2,1) = ...
35             sum(mat_temp(mat_2,:))/num_ele_in_component(mat_2,1);
36         % nnz -> Number of nonzero matrix elements
37     end
38 ...
```

Code 3.3: Komponentenerzeugung nach  $\Delta E$

```
1 % Ermittlung des gesamten Bereichs
2 max_E = max(E_app(:,2));
3 min_E = min(E_app(:,2));
4 range = single(max_E - min_E);
5
6 gap_E =  $\Delta E$ ; % gewünschter E-Modul-Bereich
7 num_of_mat = ceil(range/gap_E);
8 Material = zeros(num_of_mat,1);
9 num_ele_in_component = zeros(4,1);
10
11 for mat = 1:num_of_mat % all material groups
12     pr = 1;
13     for id_mittelwert = 1: num_ele % all Mittelwert values
14         if (E_app(id_mittelwert,2)  $\leq$  max_E - gap_E*(mat-1)) &&...
15             (E_app(id_mittelwert,2) > max_E - gap_E*mat)
16             mat_temp(mat,pr) = E_app(id_mittelwert,2);
17             ele_in_comp(mat,pr) = id_mittelwert; %to later on know ...
18                 which element is within witch component
19             pr = pr+1;
20         end
21     end
22 ...
```

### 3.6. Exportieren

Nach der Erzeugung der Komponenten, erstellt *MatMap* eine Input-Datei, die in ANSYS-Classic geladen werden kann. Eine Input-Datei muss alle Informationen beinhalten, damit ANSYS die Elemente mit ihren entsprechenden Materialeigenschaften generieren kann. Nachfolgend eine Auflistung der benötigten Informationen:

- 1- Knoten und ihre entsprechenden Koordinaten
- 2- Elemente und deren zugehörigen Knoten
- 3- Materialeigenschaften
- 4- Komponenten
- 5- Materialzuweisung

### 3. Die MATLAB-Routine

Die Nummerierung der Knoten und ihre entsprechenden Koordinaten, werden im ANSYS mit dem Befehl *N* geladen (siehe Abb. 3.25). Hier werden die von der NLIST eingelesenen Informationen wiedergegeben.

<pre>N,1,118.7117613,15.75632113,-1166.9889,,,,, N,2,118.7474365,14.70695491,-1163.891829,,,,, N,3,122.1905563,11.15175927,-1160.292131,,,,, N,4,106.65433,23.56198228,-1156.81696,,,,, N,5,114.607081,17.51324746,-1158.275436,,,,, N,6,109.3111821,24.84739293,-1169.645467,,,,, N,7,108.6043033,23.68937944,-1163.157521,,,,, N,8,114.050121,20.37915315,-1168.194671,,,,, N,9,118.665543,17.53937141,-1176.239267,,,,, N,10,113.9895231,21.10684935,-1172.83781,,,,, N,11,123.5510844,10.99576602,-1169.895492,,,,, N,12,98.62678831,32.0755988,-1167.726249,,,,, N,13,103.9487827,28.29478806,-1168.231551,,,,, N,14,94.19669062,33.30066625,-1162.830395,,,,, N,15,100.6137813,28.52989654,-1160.084248,,,,,</pre>	<p><b>Knoten-ID</b> <b>X-, Y- und Z-Koordinate</b></p>
--	--

Abbildung 3.25.: Erzeugung von Knoten in ANSYS-Mechanical-APDL

Zur Erzeugung von Elementen wurden die Befehle *en* und *emore* verwendet. Mit dem Befehl *en* sollen die Element-ID und die ersten 8 Knoten des Elements eingegeben werden. Wenn mehrere Knoten gebraucht werden, soll der Befehl *emore* verwendet werden. Vor der Erzeugung der Elemente soll der Elementtyp eingegeben werden, da es in ANSYS verschiedene gibt. Für den Elementtyp 186 sind insgesamt 20 Knoten einzugeben (siehe Abb. 3.26).

<pre>et,1,186,,,,, en,1,1802,1866,2360,2360,2359,2359,2359,2359 emore,14095,14490,2360,14099,2359,2359,2359,2359, emore,14098,14489,16829,16829,,,,, en,2,2238,2255,2254,2254,3003,3003,3003,3003 emore,16543,16598,2254,16542,3003,3003,3003,3003, emore,16546,16605,16601,16601,,,,, en,3,1801,1866,2360,2360,1802,1802,1802,1802 emore,14086,14490,2360,14088,1802,1802,1802,1802, emore,14083,14095,14099,14099,,,,, en,4,1962,2029,2365,2365,2028,2028,2028,2028 emore,14995,15382,2365,14996,2028,2028,2028,2028, emore,14994,15376,15378,15378,,,,, en,5,1866,1867,2360,2360,2359,2359,2359,2359 emore,14485,14499,2360,14490,2359,2359,2359,2359, emore,14489,14498,16829,16829,,,,, en,6,1739,2977,1800,1800,2796,2796,2796,2796 emore,13706,14082,1800,13696,2796,2796,2796,2796, emore,13704,19022,14079,14079,,,,, en,7,2254,2457,3003,3003,2255,2255,2255,2255 emore,16600,17410,3003,16601,2255,2255,2255,2255, emore,16598,16604,16605,16605,,,,,</pre>	<p><b>Elementtyp</b> <b>Element-ID</b> <b>Knoten des Elements</b></p>
--	---

Abbildung 3.26.: Erzeugung von Elementen in ANSYS-Mechanical-APDL

Die Materialeigenschaften werden mit dem Befehl *MP* (englisch: material properties) in ANSYS importiert. Hier werden E-Modul (MPa), Poissonszahl und Dichte

### 3. Die MATLAB-Routine

(g/cm<sup>3</sup>) definiert (siehe Abb. 3.27). Eine konstante Poissonsanzahl von 0.3 wurde, sowohl von *MatMap*, als auch von Bonemat exportiert. *MatMap* gibt die scheinbare Dichte ( $\rho_{APP}$ ) in der Ausgabe ein.

MP, EX, 1, 13140.06, , , , , ,	Material-ID
MP, NUXY, 1, 0.3, , , , , ,	E-Modul
MP, DENS, 1, 1.54835421251155, , , , , ,	Poissonzahl
MP, EX, 2, 12689.238, , , , , ,	Dichte
MP, NUXY, 2, 0.3, , , , , ,	
MP, DENS, 2, 1.51249731312518, , , , , ,	
MP, EX, 3, 12435.461, , , , , ,	
MP, NUXY, 3, 0.3, , , , , ,	
MP, DENS, 3, 1.49212859733469, , , , , ,	
MP, EX, 4, 12191.039, , , , , ,	
MP, NUXY, 4, 0.3, , , , , ,	
MP, DENS, 4, 1.47238110085557, , , , , ,	
MP, EX, 5, 12056.887, , , , , ,	
MP, NUXY, 5, 0.3, , , , , ,	
MP, DENS, 5, 1.46148730204302, , , , , ,	
MP, EX, 6, 11888.182, , , , , ,	
MP, NUXY, 6, 0.3, , , , , ,	
MP, DENS, 6, 1.44773086915518, , , , , ,	
MP, EX, 7, 11870.883, , , , , ,	
MP, NUXY, 7, 0.3, , , , , ,	
MP, DENS, 7, 1.4463166683268, , , , , ,	

Abbildung 3.27.: Definition von Materialeigenschaften in ANSYS-Mechanical-APDL

Zur Erzeugung der Komponenten wurde der Befehl *CMBLOCK* verwendet. Dieser Befehl weist einer Komponente die Elemente zu und gibt dieser eine ID.

CMBLOCK, Component_47, ELEM, 16, , , , ,	Komponente-ID
(8110), , , , , ,	
1379, 2120, 2574, 2584, 2676, 3092, 3112, 3487, ,	Elementanzahl
3879, 3881, 6285, 7028, 7706, 8891, 9372, 9451, ,	
CMBLOCK, Component_48, ELEM, 6, , , , ,	Elemente
(8110), , , , , ,	
2045, 2346, 4248, 7792, 7818, 9111, , , ,	
CMBLOCK, Component_49, ELEM, 14, , , , ,	
(8110), , , , , ,	
471, 2137, 2422, 2568, 2635, 3049, 3425, 5576, ,	
7875, 8270, 9223, 9682, 9830, 9987, , , ,	
CMBLOCK, Component_50, ELEM, 13, , , , ,	
(8110), , , , , ,	
467, 1161, 1824, 2280, 2612, 3217, 4563, 4631, ,	
4903, 6003, 6442, 7001, 8460, , , , ,	
CMBLOCK, Component_51, ELEM, 12, , , , ,	
(8110), , , , , ,	
470, 2301, 2812, 3231, 3470, 3499, 5072, 7502, ,	
7853, 8518, 9471, 10442, , , , ,	
CMBLOCK, Component_52, ELEM, 7, , , , ,	
(8110), , , , , ,	
59, 3875, 4585, 5935, 7492, 8037, 9586, , ,	

Abbildung 3.28.: Zuweisung von Elementen zu seiner Komponente

Um einer Komponente bestimmte Materialeigenschaften zuzuweisen, müssen die Befehle aus Abb. 3.29 in dieser Reihenfolge eingegeben werden.

```

allsel,,,,,,,,,
cmsel,all,,,,,,,,|
cmsel,s,Component_47,,,,,,,,,
emodif,all,mat,36,,,,,,,,,
allsel,,,,,,,,,
cmsel,all,,,,,,,,,
cmsel,s,Component_48,,,,,,,,,
emodif,all,mat,37,,,,,,,,,
allsel,,,,,,,,,
cmsel,all,,,,,,,,,
cmsel,s,Component_49,,,,,,,,,
emodif,all,mat,38,,,,,,,,,
allsel,,,,,,,,,
cmsel,all,,,,,,,,,
cmsel,s,Component_50,,,,,,,,,
emodif,all,mat,39,,,,,,,,,
allsel,,,,,,,,,
cmsel,all,,,,,,,,,
cmsel,s,Component_51,,,,,,,,,
emodif,all,mat,40,,,,,,,,,
allsel,,,,,,,,,
cmsel,all,,,,,,,,,
cmsel,s,Component_52,,,,,,,,,
emodif,all,mat,41,,,,,,,,,

```

Abbildung 3.29.: Zuweisung von Materialeigenschaften zur Komponente

Es ist auch wichtig zu erwähnen, dass vor all diesen Befehlen der Befehl */PREP7* kommen muss. Mit diesem Befehl geht ANSYS in den „model creation“ Vorprozessor hinein. Ohne diesen werden die andere Befehle von ANSYS nicht wahrgenommen (unter der ANSYS-Hilfe können die wichtigsten Informationen zur APDL-Programmierung gefunden werden).

*MatMap* bietet dem Benutzer an, die Möglichkeit ein „Offset“ bei der Erstellung der Datei einzufügen. Wenn mehrere Bauteile bzw. Knochen in ANSYS-Classice geladen werden müssen, dürfen sie die selben Knoten- und Elementen-ID nicht besitzen. Sprich wenn das FE-Netz des ersten Knochens 100 Elemente hat (Element-ID von 1 bis 100), darf das Netz des zweiten Knochens ein Element-ID zwischen 1 und 100 nicht besitzen. Das erste Element des zweiten Netzes muss eine Element-ID von mindestens 101 besitzen, weil sonst interpretiert ANSYS das erste Element des zweiten Netzes wie das erste Element des ersten Netzes. Dieses darf nicht passieren. Daher wurde im GUI (siehe Abb. 3.1) von *MatMaps* ein Bedienfeld zur Eingabe des Offsets für die Knoten-, Element-, Materialeigenschaft- und Komponenten-ID erzeugt. Um zu wissen, welcher Offset für welche ID benötigt wird muss die vorherige erzeugte Datei bzw. die IDs der vorher in ANSYS geladen Knochen beachtet werden (Bei der Erzeugung der Input-Datei des ersten Knochens ist kein Offset notwendig. Es wird also ein Offset von 0 eingesetzt).

Das Exportieren der Input-Datei ist der letzte Schritt von *MatMap*. Mit all diesen Informationen ist die Input-Datei in der Lage, in ANSYS ein Model zu erzeugen. Es muss beachtet werden, dass MATLAB eigentlich eine Text-Datei generiert und dass

diese unbenannt (.inp am Ende einfügen) werden muss, damit sie in eine Input-Datei konvertiert wird.

Abb. 3.30 stellt die komplette Funktionsweise von *MatMap* dar.

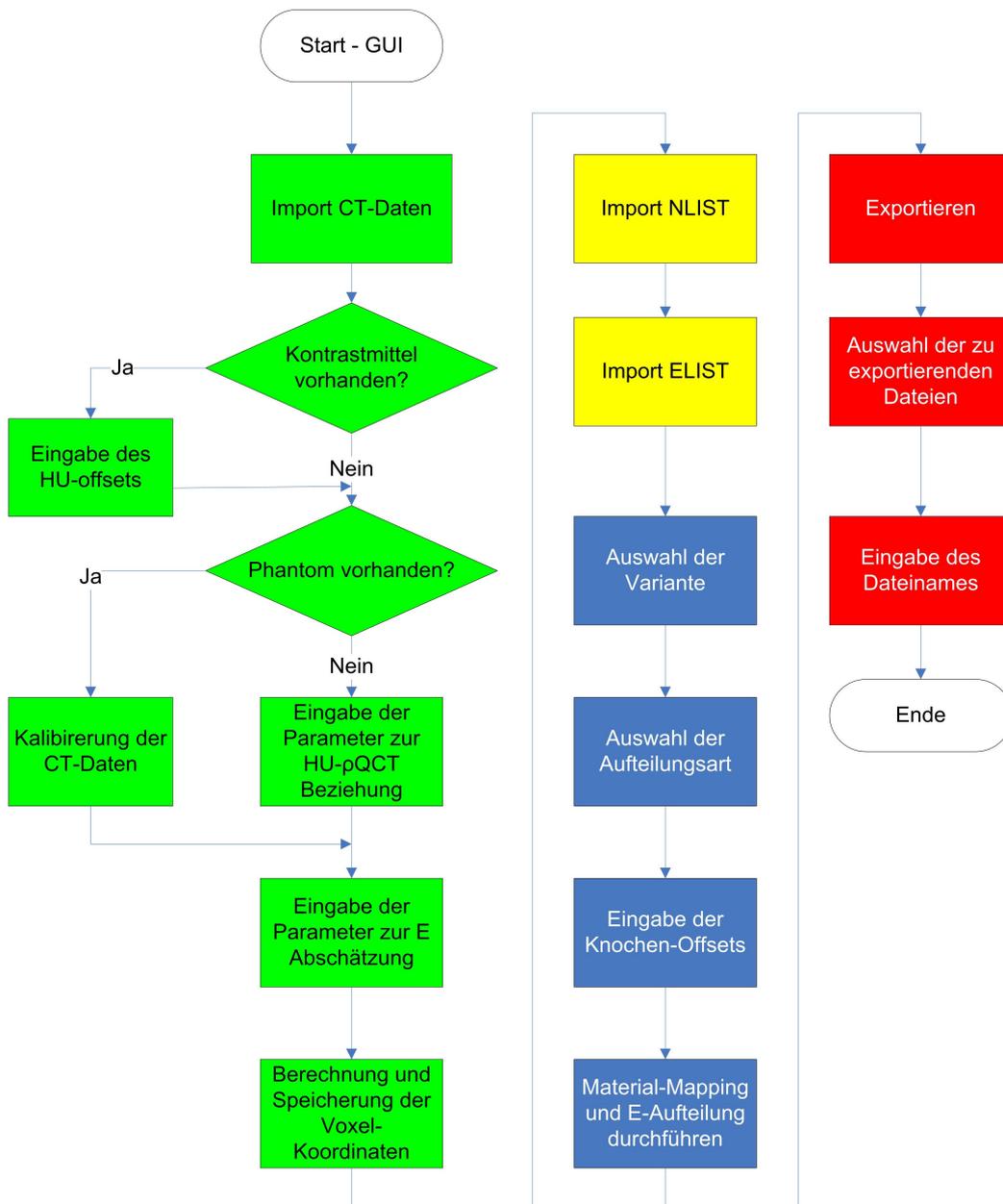


Abbildung 3.30.: *MatMaps* Ablaufplan

## 4. Material und Methoden

Ein komplettes Becken und ein Wirbelkörper einer siebenundsiebzig jährigen Frau mit Osteoporose wurden in dieser Arbeit zur Evaluierung und Validierung von *Mat-Map* untersucht. Abb. 4.1 stellt die knöchernen Bestandteile eines Beckens dar.

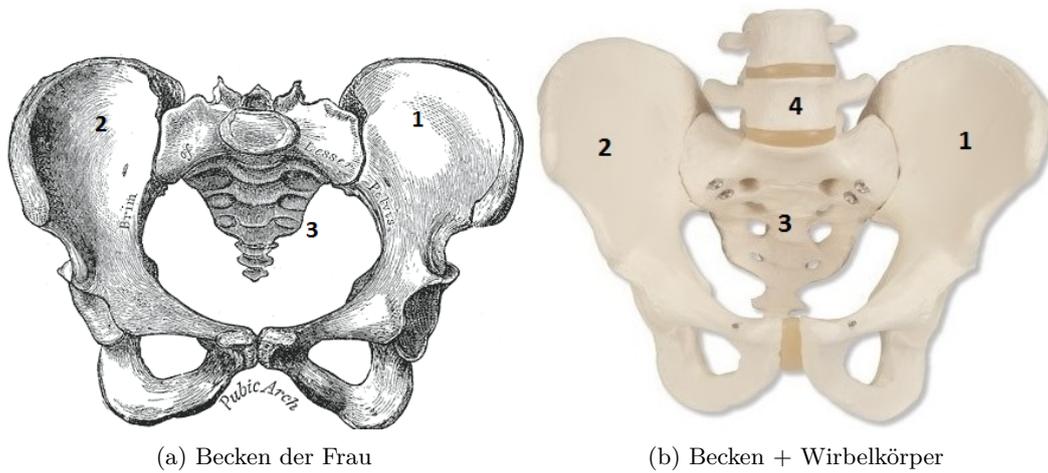


Abbildung 4.1.: Beckenknochen [4]

Die Positionsnummer eins und zwei stellen dabei die Darmbeine (latein: Os ilium) dar. Punkt drei stellt das Kreuzbein (Os sacrum) und Punkt vier den Lendenwirbel L5 dar. Unterschiedliche Netze mit verschiedenen Elementgrößen wurden für jeden Knochen in ANSYS-Workbench erstellt (siehe Anhang A.1). Tabelle 4.1 stellt die erzeugten Netze und ihren Eigenschaften dar.

Tabelle 4.1.: Eigenschaften der erzeugten Netze

Knochen	Elementgröße	Anzahl Knoten	Anzahl Elemente
Os ilium links	12 mm	19380	10785
	5 mm	56692	34791
	2 mm	557350	385547
Os ilium rechts	12 mm	19161	10648
	5 mm	55896	34217
	2 mm	553336	382397
Os sacrum	12 mm	37582	21678
	5 mm	60751	36651
	1,5 mm	961715	673566
L5	12 mm	18099	10284
	5 mm	22795	13225
	2 mm	105358	69881

Diese Netze wurden anschließend mit dem Tool „Mechanical APDL“ von Workbench in ANSYS Classic geladen. Dort wurden sie als NLIST und ELIST bzw. als cdb-Datei gespeichert (siehe Anhang). Die erzeugten Dateien wurden in MATLAB importiert und ihre entsprechenden Input-Dateien wurden generiert. Auch in Bonemat wurde für jedes Netz eine Input-Datei anhand der Anleitung<sup>4</sup> generiert. Sowohl bei Bonemat, als auch bei *MatMap* wurden die Komponenten mit einem  $\Delta E$  von 100 MPa erzeugt.

Alle erzeugten Input-Dateien (von Bonemat und von *MatMap*) wurden anschließend wieder in ANSYS-Classic geladen. Jetzt besitzt jedes Element des FE-Netzes (aus CT-Daten mittels Material-Mapping) abgeleitete Materialeigenschaften. Zum Vergleich der Dateien wurden in ANSYS-Classic E- und MPLISTEN (material properties list) erzeugt und verglichen.

Die ELIST enthält alle generierten Elemente mit ihren ID's und der dazugehörigen Material-ID (siehe Abb. 4.2). Mittels der MPLIST können zu jeder Material-ID die elastischen Eigenschaften (E-Modul, Querkontraktionszahl und Dichte) ausgelesen werden.

LIST ALL SELECTED ELEMENTS. (LIST NODES)										Element-ID			
ELEM	MAT	TYP	REL	ESY	SEC	NODES				Material-ID			
1	78	1	1	0	1	948	950	2357	2357	2765	2765	2765	2765
						8900	8918	2357	8905	2765	2765	2765	2765
						8906	8920	16669	16669				
2	108	1	1	0	1	670	767	2356	2356	2348	2348	2348	2348
						7208	7771	2356	7210	2348	2348	2348	2348
						7209	7770	16610	16610				
3	101	1	1	0	1	736	838	2341	2341	737	737	737	737
						7589	8232	2341	7593	737	737	737	737
						7587	7598	7600	7600				
4	101	1	1	0	1	736	838	862	862	2341	2341	2341	2341
						7589	8229	862	7590	2341	2341	2341	2341
						7593	8232	8377	8377				
5	95	1	1	0	1	527	2788	661	661	2359	2359	2359	2359
						6399	7161	661	6392	2359	2359	2359	2359
						6398	16674	7160	7160				
6	106	1	1	0	1	660	2788	2348	2348	2356	2356	2356	2356
						7155	16612	2348	7153	2356	2356	2356	2356
						7154	16667	16610	16610				
7	96	1	1	0	1	326	2387	2950	2950	2926	2926	2926	2926
						5159	16854	2950	5161	2926	2926	2926	2926
						5160	16853	19099	19099				

Abbildung 4.2.: Beispiel einer ELIST

Eine MPLIST stellt die Komponenten (mittels Material-ID) und ihre zugehörigen E-Moduli dar (siehe Abb. 4.3).

<sup>4</sup><http://www.bonemat.org/>

#### 4. Material und Methoden

---

EVALUATE MATERIAL PROPERTIES FOR MATERIALS		1 TO	125 IN INCREMENTS OF	1
MATERIAL NUMBER = <u>EX = 14341.</u> NUXY = 0.30000 DENS = 1.6419	1	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = <u>EX = 12933.</u> NUXY = 0.30000 DENS = 1.5319	2	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = <u>EX = 12375.</u> NUXY = 0.30000 DENS = 1.4873	3	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = <u>EX = 12295.</u> NUXY = 0.30000 DENS = 1.4808	4	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = <u>EX = 11999.</u> NUXY = 0.30000 DENS = 1.4568	5	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = <u>EX = 11917.</u> NUXY = 0.30000 DENS = 1.4501	6	EVALUATED AT TEMPERATURE OF	0.0000	
MATERIAL NUMBER = EX = 11790. NUXY = 0.30000	7	EVALUATED AT TEMPERATURE OF	0.0000	

**Material-ID**  
**E-Modul**

Abbildung 4.3.: Beispiel einer MPLIST

## 5. Ergebnisse und Diskussion

### 5.1. Ergebnisse

Um die Funktionsfähigkeit und die Aussagekraft der Ergebnisse von *MatMap* besser einschätzen zu können, wird ein Vergleich zu Bonemat angestellt.

Hierfür werden MATLAB-Algorithmen zum Einlesen und Vergleichen der Ergebnisse (ELIST + MPLIST der erzeugten Input-Dateien) generiert. Nachfolgend werden Grafiken und Tabellen aufgeführt, welche einen Vergleich der ermittelten Daten zulassen.

Die E-Moduli-Tabellen zeigen den mittleren, den größten und den niedrigsten E-Modul für alle Knochen und Vernetzungskonfigurationen (siehe Tabellen 5.1, 5.4, 5.7 und 5.10).

Die E-Modul-Differenz-Tabellen verdeutlichen die Unterschiede zwischen Bonemat und den beiden Varianten bzw. zwischen V1 und V2. Sie stellen die mittlere E-Modul-Differenz bzw. -Abweichung zwischen allen Elementen von Bonemat und V1 bzw. V2 und auch zwischen V1 und V2 dar (siehe Tabellen 5.2, 5.5, 5.8 und 5.11).

Die Tabellen 5.3, 5.6, 5.9 und 5.12 stellen den prozentuellen Anteil der Elementen, welche einen E-Modul-Wert aufweisen, der sich um 10, 20 oder 50% des E-Modul-Wertes von Bonemat unterscheidet. Wenn z.B. das Element X einen E-Modul von 100 MPa besitzt, stellt die Spalte „über 20%“ den Prozentanteil der Elemente von V1 bzw. V2 dar, welche einen E-Modul über 120 MPa oder unter 80 MPa haben. Sprich der Prozentanteil der Elemente, die sich um mindestens 20% vom Bonemat E-Modul unterscheiden.

5.1.1. Os ilium links

Tabelle 5.1.: E-Moduli-Os ilium links [MPa]

	Elementgröße	Bonemat	V1	V2
mittlerer E-Modul	12 mm	3025.562	2866.65	1972.60
	5 mm	2786.391	2672.56	1951.65
	2 mm	2479.586	2430.23	2283.76
größter E-Modul	12 mm	13411	13140	9239
	5 mm	14274	14655	11119
	2 mm	16237	16825	15346
niedrigster E-Modul	12 mm	419.76	154.35	200.23
	5 mm	339.47	43.43	120.66
	2 mm	397.95	1.54	10.76

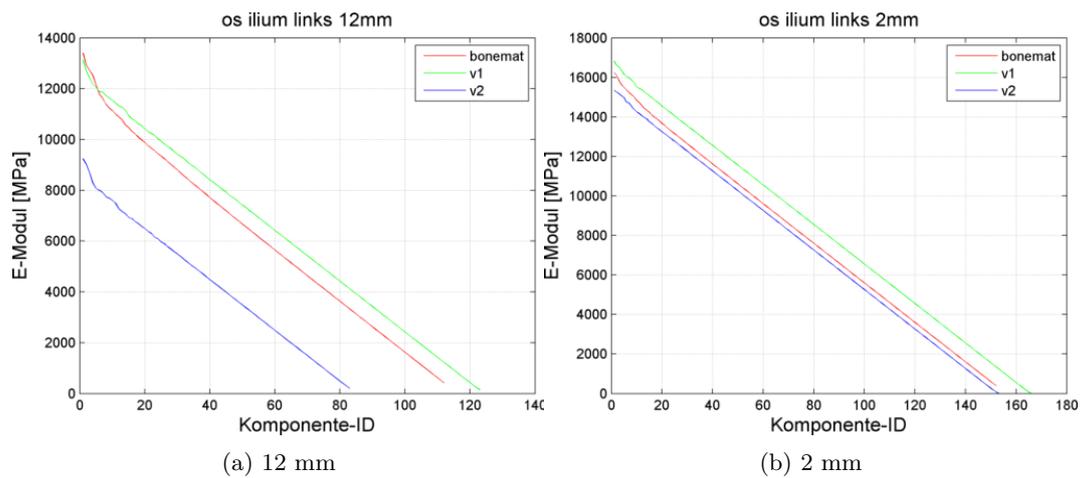


Abbildung 5.1.: E-Modul vs Komponente-ID - Os ilium links

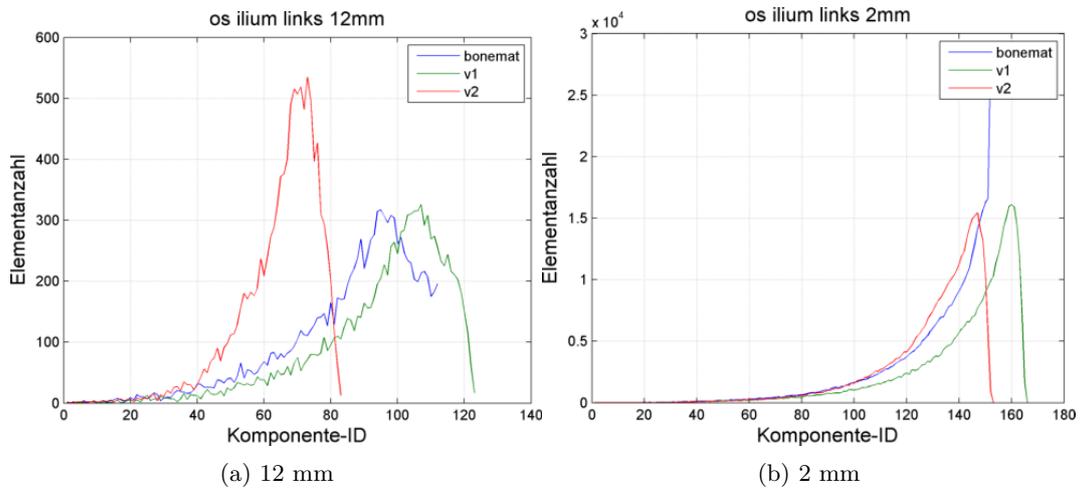


Abbildung 5.2.: Elementanzahl vs Komponente-ID - Os ilium links

Tabelle 5.2.: E-Modul-Abweichung - Os ilium links [MPa]

	Elementgröße	Bonemat vs V1	Bonemat vs V2	V1 vs V2
mittleres $\Delta E$	12 mm	249.072	1118.439	988.663
	5 mm	185.231	956.766	887.029
	2 mm	280.783	277.211	391.527

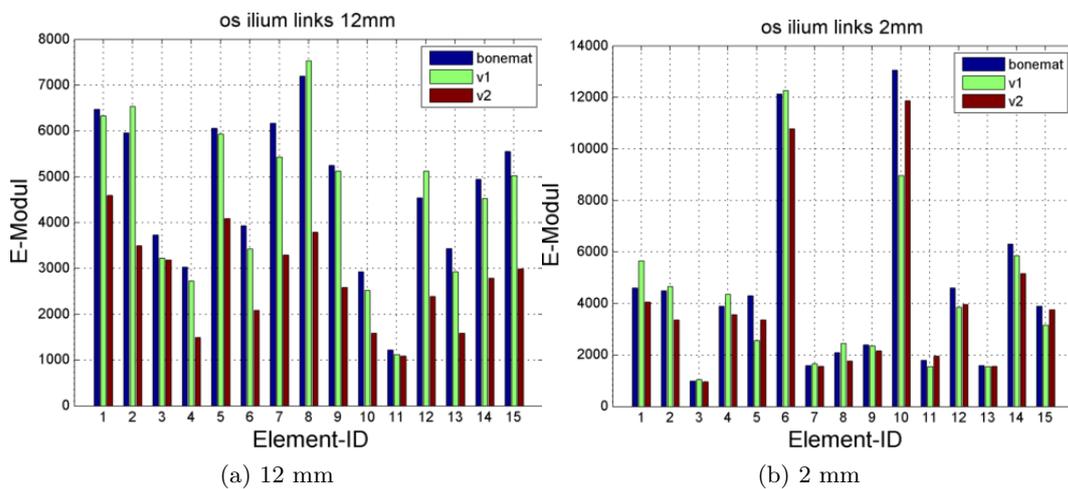


Abbildung 5.3.: E-Modul vs Element-ID für die ersten 15 Elemente- Os ilium links

Tabelle 5.3.: Anteil der Elemente von V1 und V2, welche einen E-Modul-Wert aufweisen, der sich um 10, 20 bzw. 50% des E-Moduls-Wertes von Bonemat unterscheidet - Os ilium links

	Elementgröße	über 10%	über 20%	über 50%
V1	12 mm	0.3503	0.0804	0.0028
	5 mm	0.2607	0.0915	0.0111
	2 mm	0.4631	0.2279	0.0323
V2	12 mm	0.8803	0.7382	0.1785
	5 mm	0.8439	0.6699	0.1340
	2 mm	0.4580	0.1752	0.0193

### 5.1.2. Os ilium rechts

Tabelle 5.4.: E-Moduli - Os ilium rechts [MPa]

	Elementgröße	Bonemat	V1	V2
mittlerer E-Modul	12 mm	2826.483	2667.34	1794.59
	5 mm	2687.038	2569.28	1878.62
	2 mm	2448.375	2394.04	2267.72
größter E-Modul	12 mm	15158	14341	8800.1
	5 mm	16068	15649	13198
	2 mm	17649	17980	17165
niedrigster E-Modul	12 mm	412.45	1e-06	17.49
	5 mm	404.95	18.257	17.49
	2 mm	411.97	50.711	23.16

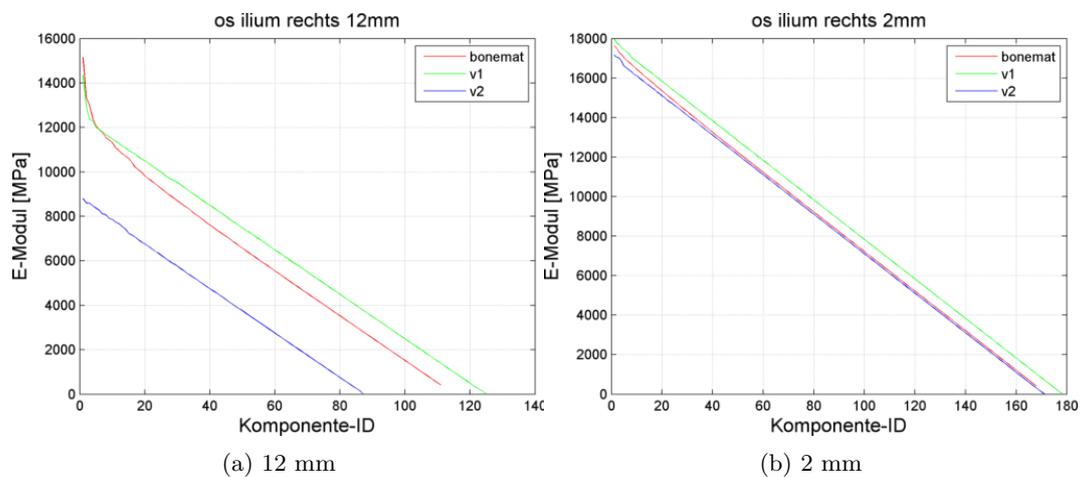


Abbildung 5.4.: E-Modul vs Komponente-ID - Os ilium rechts

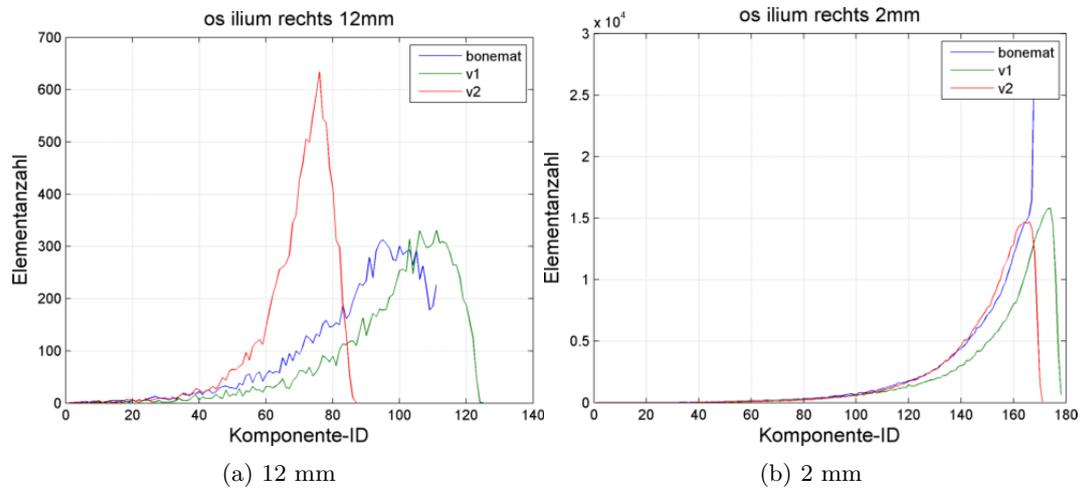


Abbildung 5.5.: Elementanzahl vs Komponente-ID - Os ilium rechts

Tabelle 5.5.: E-Modul-Abweichung - Os ilium rechts [MPa]

	Elementgröße	Bonemat vs V1	Bonemat vs V2	V1 vs V2
mittleres $\Delta E$	12 mm	235.374	1087.192	957.766
	5 mm	181.596	928.591	858.621
	2 mm	280.032	257.595	380.019

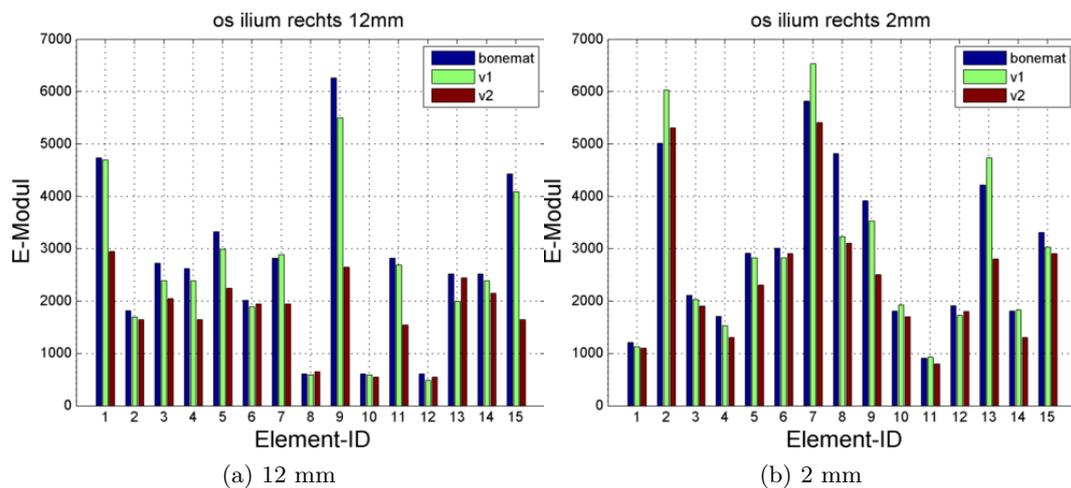


Abbildung 5.6.: E-Modul vs Element-ID für die ersten 15 Elemente- Os ilium rechts

Tabelle 5.6.: Anteil der Elemente von V1 und V2, welche einen E-Modul-Wert aufweisen, der sich um 10, 20 bzw. 50% des E-Moduls-Wertes von Bonemat unterscheidet - Os ilium rechts

	Elementgröße	über 10%	über 20%	über 50%
V1	12 mm	0.3613	0.0841	0.0066
	5 mm	0.2957	0.0887	0.0143
	2 mm	0.5016	0.1973	0.0261
V2	12 mm	0.8858	0.7515	0.2114
	5 mm	0.8422	0.6719	0.1560
	2 mm	0.4792	0.1817	0.0064

### 5.1.3. Os sacrum

Tabelle 5.7.: E-Moduli - Os sacrum [MPa]

	Elementgröße	Bonemat	V1	V2
mittlerer E-Modul	12 mm	1832.889	1732.265	1418.068
	5 mm	1740.282	1640.580	1373.463
	1,5 mm	1571.075	1483.878	1441.608
größter E-Modul	12 mm	12314	12240	11679
	5 mm	11887	12096	11764
	1,5 mm	12830	13022	12848
niedrigster E-Modul	12 mm	367.01	19.387	52.869
	5 mm	409.21	24.741	49.015
	1,5 mm	430.77	0.083	34.197

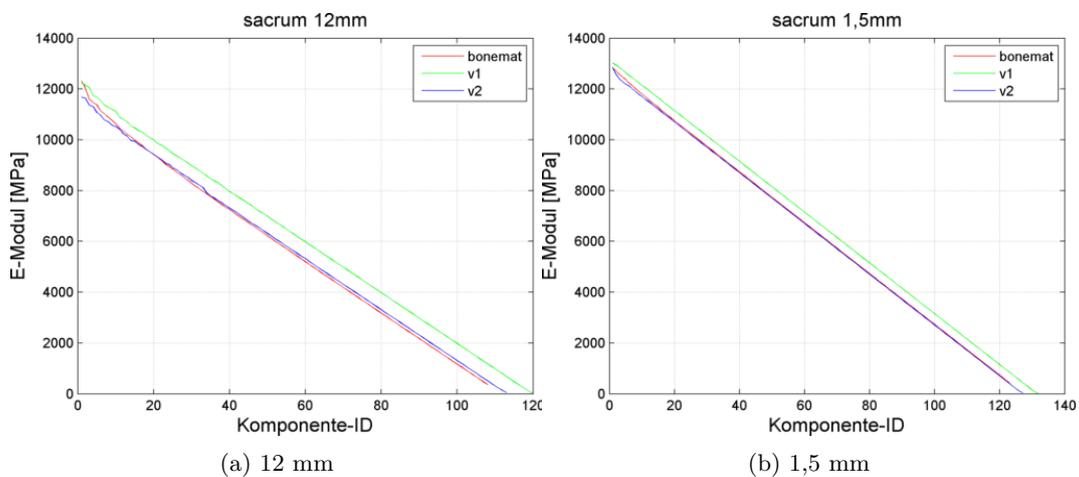


Abbildung 5.7.: E-Modul vs Komponente-ID - Os sacrum

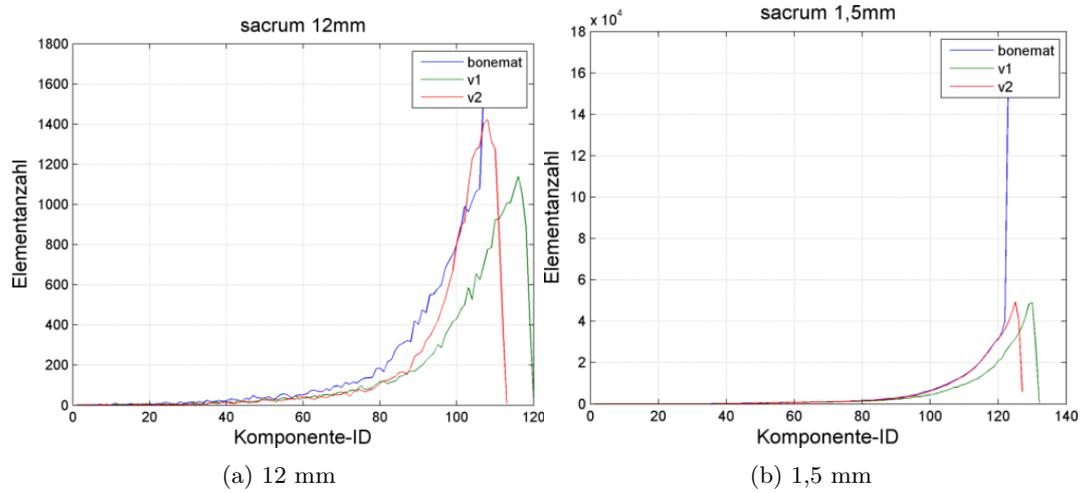


Abbildung 5.8.: Elementanzahl vs Komponente-ID - Os sacrum

Tabelle 5.8.: E-Modul-Abweichung - Os sacrum [MPa]

	Elementgröße	Bonemat vs V1	Bonemat vs V2	V1 vs V2
mittleres $\Delta E$	12 mm	181.246	469.188	439.134
	5 mm	166.376	425.479	395.509
	1,5 mm	190.772	163.368	181.786

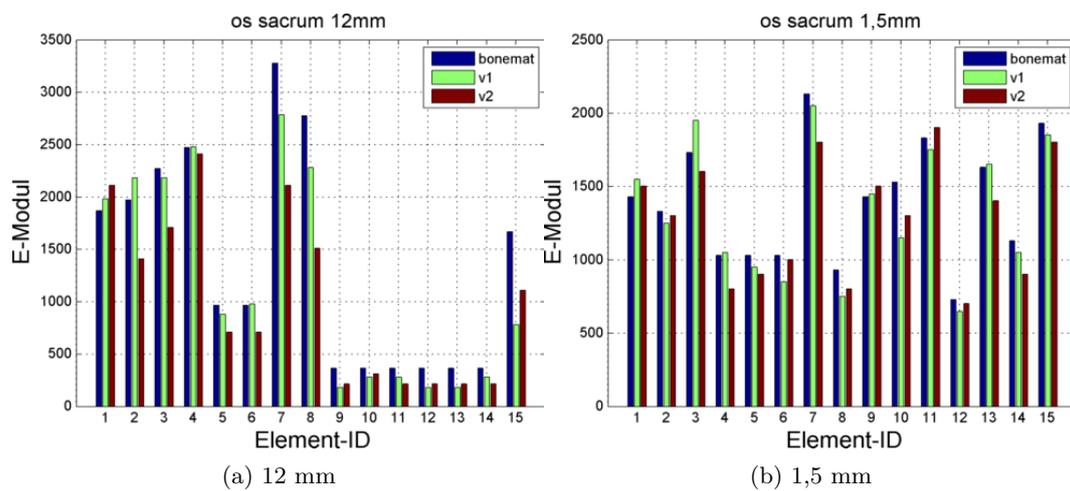


Abbildung 5.9.: E-Modul vs Element-ID für die ersten 15 Elemente- Os sacrum

Tabelle 5.9.: Anteil der Elemente von V1 und V2, welche einen E-Modul-Wert aufweisen, der sich um 10, 20 bzw. 50% des E-Moduls-Wertes von Bonemat unterscheidet - Os sacrum

	Elementgröße	über 10%	über 20%	über 50%
V1	12 mm	0.4902	0.2288	0.0389
	5 mm	0.4683	0.2506	0.0942
	1,5 mm	0.5562	0.3040	0.1261
V2	12 mm	0.7940	0.5870	0.1033
	5 mm	0.8060	0.5968	0.0862
	1,5 mm	0.4841	0.2954	0.1427

### 5.1.4. L5

Tabelle 5.10.: E-Moduli - L5 [MPa]

	Elementgröße	Bonemat	V1	V2
mittlerer E-Modul	12 mm	3771.049	3715.146	3449.007
	5 mm	3457.252	3401.444	3153.738
	2 mm	2827.786	2781.921	2699.802
größter E-Modul	12 mm	13382	13811	13068
	5 mm	13609	13780	13172
	2 mm	13479	13842	13309
niedrigster E-Modul	12 mm	387.70	199.480	321.16
	5 mm	428.89	93.596	256.67
	2 mm	349.33	50.316	133.65

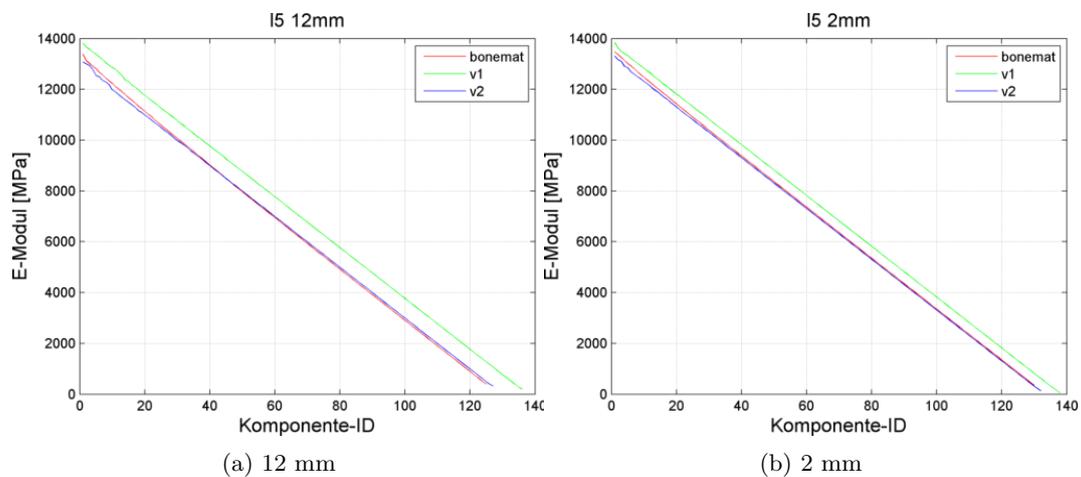


Abbildung 5.10.: E-Modul vs Komponente-ID - L5

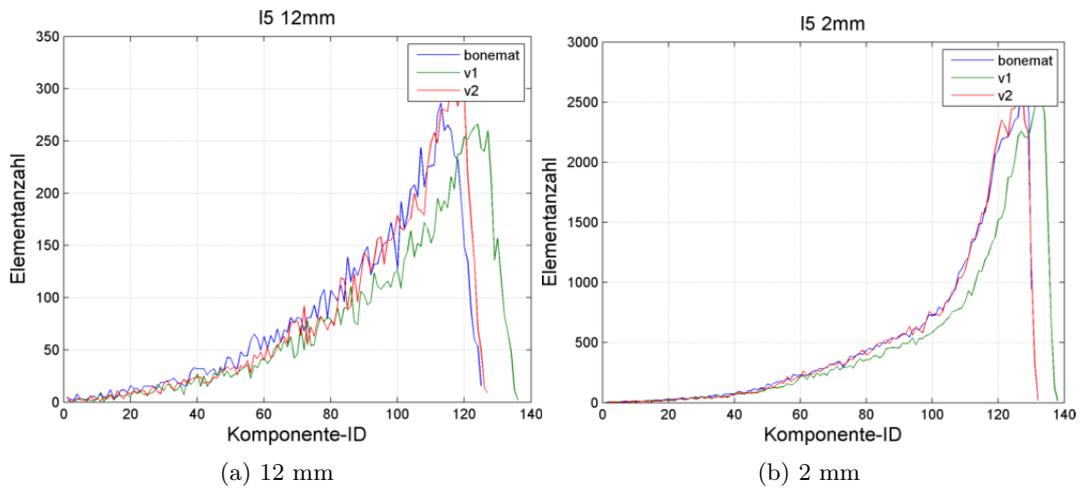


Abbildung 5.11.: Elementanzahl vs Komponente-ID - L5

Tabelle 5.11.: E-Modul-Abweichung - L5 [MPa]

	Elementgröße	Bonemat vs V1	Bonemat vs V2	V1 vs V2
mittleres $\Delta E$	12 mm	277.705	471.773	552.475
	5 mm	248.877	441.923	511.054
	2 mm	230.627	205.121	304.742

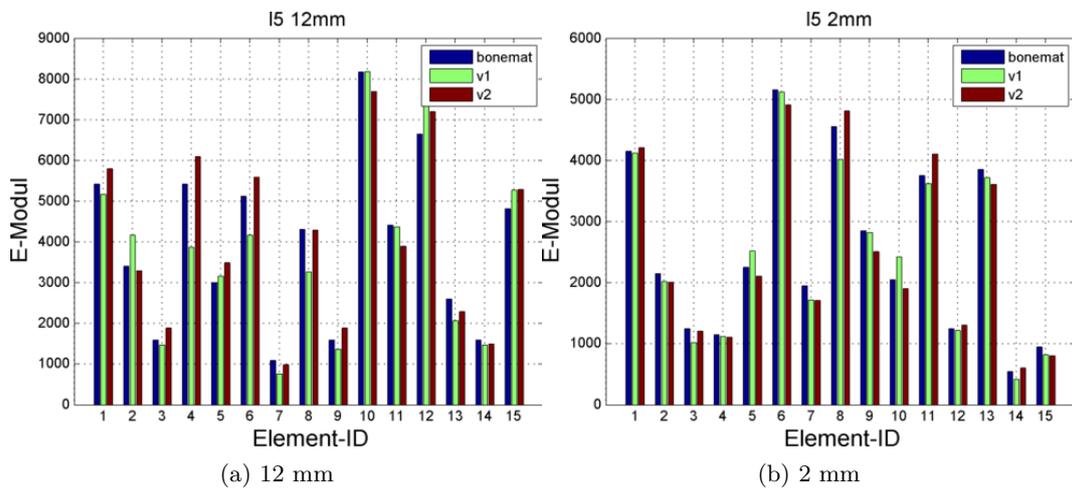


Abbildung 5.12.: E-Modul vs Element-ID für die ersten 15 Elemente- L5

Tabelle 5.12.: Anteil der Elemente von V1 und V2, welche einen E-Modul-Wert aufweisen, der sich um 10, 20 bzw. 50% des E-Moduls-Wertes von Bonemat unterscheidet - L5

	Elementgröße	über 10%	über 20%	über 50%
V1	12 mm	0.3012	0.1091	0.0096
	5 mm	0.2884	0.1014	0.0083
	2 mm	0.3504	0.1403	0.0095
V2	12 mm	0.5367	0.2609	0.0180
	5 mm	0.5492	0.2619	0.0157
	2 mm	0.3018	0.0790	0.0010

## 5.2. Diskussion

Zur Diskussion der Ergebnisse werden nur die Daten des Knochens „Os ilium links“ betrachtet, da die anderen Ergebnisse zu den selben Schlussfolgerungen führen.

Die Tabelle 5.1 zeigt, dass sich die Variante 1 sehr stark an Bonemat annähert. Der mittlere und der größte E-Modul der Variante 1 unterscheiden sich kaum und zeigen keine signifikante Abhängigkeit von der Elementgröße. Die Variante 2 ist andererseits sehr stark von der Elementgröße abhängig. Je feiner die Vernetzung, desto ähnlicher sind die Werte für den mittleren und den größten E-Modul.

Der niedrigste E-Modul-Wert der beiden Varianten von *MatMap* liegt jedoch weit unter dem niedrigsten Wert von Bonemat. Der Grund, warum der niedrigste E-Modul Wert von Bonemat sehr hoch kalkuliert wird, ist unbekannt, aber es ist plausibel, dass die niedrigsten Werte der Elemente der Varianten 1 und 2 bei null liegen, da es andere Gewebe bzw. Wasser oder Luft innerhalb eines Knochens geben kann. Ein E-Modul von z.B 10 MPa in *MatMap* (niedrigster Wert von V2 in der Tabelle 5.1) entspricht umgerechnet einem Wert von etwa -100 HU. Dieser entspricht Fett und kann durchaus bei einem Knochen mit Osteoporose vorhanden sein [34; 35]. Ein anderer Grund für die kleineren E-Moduli ist die Umrechnung von HU-Wert in E-Modul. Die Gleichungen des Abschnitts 3.4 sind nur für Knochen ermittelt worden. Sie werden jedoch trotzdem für die anderen Gewebe bzw. Wasser und Luft extrapoliert.

Die Grafiken der Abb. 5.2 zeigen die Korrelation zwischen den Komponenten und ihrer Elementanzahl. In diesen ist deutlich zu erkennen, dass die letzte Komponente von Bonemat (die Komponente mit dem niedrigsten E-Modul) eine ziemlich große Anzahl von Elementen, im Vergleich zu den letzten Komponente von V1 und V2 (insbesondere in der Abb. 5.2 (b)), aufweist. Die letzte Komponente der Varianten 1

und 2 hat im Gegensatz zu Bonemat sehr wenige Elemente. Doch wenn man sich die Tabelle 5.1 anschaut, sieht man, dass der niedrigste E-Modul von Bonemat weit über dem niedrigsten E-Modul der *MatMap*-Varianten liegt (z.B. bei einer Elementgröße von 2 mm:  $397 \gg 10 > 1$  MPa). Dies bedeutet, dass die letzten Komponenten der Varianten 1 und 2 niedrigere E-Moduli zuweisen und dass diese Komponenten nur wenige Elemente haben (siehe Abb. 5.2). Es scheint, dass Bonemat allen Komponenten unterhalb eines gewissen E-Modul den selben E-Modul zuweist und deswegen die letzte Komponente eine höhere Elementanzahl beinhaltet.

Es kann mithilfe der Tabelle 5.2 ebenfalls festgestellt werden, dass V2 genauer (angenommen Bonemat liefert die genauesten Ergebnisse) wird, wenn die Elementgröße minimiert wird (Abb. 5.3 verdeutlicht diese Schlussfolgerung). Auf der anderen Seite ändert sich die Abweichung zwischen V1 und Bonemat nicht sehr stark. Hier sieht man, dass V1 Werte liefert, die sich Bonemat annähern. Ein mittleres  $\Delta E$  von 280 MPa (Tabelle 5.2, 2 mm) weist keine große Differenz auf, da der mittlere E-Modul bei 2500 MPa (Tabelle 5.1) und der größte E-Modul bei 16000 MPa liegt. Die E-Modul-Abweichung ist aber nicht für jedes Element so gering. Es sind auch große bzw. sehr große Differenzen vorhanden. Die Tabelle 5.3 zeigt jedoch, dass es sich bei diesen zu großen Abweichungen um Ausnahmen handelt.

Die Tabelle 5.3 stellt die prozentualen Anteile der Elemente von V1 bzw. V2 dar, deren E-Modul-Werte sich leicht (10%), nicht stark (20%) und stark (50%) von dem gezielten E-Modul (dem E-Modul, den Bonemat zu demselben Element zugewiesen hat) unterscheiden. Es ist deutlich erkennbar, dass sich V2 an V1 und somit auch an Bonemat annähert, je feiner die Vernetzung ist. Bei einer Elementgröße von 12 mm weisen 73% der Elemente von V2 einen E-Modul auf, der sich um mindestens 20% von Bonemat unterscheidet. Doch bei einer Elementgröße von 2 mm weisen nur 17% der Elemente von V2 einen solchen Unterschied auf.

Darüber hinaus weist V1 quasi konstante Werte (Tabellen 5.3 und 5.12) im Vergleich zu Bonemat auf; sprich V1 hängt nicht bzw. nicht stark von der Elementgröße ab.

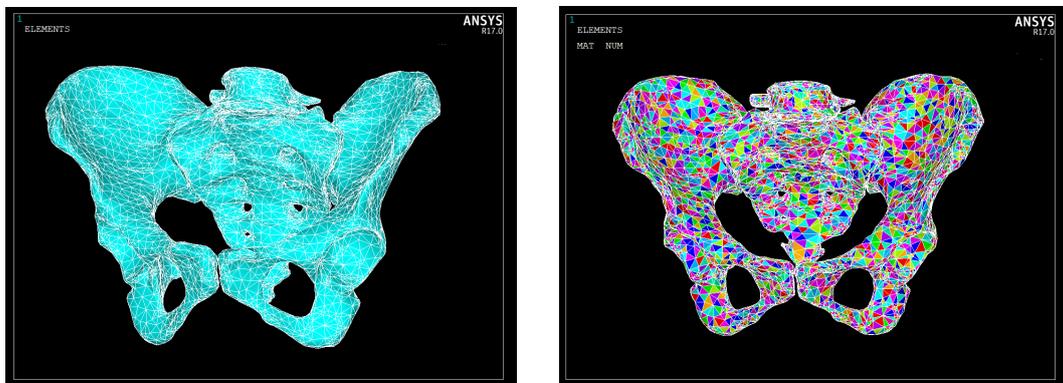
Dennoch lässt sich der Tabelle 5.3 entnehmen, dass bei einer feineren Vernetzung; beide Varianten den selben Elementen ähnliche E-Moduli wie Bonemat zuweisen. Bei einer Elementgröße von 2 mm z.B. besitzt kaum ein Element von V1 oder V2 einen E-Modul-Wert, der sich stark (über 50%) von den Bonemat-Werten unterscheidet. Die Mehrheit der Elemente von V1 und V2 (etwa 80%) besitzen einen ähnlichen E-Modul ( $\Delta E < 20\%$ ) wie die Bonemat-Elemente. Die untenstehende Tabelle verdeutlicht diese Aussage.

Tabelle 5.13.: Anteil der Elemente von V1 und V2, welche einen E-Modul-Wert aufweisen, der sich um 100, 200, 500 bzw. 1000 MPa von Bonemat unterscheidet - Os ilium links

	Elementgröße	über 100 MPa	über 200 MPa	über 500 MPa	über 1000 MPa
V1	12 mm	0.6588	0.4373	0.1425	0.0245
	5 mm	0.6069	0.3109	0.07044	0.0109
	2 mm	0.6419	0.4037	0.1579	0.0509
V2	12 mm	0.9131	0.8360	0.6557	0.4213
	5 mm	0.8482	0.7338	0.5553	0.3499
	2 mm	0.6425	0.4142	0.1726	0.04956

Die Tabelle 5.13 bestätigt, dass sich nur eine Minderheit der Elemente von V1 und V2 stark von Bonemat unterscheidet.

Zur Überprüfung der Funktionsfähigkeit von *MatMap* für die Generierung von Assemblies, wurden Input-Dateien mit entsprechenden Offsets (für Knoten-, Elemente-, Material- und Komponente-ID) generiert und anschließend in ANSYS-Classic geladen (siehe Abb. 5.13).



(a) Darstellung der Elemente

(b) Darstellung der Materialien

Abbildung 5.13.: Importieren des kompletten Beckens und des Wirbelkörpers in ANSYS-Classic

Das Bild 5.13 (b) zeigt die in *MatMap* durchgeführte Materialzuordnung. Hierfür wurden Offsets in der Input-Dateien addiert, die mit einer Elementgröße von 12 mm mittels V1 generiert wurden, da dieses Verfahren sich als das schnellste bewies.

Für ein maximales Maß an Flexibilität, wurde eine Lösung zum Import dieser Daten in ANSYS-Workbench untersucht (siehe Abb. 5.14). Dazu muss das in ANSYS-Classic geladene Modell als .cdb gespeichert (siehe Abb. A.2 im Anhang) und anschließend in ANSYS-Workbench mit dem „Finite Element Modeler“-Tool geladen

werden. Aus unbekanntenen Gründen scheint ANSYS-Workbench die Materialzuordnung der Elemente nicht wahrzunehmen (auch wenn alle vorhandenen Materialien in den „Technischen Daten“ der Workbench angezeigt werden).

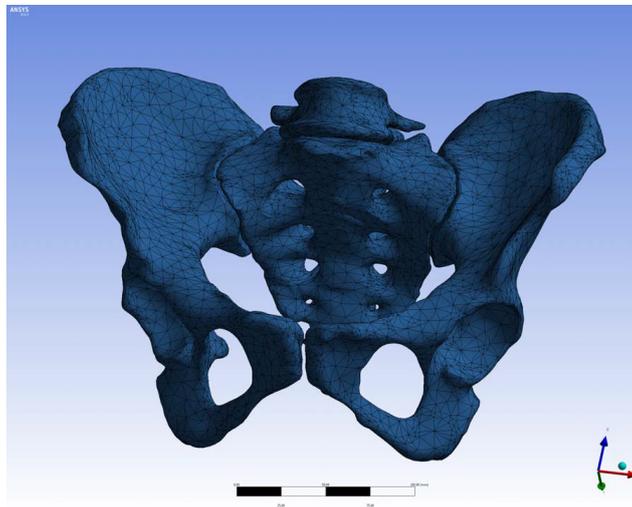


Abbildung 5.14.: Becken in ANSYS-Workbench

Damit ANSYS-Workbench jedem Element seine entsprechenden Materialeigenschaften zuweisen kann, müssen APDL-Befehle eingefügt werden, in denen eine komplette Beschreibung der Materialzuordnung (Erzeugung von Materialien, Komponenten bzw. Zuweisung von Materialien zu den Komponenten) beschrieben wird (siehe Abb. 5.15).

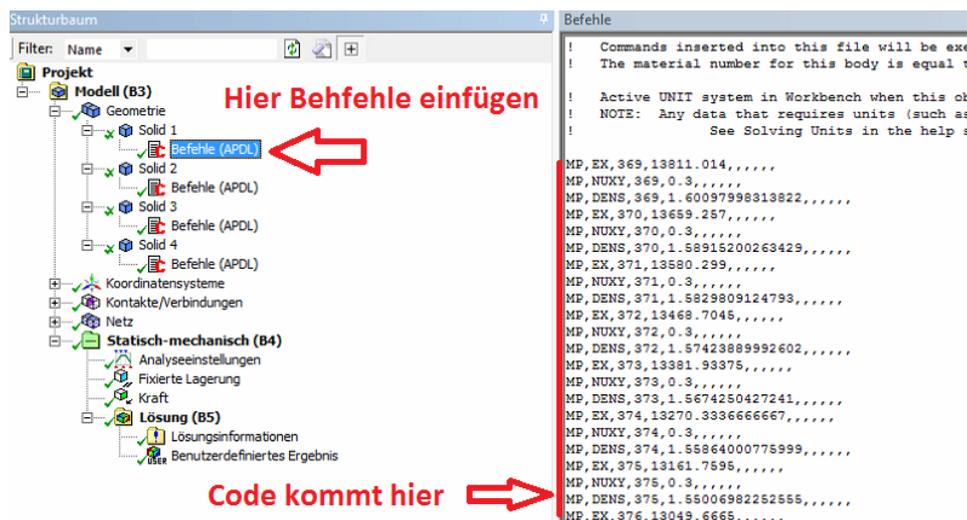


Abbildung 5.15.: Materialzuordnung in ANSYS-Workbench

Der Großteil der einzugebenden Befehle ist identisch zum Code der bereits erzeugten Input-Datei. Dieser muss nur kopiert und in ANSYS-Workbench in der richtigen

Reihenfolge (genauso wie beim Exportieren der Input-Datei: Erzeugung der Materialeigenschaften -> Erzeugung der Komponenten -> Zuweisung der Materialeigenschaften zu den Komponenten) unter jedem Körper bzw. Solid (siehe Abb. 5.15) eingefügt werden. Um dem Benutzer dennoch die Arbeit zu erleichtern, bietet *MatMap* die Möglichkeit, diese Befehle (APDL-Dateien) für ANSYS-Workbench einzeln zu exportieren.

Abb. 5.16 zeigt die Trennung der Knochenmaterialien über die Funktion PNUMMAT in Workbench.

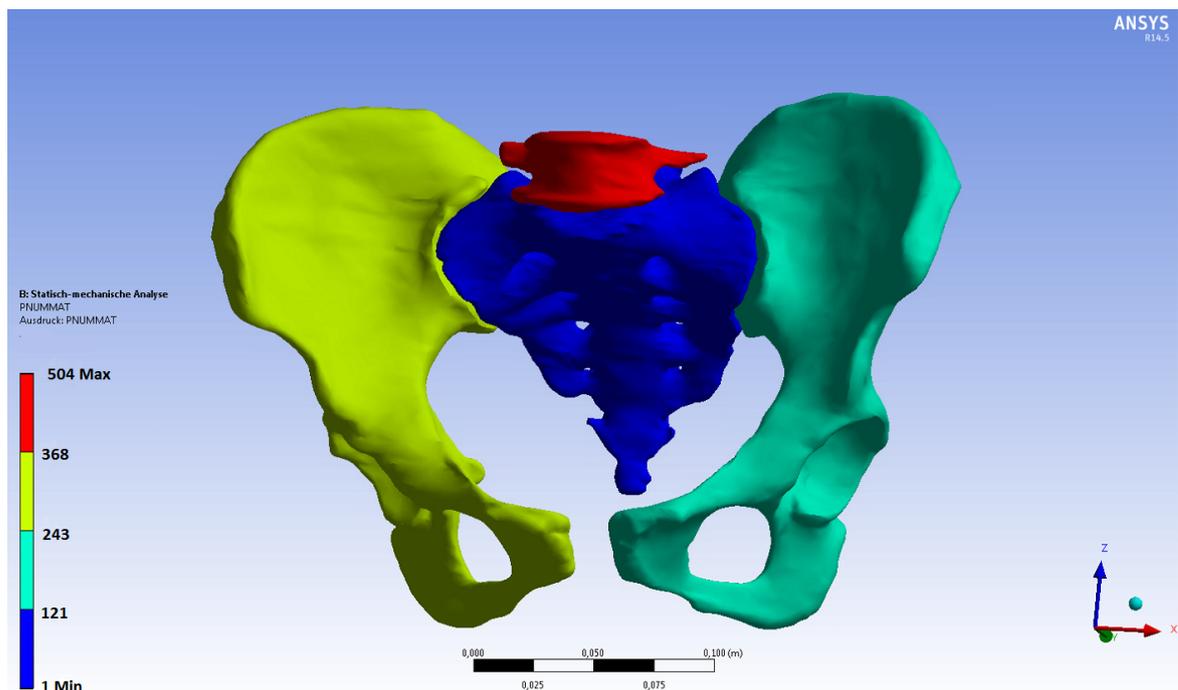


Abbildung 5.16.: Trennung der Knochenmaterialien über die Funktion PNUMMAT in ANSYS-Workbench

Aus der Abb. 5.16 geht hervor, dass z.B. den Elementen von Os sacrum die Material-IDs 1 bis 121 zugewiesen wurden und dass den Material-IDs der anderen Knochen einen Offset addiert wurde.

### 5.2.1. Betrachtung der Rechenzeit

Abschließend wurde eine empirische Betrachtung der Rechenzeit der in dieser Arbeit entwickelten Varianten durchgeführt. Das Ziel dieser Betrachtung ist es, dem Benutzer bzw. dem Leser einen Einblick in der Rechenzeit von *MatMap* zu geben. Die Tabelle 5.14 stellt die Rechenzeit der Variante 1 und der Variante 2 von *MatMap* für alle analysierten Knochen in Minuten dar.

Tabelle 5.14.: Rechenzeit der entwickelten Varianten [min]

Knochen	Elementgröße	Anzahl Elemente	Rechenzeit V1	Rechenzeit V2
Os ilium links	12 mm	10785	38 (p)	62 (p)
	5 mm	34791	50 (*)	145 (*)
	2 mm	385547	297 (s)	1198 (s)
Os ilium rechts	12 mm	10648	45 (b/p)	75 (s)
	5 mm	34217	67 (p)	145 (p)
	2 mm	382397	1252 (p)	2735 (p)
Os sacrum	12 mm	21678	52 (b)	87 (*)
	5 mm	36651	53 (s)	157 (s)
	1,5 mm	673566	528 (p)	1721 (p)
L5	12 mm	10284	22 (b/p)	29 (b)
	5 mm	13225	16 (p)	37 (p)
	2 mm	69881	62 (p)	139 (p)

Idealerweise sollte eine solche Betrachtung auf demselben Rechner durchgeführt werden. Aus Zeitgründen wurden in dieser Arbeit aber sechs verschiedene Rechner genutzt. Die Abkürzungen „b“, „p“ und „s“ stehen für Büro-Rechner, Pool-Rechner und Schnell-Rechner (\* heißt, dass der Rechner unbekannt ist bzw. nicht dokumentiert wurde). Tabelle 5.15 stellt die grundlegenden Eigenschaften der genutzten Rechner dar.

Tabelle 5.15.: Eigenschaften der Computer

Rechner	Prozessor	Ram	Windows-Edition V1
s	Intel(R) Xeon(R) 2.93 GHz (2 Prozessoren)	64 GB	Windows 7 Professional
b*	Intel(R) Xeon(R) 2.8 GHz	16 GB	Windows 7 Professional
p	Intel(R) Xeon(R) 2.8 GHz	16 GB	Windows 7 Professional

\*Obwohl die Rechner vom Pool und vom Büro eine identische Hardware-Konfiguration besitzen, haben sich die Pool-Rechner als etwas schneller bewiesen. Dies liegt vermutlich daran, dass auf dem Büro-Rechner mehr Software installiert war.

Die Tabelle 5.14 zeigt deutlich, dass die Rechenzeit erwartungsgemäß sehr stark von der Elementanzahl abhängt. Es ist auch zu erwähnen, dass *MatMap* im Vergleich zu Bonemat ziemlich langsam rechnet. Innerhalb von wenigen Minuten (max. 10) führt Bonemat ein Material-Mapping durch und erstellt eine Input-Datei für eine feine Vernetzung (z.B. 2 mm).

Aus der Tabelle 5.14 lässt sich ebenso feststellen, dass die Variante 1 eine niedrigere Rechenzeit aufweist, was nicht zu erwarten war. Erst nach Überprüfung des Programms wurde entdeckt, dass es zusätzlich unnötige Befehle in der Schleife der Variante 2 gab. Diese hatten zwar keinen Einfluss auf das Ergebnis des Material-Mappings, haben diese jedoch verlangsamt.

Nach einer Aktualisierung des Algorithmus von V2 wurde eine neue „Zeituntersuchung“ für Os ilium rechts durchgeführt. Die Tabelle 5.16 stellt das Ergebnis dieser Untersuchung dar.

Tabelle 5.16.: Rechenzeit der V1 mit der korrigierten V2 [min]

Knochen	Elementgröße	Anzahl Elemente	Rechenzeit V1	Rechenzeit V2
Os ilium rechts	12 mm	10648	45.40 (b/p)	54.09 (s)
	5 mm	34217	67.30 (p)	129.86 (p)
	2 mm	382397	1252.60 (p)	762.80 (p)

Bevor man aus der Tabelle 5.16 Schlussfolgerungen zieht, soll hier erwähnt werden, dass die Variante 1 eine xyz-Begrenzung besitzt, während die Variante 2 nur eine z-Begrenzung besitzt. Es wurde vermutet, dass V2 schneller als V1 sei (wegen der Komplexität der eingesetzten Funktionen) und daher keine ganze xyz-Begrenzung benötigt, sondern nur eine Begrenzung der zu überprüfenden Schichten (z-Begrenzung). Trotz des Vorteils, dass bei den xyz-Begrenzungen nur wenige Elemente überprüft werden müssen, kostet sie viel Zeit. Es dauert relativ lange, bis das Programm die in der Nähe eines Elementes liegenden Voxel-SP bestimmt. Da V2 angeblich nicht sehr zeitaufwendig sei, wurde bei ihr keine xyz-Begrenzung eingesetzt.

Aus der Tabelle 5.16 geht hervor, dass V2 bei der feinen Vernetzung viel schneller ist. Jedoch hat sich bei einer Elementgröße von 12 und 5 mm die erste Variante schneller bewiesen. Der Grund dafür ist, dass die xyz-Begrenzung für viele Elemente durchgeführt werden musste, sodass sich die Rechenzeit sehr stark erhöhte.

V1 verwendet viel Zeit für die Bestimmung der in der Nähe des Elements liegenden Voxel-SP und wenig Zeit für die Ermittlung der konvexen Hülle. V2 benötigt andererseits wenig Zeit für die Bestimmung der daneben liegenden Schichten (z-Begrenzung), aber relativ lange für die Suche nach den nächstgelegenen Punkten. Eine genauere Betrachtung bzw. Untersuchung des Rechenaufwands wird hier für zukünftige Arbeiten empfohlen.

## 6. Zusammenfassung und Ausblick

*MatMap*, die hier vorgestellte MATLAB-Routine, führt ein Material-Mapping an einem FE-Netz eines Knochens durch und erstellt eine Input-Datei, welche den Elementen des Netzes Materialeigenschaften anhand von CT-Daten zuweist. Doch wie kann überprüft werden, ob diese Materialeigenschaften realistisch sind? Wie kann überprüft werden, ob die Knochen auch in der Realität die zugewiesenen elastischen Materialeigenschaften besitzen und sich so wie in der Wirklichkeit in der späteren Simulation verhalten werden? Es sind Fragen, die sich viele Forscher und Wissenschaftler stellen und die ihr Interesse wecken; Fragen, die noch nicht abschließen beantwortet sind. Viel Forschungsaufwand muss in diesem Bereich noch betrieben werden. Bislang ist nur teilweise bekannt, wie genau sich HU-Werte und Knochendichte bzw. Knochendichte und elastische Materialeigenschaften miteinander verknüpfen lassen. Doch eine große Anzahl von Veröffentlichungen zeigt, dass eine Abschätzung der elastischen Eigenschaften eines Knochens (anhand seiner Dichte bzw. CT-Daten) durch Anwendung von empirisch ermittelten mathematischen Beziehungen durchaus nachvollziehbar ist.

Eine Studie aus dem Jahr 1998 [36] entwickelte eine Software (Bonemat), die aus CT-Daten elastische Materialeigenschaften von Knochen ableitet und diese den Elementen eines FE-Netzes anhand ihrer Position zuweist, sodass jedes Element einen realistischen E-Modul besitzt. Bonemat wurde Ende 1998 [36] der wissenschaftlichen Gemeinschaft vorgestellt und wird seitdem stets weiterentwickelt [37; 13]. Viele Veröffentlichungen dieses Bereichs zitieren Bonemat und verwenden es als Referenz. Es lässt sich daher annehmen, dass Bonemat (in der Größenordnung) vertrauenswürdige Ergebnisse generiert. Genau deswegen wurde *MatMap* mit Bonemat verglichen.

Die in dieser Arbeit vorgestellten Ergebnisse beweisen, dass die „Convex Hull“ Variante nicht signifikant von der Elementgröße abhängt und mit Bonemat vergleichbare Ergebnisse liefert. Die „Nearest Point Search“ Variante zeigte sich andererseits sehr stark von der Elementgröße abhängig und lieferte - im Vergleich zu Bonemat - signifikant unterschiedliche Ergebnisse. Trotzdem hat sich diese Variante bei sehr feinen Vernetzungen bewährt.

Es ist wichtig hervorzuheben, dass *MatMap* innerhalb von fünf Monaten komplett geschrieben wurde. Trotz der zufriedenstellenden Ergebnisse, gibt es an den Algorithmen noch einiges zu verbessern bzw. zu optimieren. Abschließend werden Optimierungsvorschläge bzw. Punkte von *MatMap*, die evtl. verbessert bzw. untersucht werden könnten, aufgelistet:

- Codeoptimierung:

Eine Optimierung und eine genauere Zeitbetrachtung des Codes soll durchgeführt werden. Programmierfehler bzw. -redundanzen, die dem Autor unbemerkt geblieben sind und zur Erhöhung der Rechenzeit führen können, sollen gesucht werden. Desweiteren wird empfohlen, den genaueren Einfluss von jeder Funktion auf die Rechenzeit zu bestimmen.

- Einfluss der Komponentenerzeugung:

Eine Untersuchung des Einflusses bzw. des Unterschiedes der Erzeugung der Komponenten (über eine vorgegebene Komponentenanzahl und über ein vordefiniertes  $\Delta E$ ) auf das Endergebnis ist empfohlen.

- Einfluss der Kalibrierung:

Trotz der Entwicklung eines Algorithmus zur Kalibrierung der CT-Daten, standen bei der Durchführung dieser Arbeit keine Informationen über die Knochenmineralstandards zur Verfügung. Es wird empfohlen, den Einfluss der Kalibrierung auf das Endergebnis zu überprüfen und die daraus ermittelten Parameter mit denen in der Literatur [31] zu vergleichen. Auch interessant wäre die Erforschung eines anderen Zusammenhangs (nicht nur linear) zur Berechnung der Knochenmineraldichte (evtl. nicht-linearen, segmentweise etc. (siehe Abb. 3.14)).

- Vorstellung neuer Material-Mapping Strategien:

Anstatt nur die Voxel-Schwerpunkte zu betrachten, ist es empfehlenswert, das komplette Voxel-Volumen zu betrachten (solle dies die Rechenzeit nicht drastisch erhöhen). Es wird auch vorgeschlagen, bei der ersten Variante das Voxel solange aufzuteilen, bis sich einer der neuen Voxel-SP innerhalb der konvexen Hülle des Elements befindet. Desweiteren könnte der Einfluss der Verwendung aller Elementknoten (statt nur die Eckknoten) zur Konstruktion der konvexen Hülle untersucht werden.

- E-Modul Ableitung der nicht knöchernen Gewebe:

Es wird für nicht knöchernen Gewebe (z.B. wenn der HU-Wert kleiner als 0 ist) empfohlen, eine andere Gleichung zur Ableitung des E-Moduls zu verwenden.

## 7. Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Abschlussarbeit unterstützt und motiviert haben sowie bei all denjenigen, die mein Studium in Merseburg ermöglicht und zu einer wunderbaren Erfahrung gemacht haben.

Zuerst gebührt mein Dank Herrn M. Eng. Mehle, der mich zur Bearbeitung dieses Thema eingeladen und meine Bachelorarbeit sehr nah betreut und begutachtet hat. Für Ihre fachliche Unterstützung, die hilfreichen Anregungen und die konstruktive Kritik bei der Erstellung dieser Arbeit möchte ich mich herzlich bedanken.

Ganz besonders möchte ich Herrn Prof. Dr. -Ing. Knoll für seine Bereitschaft, meine Abschlussarbeit zu betreuen und zu unterstützen, danken - besonders für die persönliche Unterstützung, als auch für die mir gebotene Gelegenheit, diese Arbeit im Rahmen eines HiWi-Vertrags durchzuführen. Ohne eine finanzielle Unterstützung wäre die Bearbeitung meiner Bachelorarbeit in Deutschland nicht möglich gewesen.

Frau Dipl.-Ing. Gentzsch danke ich für die Bereitstellung der Räumlichkeiten und Materialien bzw. Software sowie dafür, dass sie immer gute Laune hatte und jederzeit zu mir nett war.

Mein herzlicher Dank gilt auch Herrn Prof. Dr.-Ing. Merklinger und Herrn Prof. Dr.-Ing. Staiger, die stets ein offenes Ohr für mich hatten und mir immer mit Rat und guter Laune unter die Arme gegriffen haben. Danke für Ihre unermüdliche Unterstützung und für die zahlreichen netten und bereichernden Gespräche.

Bedanken möchte ich mich an dieser Stelle auch bei allen Mitarbeitern der Hochschule Merseburg, die mir gegenüber immer nett und offen waren; insbesondere bei den Mitarbeitern des Auslandsamts, die meinen Aufenthalt in Merseburg erleichtert haben.

Ebenso danke ich meinem Kommilitonen Paul Jürgens, Mateo Topalovic, Frau Astrid Schmeken und meiner Freundin Svenja Niedermaier für das Korrekturlesen. Svenja, dir danke ich auch, dass du immer für mich da warst und mich in dieser Zeit ermutigt hast.

Darüber hinaus möchte ich mich bei meinen Eltern bedanken, die mir mein Studium ermöglicht, mich in all meinen Entscheidungen unterstützt haben und außerdem stets ein offenes Ohr für meine Sorgen hatten.

## *7. Danksagung*

---

Auch möchte ich mich an dieser Stelle ganz ausdrücklich bei dem Stipendienprogramm der brasilianischen Regierung „Ciência sem Fronteiras“ (deutsch Wissenschaft ohne Grenzen) bedanken. Im Rahmen dieses Programms konnte ich mein in Brasilien begonnenes Studium in Deutschland fortsetzen. Ohne diese Förderung wäre ich während meines Bachelorstudiums wahrscheinlich nicht nach Deutschland gekommen. Deshalb bedanke ich mich vielmals für die Gelegenheit und das Stipendium.

Schließlich danke ich all meinen Freunden während der Studienzeit für drei sehr schöne Jahre in Deutschland.

*Amadeus Cavalcanti Salvador de Alcântara,*

Merseburg, 17.03.2017

## Literaturverzeichnis

- 1 Hatem Alkadhi, Sebastian Leschka, Paul Stolzmann, and Hans Scheffel. *Wie funktioniert CT?* Springer, 2011. (document), 2.1, 2.2
- 2 O Dösel. *Bildgebende Verfahren in der Medizin. Von der Technik zur medizinischen Anwendung*, 2000. (document), 2.5
- 3 David Atkinson. *DICOM (for MRI images)*. (document), 3.6
- 4 Gray Henry. *Anatomy of the human body. Organon olfactorius—the nose*, 1918. (document), 4.1
- 5 World Health Organization. *World health statistics 2016: monitoring health for the SDGs, sustainable development goals*. World Health Organization, 2016. 1
- 6 Lorenz C Hofbauer Tilman D Rachner, Sundeep Khosla. Osteoporosis: now and the future. 377(9773):1276–1287, April 2011. 1
- 7 Prof. Dr. W. Eger. *Pathologische Anatomie der Osteoporose unter besonderer Berücksichtigung der Mineralstoffwechselfvorgänge im Knochengewebe*, volume 71 of 0070-4067, chapter Verhandlungen der Deutschen Gesellschaft für Innere Medizin, pages 533–568. Springer, 1965. 1
- 8 Dieter Lüttje. *Osteoporose im Alter*. Thieme, January 2015. 1
- 9 C Cooper AW Lane BL Riggs LJ Melton III, EA Chrischilles. Perspective: how many women have osteoporosis? *Journal of Bone and Mineral Research*, 7(9):1005–1010, September 1992. 1
- 10 F. Jakob, L. Seefried, and M. Schwab. Alter und Osteoporose. *Internist*, 55(7):755, 2014. 1
- 11 R Huiskes and EYS Chao. A survey of finite element analysis in orthopedic biomechanics: the first decade. *Journal of biomechanics*, 16(6):385–409, 1983. 1, 2.6
- 12 PJ Prendergast. Finite element models in tissue mechanics and orthopaedic implant design. *Clinical Biomechanics*, 12(6):343–366, 1997. 1, 2.6
- 13 Fulvia Taddei, Enrico Schileo, Benedikt Helgason, Luca Cristofolini, and Marco Viceconti. The material mapping strategy influences the accuracy of ct-based finite element models of bones: an evaluation against experimental measurements. *Medical engineering & physics*, 29(9):973–979, 2007. 1.1, 3.5, 6
- 14 Thorsten M Buzug. *Einführung in die Computertomographie: mathematisch-physikalische Grundlagen der Bildrekonstruktion*. Springer-Verlag, 2011. 2.1

- 15 Peter Zioupos, Richard B Cook, and John R Hutchinson. Some basic relationships between density values in cancellous and cortical bone. *Journal of biomechanics*, 41(9):1961–1968, 2008. 2.5
- 16 Enrico Schileo, Enrico Dall’Ara, Fulvia Taddei, Andrea Malandrino, Tom Schotkamp, Massimiliano Baleani, and Marco Viceconti. An accurate estimation of bone density improves the accuracy of subject-specific finite element models. *Journal of biomechanics*, 41(11):2483–2491, 2008. 2.5, 3.2, 3.4
- 17 J Galante, W Rostoker, and RD Ray. Physical properties of trabecular bone. *Calcified tissue research*, 5(1):236–246, 1970. 2.5
- 18 Norman A Johanson, Mary E Charlson, Laure Cutignola, Manuel Neves, Edward F DiCarlo, and Peter G Bullough. Femoral neck bone density: Direct measurement and histomorphometric validation. *The Journal of arthroplasty*, 8(6):641–652, 1993. 2.5
- 19 Nikolas K Knowles, Jacob M Reeves, and Louis M Ferreira. Quantitative computed tomography (qct) derived bone mineral density (bmd) in finite element studies: a review of the literature. *Journal of Experimental Orthopaedics*, 3(1):36, 2016. 2.5
- 20 K Engelke, C Libanati, T Fuerst, Philippe Zysset, and HK Genant. Advanced ct based in vivo methods for the assessment of bone density, structure, and strength. *Current osteoporosis reports*, 11(3):246–255, 2013. 2.5
- 21 CM Les, JH Keyak, Susan M Stover, KT Taylor, and AJ Kaneps. Estimation of material properties in the equine metacarpus with use of quantitative computed tomography. *Journal of Orthopaedic Research*, 12(6):822–833, 1994. 2.5
- 22 JH Keyak, IY Lee, and HB Skinner. Correlations between orthogonal mechanical properties and density of trabecular bone: use of different densitometric measures. *Journal of Biomedical Materials Research Part A*, 28(11):1329–1336, 1994. 2.5, 3.2
- 23 Judith E. Adams. Quantitative computed tomography. *European Journal of Radiology*, Volume 71(3):415–424, September 2009. 2.5
- 24 M. Maciel and J. Correia. Quantitative computed tomography versus densitometry in diagnostic of osteoporosis. In *Proc. IEEE 3rd Portuguese Meeting in Bioengineering (ENBENG)*, pages 1–6, February 2013. 2.5
- 25 J. Keenan Brown Alan D. Brett. Quantitative computed tomography and opportunistic bone density screening by dual use of computed tomography scans. *Journal of Orthopaedic Translation*, 3(4):178–184, October 2015. 2.5, 3.2

- 26** Willi A Kalender. A phantom for standarization and quality control in spinal bone mineral measurements by qct and dxa: Design considerations and specifications. *Medical physics*, 19(3):583–586, 1992. [2.5](#), [3.2](#)
- 27** Holzmann Meyer Schumpich Altenbach, Holm. *Technische Mechanik Festigkeitslehre*. Springer, 2014. [2.6](#)
- 28** Peter Steinke. *Finite-Elemente-Methode: Rechnergestützte Einführung*. Springer-Verlag, 2015. [2.6](#)
- 29** Enrico Schileo, Fulvia Taddei, Andrea Malandrino, Luca Cristofolini, and Marco Viceconti. Subject-specific finite element models can accurately predict strain levels in long bones. *Journal of biomechanics*, 40(13):2982–2989, 2007. [3.2](#)
- 30** FP Preparata and MI Shamos. *Computational geometry: an introduction*. Springer, 1985. [3.3.1](#)
- 31** Benedikt Helgason, Egon Perilli, Enrico Schileo, Fulvia Taddei, Sigurður Brynjólfsson, and Marco Viceconti. Mathematical relationships between bone density and mechanical properties: a literature review. *Clinical biomechanics*, 23(2):135–146, 2008. [3.4](#), [6](#)
- 32** Xinshan Li, Marco Viceconti, Marta C Cohen, Gwendolen C Reilly, Matt J Carré, and Amaka C Offiah. Developing ct based computational models of pediatric femurs. *Journal of biomechanics*, 48(10):2034–2040, 2015. [3.4](#)
- 33** Elise F Morgan, Harun H Bayraktar, and Tony M Keaveny. Trabecular bone modulus–density relationships depend on anatomic site. *Journal of biomechanics*, 36(7):897–904, 2003. [3.4](#)
- 34** Pouneh K Fazeli, Mark C Horowitz, Ormond A MacDougald, Erica L Scheller, Matthew S Rodeheffer, Clifford J Rosen, and Anne Klibanski. Marrow fat and bone—new perspectives. *The Journal of Clinical Endocrinology & Metabolism*, 98(3):935–945, 2013. [5.2](#)
- 35** Natascia Di Iorgi, Michael Rosol, Steven D Mittelman, and Vicente Gilsanz. Reciprocal relation between marrow adiposity and the amount of bone in the axial and appendicular skeleton of young adults. *The Journal of Clinical Endocrinology & Metabolism*, 93(6):2281–2286, 2008. [5.2](#)
- 36** Cinzia Zannoni, Raffaella Mantovani, and Marco Viceconti. Material properties assignment to finite element models of bone structures: a new method. *Medical Engineering & Physics*, 20(10):735–740, 1999. [6](#)

- 37** Fulvia Taddei, Alberto Pancanti, and Marco Viceconti. An improved method for the automatic mapping of computed tomography numbers onto finite element models. *Medical engineering & physics*, 26(1):61–69, 2004. **6**

## **Eidesstattliche Erklärung**

Ich, Amadeus Cavalcanti Salvador de Alcântara, Matrikel-Nr. 21053, versichere hiermit, dass ich meine Bachelorarbeit mit dem Thema

*Implementierung verschiedener Algorithmen zur automatisierten Berechnung und Zuweisung von Materialgesetzen von CT-Daten auf FE-Netze*

selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, wobei ich alle wörtlichen und sinngemäßen Zitate als solche gekennzeichnet habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Mir ist bekannt, dass ich meine Bachelorarbeit zusammen mit dieser Erklärung fristgemäß nach Vergabe des Themas in dreifacher Ausfertigung und gebunden im Prüfungsamt der Hochschule Merseburg abzugeben oder spätestens mit dem Poststempel des Tages, an dem die Frist abläuft, zu senden habe.

Merseburg, den 17. März 2017

---

AMADEUS CAVALCANTI SALVADOR DE ALCÂNTARA

## A. Anhang

### A.1. Anleitung zur Erstellung und Speicherung eines FE-Netzes

Die in dieser Arbeit verwendeten FE-Netze wurden in ANSYS-Workbench erstellt und mit dem Tool „Mechanical APDL“ in ANSYS-Classice exportiert (siehe Abb. A.1).

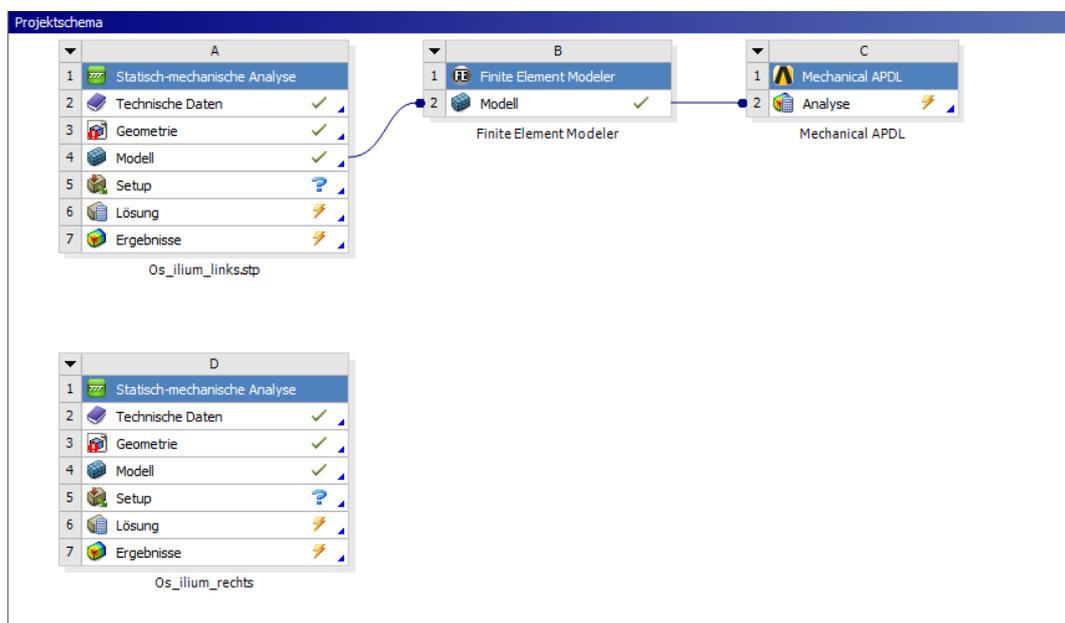


Abbildung A.1.: ANSYS-Workbench Modell

In ANSYS-Classice wurden die Netz-Informationen (Knoten + Elemente) sowohl als cdb-Datei (für Bonemat), als auch als E- und NLIST (für *MatMap*) gespeichert (siehe Abb. A.2).

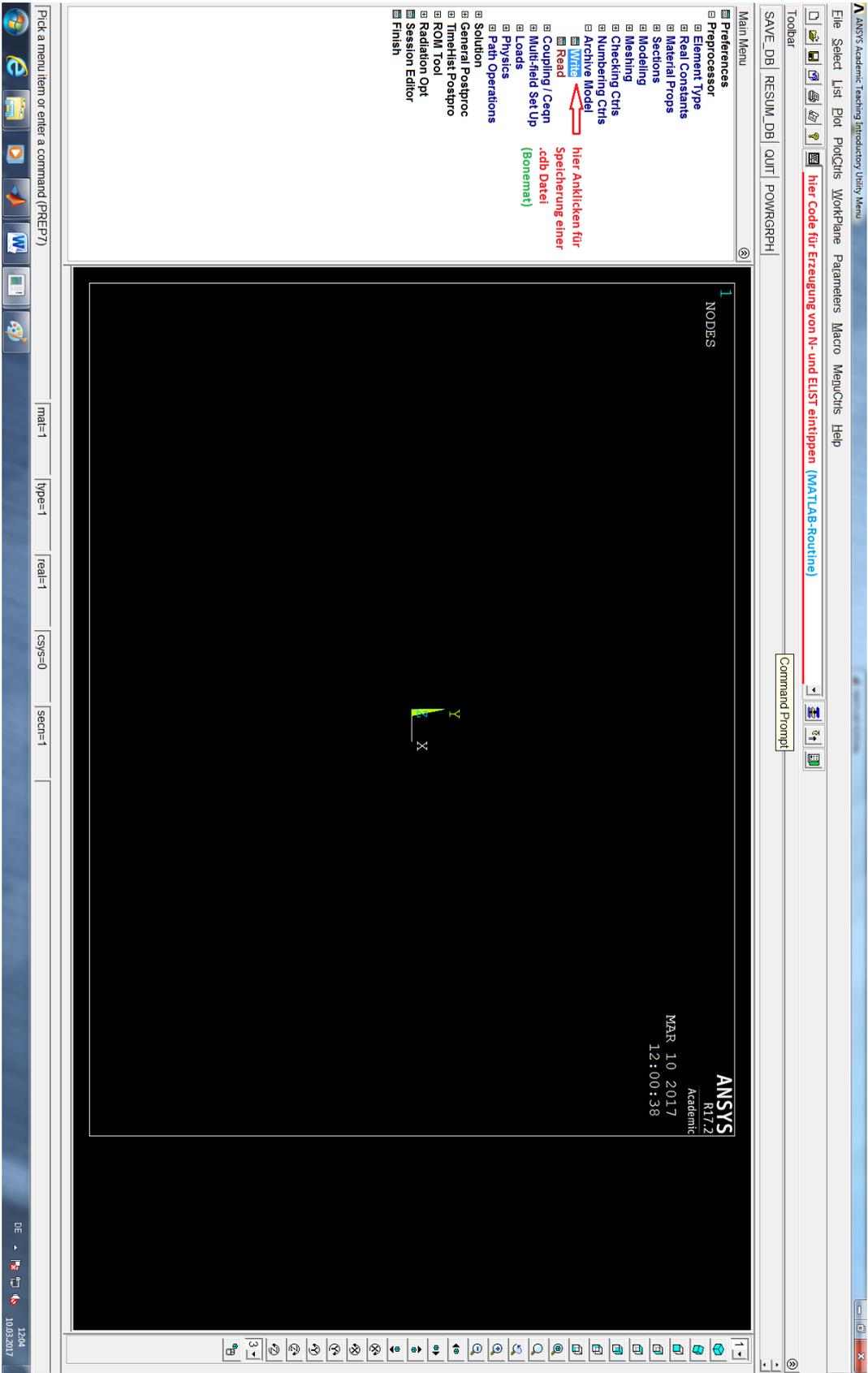


Abbildung A.2.: Speicherung einer cdb-Datei bzw. E- und NLIST

Zur Speicherung der Listen wurden folgende Befehle verwendet.

Code A.1: Exportieren der N- und ELIST in ANSYS-Mechanical-APDL (Classic)  
zum Einlesen des FE-Netzes in *MatMap*

```
1 /prep7           !% Auswahl des Praeprocessors
2 /format,6,G !% Genauigkeit von 6 Nachkommastellen (ausgewaehlt)
3 nwrite,path+filename,lis           !% NLIST exportieren als .lis
4 ewrite,path+fliname,lis           !% ELIST exportieren als .lis
5
6 !% Beispiel
7 !% nwrite,C:\Users\Username\Desktop\OrdnerX\NLIST_Y,lis
```