

## Numerical study of bone fracture healing using a continuum damage mechanics model

**Auteur :** Hu, Louis

**Promoteur(s) :** Boman, Romain

**Faculté :** Faculté des Sciences appliquées

**Diplôme :** Master en ingénieur civil biomédical, à finalité spécialisée

**Année académique :** 2020-2021

**URI/URL :** <http://hdl.handle.net/2268.2/13164>

---

### Avertissement à l'attention des usagers :

*Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.*

*Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.*

---

UNIVERSITY OF LIÈGE

SCHOOL OF ENGINEERING AND COMPUTER SCIENCE

---

**Numerical study of bone fracture healing using  
a continuum damage mechanics model**

---

*Author*  
Louis HU

*Supervisor*  
R. BOMAN

MASTER'S THESIS CARRIED OUT TO OBTAIN THE DEGREE OF  
MASTER OF SCIENCE IN BIOMEDICAL ENGINEERING

ACADEMIC YEAR 2020-2021



## ACKNOWLEDGMENTS

I would like to formally thank my promoter Romain Boman for his support in the day-to-day supervision of this work. His availability and readiness to answer my questions regarding the inner workings of *Metafor* was deeply appreciated. His advice on keeping complex projects didactic and reusable is one that will follow me for the future.

I address my thanks to professor Jean-Philippe Ponthot for his interest in sharing his network of researchers to come up with the subject of this work. Additionally, I thank him for being an exceptional educator and teacher, instilling excellence and rigor in all of his courses, even when their workloads were sometimes overwhelming. Taking up the challenge ultimately pushed us to higher grounds.

I also thank Cédric Laurent for his time when I had questions and doubts about the methodology to adopt and his trust in my ability to handle the complex models involved in the biomedical fields.

To Marc Balligand, while we did not have much interactions in these covid riddled times, I wish him a great read and thank him for his involvement and partnership with professor Ponthot, ultimately giving me the opportunity to complete this thesis.

I want to mention my cat, Chaussette, whose picture has served as a temporary placeholder for many of the figures produced in this work, graphically providing mental support during the writing of this thesis. She can be found at the bottom of this page

Finally, a word for my partner in life Marine Leclercq for her availability, care and support in helping me keep this manuscript readable by a general public. The sentences would have been much longer and convoluted, and commas much more frequent after but without her.



# ABSTRACT

Managing the outcome of bone fracture healing is one of the most challenging tasks that a veterinary orthopedic surgeon has to accomplish. On top of the difficulties involved with unusual patient morphology, tight budgets and often short times to plan out surgeries in traumatic cases, surgeons have to deal with the delicate trade-off between assuring fracture stability and leaving enough interfragmentary motion to promote optimal bone healing on a case by case basis. While handbooks provide general guidelines for surgeons to follow in some situations, decisions are still often based on their experience and expertise gained on previous cases.

Subject-specific models display the potential to positively alter the outcome of surgical operations by providing surgeons with numerical twins to test treatments on. They have previously been shown to be able to reproduce mechanical responses of both intact and orthopedically repaired bones. A continuation of these works is conducted here.

In this thesis, an exploratory study is conducted to assess the capability of a Continuum Damage Mechanics-based remodeling model to represent the healing characteristics found in bone fracture healing. Along the way, the full numerical pipeline from the initial CT image to the Finite Element simulation is didactically presented and improvements made to it are detailed.

Results include improvements to the image-to-mesh pipeline to increase its modularity and robustness. Meshes generated using the improved pipeline show greatly enhanced quality characteristics.

Results also show that the extension of the Continuum Damage Mechanics-based remodeling model is able to qualitatively describe the multi-staged nature of bone fracture healing from a mechanical point of view.



# Contents

<b>I</b>	<b>Towards subject-specific finite element models</b>	<b>1</b>
I.1	Veterinary context of bone fracture healing . . . . .	1
I.2	A need for subject-specific FE models . . . . .	4
<b>II</b>	<b>From bone sample to numerical simulation</b>	<b>7</b>
II.1	Image processing pipeline . . . . .	7
II.1.1	CT Volume labeling . . . . .	7
II.1.2	Labeled CT Volume image to preliminary surface mesh using <i>geniso</i> .	10
II.1.3	Preliminary surface mesh cleaning . . . . .	10
II.1.4	Creation of implants, drilling and cutting of bone surface mesh . . .	14
II.1.5	Volume meshing . . . . .	19
II.1.6	Material property assignation to the volume mesh . . . . .	21
II.2	Numerical Simulation in <i>Metafor</i> . . . . .	25
II.2.1	Importing VTK meshes into <i>Metafor</i> . . . . .	25
II.2.2	Modeling assumptions . . . . .	27
II.2.3	Simulation results . . . . .	28
II.2.4	Improving the management of contact . . . . .	31
II.2.5	Sensitivity study of parameters in contact management . . . . .	33
II.2.6	Simulation results with updated contact management . . . . .	36
II.3	Chapter summary . . . . .	36
<b>III</b>	<b>Continuum Damage Mechanics as a tool to study bone fracture healing</b>	<b>41</b>
III.1	Stanford Model for isotropic bone remodeling . . . . .	44
III.2	CDM for isotropic bone remodeling . . . . .	47
III.3	Sensitivity study of remodeling parameters . . . . .	49
III.3.1	Homeostatic stimulus value . . . . .	51
III.3.2	Remodeling velocities . . . . .	51
III.3.3	Surface area available for remodeling . . . . .	52
III.3.4	Half-width of the lazy zone . . . . .	53
III.3.5	Number of daily load cycles . . . . .	54
III.3.6	Summary of parameter's effects on remodeling behavior . . . . .	55
III.4	Application to bone fracture healing . . . . .	58
III.4.1	Preliminary modeling assumptions . . . . .	58
III.4.2	Determination of the homeostatic tissue stimulus value . . . . .	59
III.4.3	Determination of the remodeling velocity . . . . .	60
III.4.4	Results using determined parameters . . . . .	61
III.4.5	Influence of implant choice and positioning on the healing behavior .	63
III.5	Chapter summary . . . . .	67



# List of Figures

I.1	Fracture healing process, with major metabolic phases (blue bars), biological stages (brown bars), primary cell types found at each stage and time span of each phase. The time-scales are representative of a femur fracture repaired with an intramedullary rod in a mouse. Figure from [5]. . . . .	3
II.1	Image treatment pipeline from a raw CT Volume to a volume mesh suitable for FE simulations. . . . .	8
II.2	Visualization of a CT image slices in <i>ParaView</i> . . . . .	9
II.3	Visualization of a labeled CT image slice in <i>ParaView</i> . . . . .	9
II.4	Clip of the result of a surface reconstruction using <i>geniso</i> . . . . .	10
II.5	(a) Close-up of the result of a surface reconstruction using <i>geniso</i> , with polygon edges displayed. (b) Result of surface decimation done in [9][10] using <i>geniso</i> . . . . .	11
II.6	Process of mesh decimation using <i>pyacvd</i> . . . . .	12
II.7	Mesh fixing using <i>pymeshfix</i> . . . . .	13
II.8	Result of surface remeshing using <i>pyacvd</i> using various mesh density factors. . . . .	13
II.9	Histogram of scaled jacobian quality function for surface meshes decimated using <i>geniso</i> or <i>pyacvd</i> . . . . .	15
II.10	Using the profiles of the bone to determine the orientation of the orthopedic plate. . . . .	16
II.11	Screws generated for a plate with given orientation. . . . .	17
II.12	Surface meshes after cleaning. . . . .	17
II.13	Drilling bone with mesh of screws. . . . .	18
II.14	Mesh cleaning of drilled bone . . . . .	18
II.15	Cutting bone with mesh of a flat box. . . . .	19
II.16	Mesh cleaning of cut bone . . . . .	20
II.17	Clip in the volume mesh, the volume is filled. . . . .	20
II.18	Histogram of scaled jacobian quality function for volume meshes generated using <i>TetGen</i> or <i>gmsk</i> . . . . .	21
II.19	Clip in the volume mesh, coloring based on local density in [g/cc]. . . . .	23
II.20	Clip in the volume mesh, with defect in material properties, coloring based on Young's modulus in [MPa]. . . . .	23
II.21	Resulting distribution of material properties in the mesh after import with different number of bins. . . . .	26
II.22	Equivalent Von Mises stress field. $m = 0.7$ . End-to-end displacement 0.5 [mm], springs stiffness $K = 10^4$ [N/mm], other parameters as described in modeling assumptions. . . . .	28
II.23	Applied force at epiphysis for varying mesh density factors $m$ . . . . .	29

II.24 Maximum equivalent Von Mises stress reached in each object, varying the mesh density factor $m$ . . . . .	30
II.25 Equivalent Von Mises stress field of plate, mesh density factor $m = 0.1$ . . . .	31
II.26 Equivalent Von Mises stress field of screw 3, mesh density factor $m = 0.7$ . . .	31
II.27 Schematic representation of the spring-based contact. Plain lines indicate the surface of an object. Dotted lines indicate linking springs. The linking springs are only shown for the node in red for clarity. . . . .	32
II.28 Schematic representation of the newly implemented improved spring-based contact. Plain lines indicate the surface of an object. Dotted lines indicate linking springs. The springs are colored based on the node of the first object they are attached to. . . . .	32
II.29 Geometry used for contact sensitivity study. . . . .	33
II.30 Displacement ratio between the top and bottom surface of the screw. Element edge length $l_e$ in $\{0.25, 0.50, 1.0, 1.5, 2.0\}$ [mm], spring stiffness $K$ in $\{10^2, 10^4, 10^6, 10^8\}$ [N/mm], number of springs per node $n_S$ in $\{1, 3, 5\}$ [-]. .	34
II.31 Applied force on the top surface of the screw. Element edge length $l_e$ in $\{0.25, 0.50, 1.0, 1.5, 2.0\}$ [mm], spring stiffness $K$ in $\{10^2, 10^4, 10^6, 10^8\}$ [N/mm], number of springs per node $n_S$ in $\{1, 3, 5\}$ [-]. . . . .	37
II.32 Maximum equivalent Von Mises stress in the whole geometry. Element edge length $l_e$ in $\{0.25, 0.50, 1.0, 1.5, 2.0\}$ [mm], spring stiffness $K$ in $\{10^2, 10^4, 10^6, 10^8\}$ [N/mm], number of springs per node $n_S$ in $\{1, 3, 5\}$ [-]. . . . .	38
II.33 Results using the reduced geometry and the over-constraining contact management method. Element edge length $l_e$ in $\{0.50, 1.0, 2.0\}$ [mm], spring stiffness $K$ in $\{10^2, 10^4, 10^6, 10^7\}$ [N/mm]. . . . .	39
II.34 Maximum equivalent Von Mises stress reached in each object and total applied force on the bone, varying the mesh density factor $m$ . Contact management is done using the newly implemented springs generation method. Previous results using the over-constraining method are shown with transparency. . . . .	40
II.35 Equivalent Von Mises stress field of screw 3, mesh density factor $m = 0.7$ . Contact management is done using the newly implemented method. . . . .	40
III.1 Aspect of the remodeling rate in the Stanford model. . . . .	46
III.2 Aspect of the remodeling rate in CDM based remodeling model. . . . .	49
III.3 Results of remodeling simulation using base parameters in Table III.2 . . . .	51
III.4 Results of remodeling simulations. Parameters as in Table III.2 except for the homeostatic stimulus $\psi_t^*$ taking values in $\{1, 5, 15, 30, 50\}$ [MPa]. . . . .	52
III.5 Results of remodeling simulations. Parameters as in Table III.2 except for the remodeling velocities $c_r$ and $c_f$ taking values in $\{0.01, 0.02, 0.05, 0.10, 0.20\}$ [ $\mu\text{m}/(\text{day MPa})$ ]. . . . .	53
III.6 Results of remodeling simulations. Parameters as in Table III.2 except for the available surface area for remodeling $k$ taking values in $\{20, 40, 60, 80, 100\}\%$ . 54	54
III.7 Results of remodeling simulations. Parameters as in Table III.2 except for the homeostatic stimulus $\psi_t^* = 100$ [MPa] and the half-width of the lazy zone $\Omega$ taking values in $\{1, 5, 10, 20, 30\}\%$ . . . . .	55
III.8 Results of remodeling simulations. Parameters as in Table III.2 except for the number of load cycles $N$ taking values in $\{1, 10, 100, 1000, 10000\}\%$ . . . . .	56

III.9 Final density and damage for varying values of the homeostatic stimulus value $\psi_t^*$ . The dashed line represents the average properties of healthy bone	60
III.10 Density and damage evolution for varying values of the remodeling velocities $c_f$ and $c_r$ . Units of the legend in [ $\mu\text{m}/(\text{day MPa})$ ]	61
III.11 Density, damage and stimulus evolution for varying values of the mesh density factor. Parameters as defined in the text, and $\psi_t^* = 5$ [MPa], $c_f = c_r = 0.4$ [ $\mu\text{m}/(\text{day MPa})$ ], as determined in subsections III.4.2 and III.4.3. For the density and damage evolution the dashed line represents healthy bone properties. For the stimulus, the dashed line represents the edge of the remodeling lazy zone.	62
III.12 Evolution of EVMS characteristics in the plate, the bone and the clot and epiphysis displacement evolution during modeled bone fracture healing. Parameters as defined in the modeling assumptions and $\psi_t^* = 5$ [MPa], $c_f = c_r = 0.4$ [ $\mu\text{m}/(\text{day MPa})$ ], mesh density factor $m = 0.9$ .	68
III.13 Evolution of mean damage and density in the clot using different screw configurations. Evolution of EVMS in the plate, the bone and at the fracture site. The plate offset is held constant at 2 [mm]. The legends display the number of screws and the configuration. $m = 0.9$ .	69
III.14 Force displacement curves for different screw configurations. The plate offset is held constant at 2 [mm]. The legend displays the number of screws and the configuration. $m = 0.9$ .	70
III.15 Force displacement curves and maximum displacement for different plate offsets. The screws use configuration 1. $m = 0.9$ .	70
III.16 Evolution of mean damage and density in the clot using different plate offsets. Evolution of EVMS seen in the plate, the bone and at the fracture site. Configuration 1 is used for the screw, i.e. screws in holes number 1, 2, 7 and 8.	71

# List of Tables

II.1	Statistical quantities for the quality of elements in the surface meshes after decimation using <i>geniso</i> or <i>pyacvd</i> . . . . .	14
II.2	Statistical quantities for the quality of elements in the volume meshes generated by <i>TetGen</i> or <i>gmsh</i> . . . . .	22
II.3	Summary of mesh characteristics at the end of image-to-mesh pipeline. Volumes in [mm <sup>3</sup> ]. . . . .	24
II.4	Summary of material properties contained in the intact bone meshes at the end of image-to-mesh pipeline. Densities in [g/cc], Young's modulus in [MPa].	24
II.5	Computation times for different mesh density factors . . . . .	29
III.1	Summary of density evolution cases. . . . .	46
III.2	Base remodeling parameters used in this section . . . . .	50
III.3	Computation times in fracture healing simulations for different mesh density factors. . . . .	63
III.4	Screw configurations used in impact study on healing behavior. Pink cells indicate positions occupied by the fracture site, blue cells indicate the presence of a screw. . . . .	64
III.5	End results of simulations with different screw configurations. . . . .	65

# I Towards subject-specific finite element models

## I.1 Veterinary context of bone fracture healing

Bone fracture is one of the most frequent injuries involving the musculoskeletal system. While common in occurrence, the orthopedic processes involved in the treatment and repair of bone fractures show a very large diversity due to the wide range of possible fracture sites and the various bone geometries encountered in nature. Most healing strategies involve the stabilization of the fracture site in order to create an optimal healing environment left for the biological machinery to act upon.

Bone healing patterns are broadly classified in two categories: primary and secondary [1]. Primary bone healing occurs when the bone fragments are still in close proximity to each other and the fragments are fixed together under compression from external mechanical constraints. In this pattern, bone directly heals through the actions of osteoclasts and osteoblasts that gradually resorb the fracture and yield a consolidated material. In the more commonly observed secondary bone healing, a callus forms at the fracture site due to the slight motion that is possible between bone fragments in the site. This interfragmentary motion leads to the appearance of an anabolic callus that gradually mineralizes and resorbs concurrently to the formation of bone inside the fracture site. Formed bone then undergoes long term transformations to bring the material back to its original state before the traumatic incident.

Secondary bone fracture healing is a complex process usually split in four phases. The first phase involves an immediate inflammatory response initiated by chemical signaling at the fracture hematoma site lasting 3-7 days [2]. Then, it is followed by two bone formation phases, the first involving the formation of a cartilaginous mass at the fracture site for 2-4 weeks, and the second replacing the cartilaginous material into a precursory bone structure in 2-4 months. The fourth phase, which completes the fracture repair process, is a remodeling phase, during which the precursory bone is adapted to fit to mechanical stimulus generated at the fracture site. This last phase can last anywhere between months and years depending on the site of fracture and the individual's healing response. A remarkable feature of bone healing is that the final repaired bone looks almost identical to what it looked like before the fracture.

### Phases of fracture healing

- *Inflammation phase:* Right after the fracture event, cells in the peripheral and intramedullary blood system, as well as bone marrow cells, aggregate to form a

hematoma. The hypoxic and acidic environment of this hematoma serves as a scaffold for inflammatory cells to colonize. The inflammatory cells then release multiple cytokines and growth factors during the early phases of the inflammatory response. In combination to cytokines and growth factors, angiogenic factors also mix thanks to the hypoxic conditions created by the initial trauma, promoting the formation of new blood vessels in the fracture site. This inflammatory response has been shown to be necessary to promote subsequent healing processes [3].

- *Bone formation phases, soft callus* : After the temporary inflammatory response, the deposition of structural components helping in the healing response starts. Mesenchymal stem cells from the bone marrow and peripheral blood vessels differentiate into chondrocytes and osteoblasts and participate in the formation of extracellular matrix that helps with the stabilization of the fracture site. This extracellular matrix is initially composed mainly of soft cartilage formed by the chondrocytes, with the osteoblasts being trapped inside this matrix.
- *Bone formation phases, hard callus* : Following the deposition of an extracellular matrix, the chondrocytes then go through an hypertrophic phase and the previously deposited soft cartilage starts to mineralize. This mineralization process is due to the accumulation of calcium granules in the cytoplasm of chondrocytes, which are then transported into the extracellular matrix where they precipitate with phosphate to form mineral deposits. These mineral deposits will later serve as nucleation points for the formation of apatite crystals that compose the bone matrix [4]. Osteoclasts are then involved in the resorption of the mineralized cartilage and osteoblasts gradually replace the cartilage with woven bone. This first phase of woven bone deposition can be achieved relatively fast due to much higher achievable rates of production. While mechanically inferior to its lamellar counterpart, the woven bone can provide additional structure and support the invasion of mature vascular structures inside the fracture site. This process is called endochondral ossification [1].
- *Remodeling phase* : In this final phase of bone fracture healing, the woven bone that was previously deposited is gradually resorbed by osteoclasts and lamellar bone is deposited by osteoblasts. During this phase, the final healed bone structure is reformed to its healthy state with the osteon structures allowing the bone to remodel. The previously formed callus is slowly resorbed by osteoclasts, and the cortical arrangement of long bones is recovered. During this phase, a gradual reapplication of mechanical loading on the bone has an influence on the outcome and duration of the remodeling process. An application of pressure on the crystalline structure of the bone has an stimulating effect on the activity of osteoclasts and osteoblasts involved in remodeling[4]. As will be further exposed in later chapters, this last phase of healing gradually adapts the structure to optimally allocate bone mass in order to sustain the loading that it is subjected to.

Figure I.1 graphically illustrates the previously described fracture healing process.

Orthopedic surgeons dispose of a variety of tools and techniques to assist nature in the stabilization of traumatic fractures. Treatments mainly aim at achieving osteosynthesis of the bone fragments, i.e. the union of the bone fragments after proper alignment has been achieved. To this end, surgeons may choose techniques classified under the broad categories of internal fixation techniques and external fixation techniques. In internal fixation



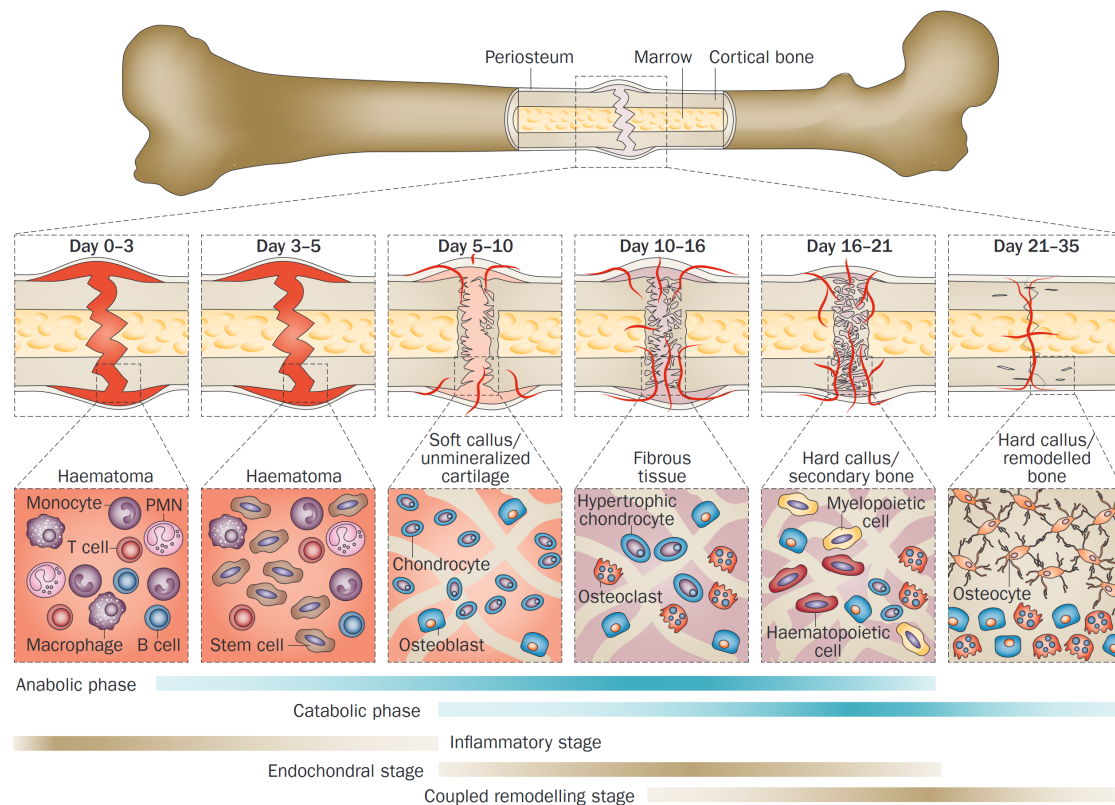


Figure I.1: Fracture healing process, with major metabolic phases (blue bars), biological stages (brown bars), primary cell types found at each stage and time span of each phase. The time-scales are representative of a femur fracture repaired with an intramedullary rod in a mouse. Figure from [5].

techniques, surgical implantation of a variety of orthopedic devices is done to mechanically stabilize bone fragments after proper alignment has been carried out. This alignment determines the type of fracture healing pattern that is subsequently achieved. Ideally, primary healing is sought by orthopedic surgeons, as the resulting fracture site is then less sensitive to complications such as implant failure, proximity tissue damage, or in general postoperative infections, as internal fixation techniques are usually stiffer than external fixation. This increased stiffness in turn prevents damaging large motions across the fracture that could negatively affect the outcome of fracture healing. However, primary healing is not always achieved due to larger than anticipated interfragmentary gaps or if the fragments are not geometrically suited for tight fixation. In this case, internal fixation may still be used to achieve secondary bone fracture healing. In internal fixation, the choice of orthopedic implant is made between screws, plates, nails, intramedullary rods, each with various available designs that are adapted to the bone geometry and fracture location.

External fixation techniques are employed when the fracture site is inadequate for internal fixation techniques. In this case, secondary fracture healing is almost always the outcome of the operation. In external fixation operations, the bone fragments are pierced by pins (long cylindrical rods with one end through the bone and the other end in a fixing jig) and the distance between the bone fragments is maintained using an external fixator. By maintaining the spacing and orientation of the bone fragments, surgeons achieve the needed osteosynthesis. Fixators can be classified depending on the number of planes in

which the pins are slotted, i.e. uniplanar, multiplanar, unilateral, bilateral, or circular fixators. The choice of fixator type is not so much dictated by the geometrical factors at the fracture site as it is by the required rigidity of the fixating jig to properly maintain the broken fragments apart; with more pins and more planes bringing additional stability at the cost of both longer and more challenging surgeries, and also more potential spots for post-operative infections. External fixation operations often require much closer attention to post operative complications[6], not only due to the fact that they do not provide the rigidity of internal fixation techniques but also because the choice of an external fixator is conditioned by a worse situation at the fracture site.

In the case of veterinary orthopedics, in addition to the inherent challenges in stabilizing fractures, other factors impair the surgeons ability to optimally repair fractures.

While most experimental studies on bone growth, remodeling and repair in the scientific literature is done on animals, this infatuation does not translate to the clinical domain of bone fracture healing. Indeed, while plates, screws, and rods are available in a profusion of designs, variants, and materials for essentially any geometry of human bone, veterinary surgeons often have to make use of a limited set of surgical implant designs. The choice not being between two plates of slightly different designs both adapted to the current bone geometry, but between different implant types where one implant is poorly adapted and the other requires much more invasive surgery. This more limited choice of implant design may be explained by the much larger variability in patient morphology encountered in the veterinary field. Just in the canine family, size differences can span whole orders of magnitudes, which would require prohibitive resources to fully cover the whole range of sizes and geometries.

In addition to the more limited implant designs, veterinary surgeons must deal with highly reduced post-operative patient inactivity times, as animal patients tend to have difficulties limiting their mobility after a surgery. This limitation leads to a major trade-off that veterinary surgeons have to balance based on their experience. On one side, the surgeon can repair a fracture to optimize post operative healing characteristics (especially in the remodeling part of healing), thus choosing lower implant rigidity to allow for some beneficial interfragmentary movement during the remodeling phase of the healing, the risk of this decision being the possibility of premature failure of the implant due to overloading on the fracture site due to animal impulsivity. On the other side, the surgeon may choose to put a very rigid implant to strongly stabilize the fracture site to promote the earlier stages of bone healing (callus formation) and not risk the failure of the implant, at the cost of having a less satisfactory remodeling phase of healing, leading to slightly more fragile bone after healing, potentially leading to damaging long-term stress shielding effects[7], even though recent advances in plate design seem to slightly alleviate that last limitation [8].

Another general difficulty is the budget constraints that veterinary surgeons are faced with due to the lack of health insurance programs for animals.

## **I.2 A need for subject-specific FE models**

From the difficulties outlined in the previous section, one can hope to reduce the uncertainties in the determination of the post-operative stiffness of a bone-implant construct

by simulating multiple surgical scenarios using the specific geometry of the bone at hand. Provided that the numerical models are able to accurately reproduce the stiffness of the bones fixed using the aforementioned surgical scenarios. The motivation behind the introduction of subject-specific models is to save the need for performing costly large scale experimental studies testing multiple surgical scenarios on large samples of bones to determine the resulting post-operative stiffness. Moreover, subject-specific numerical studies may be able to overcome the inherent variability between individuals of such large scale studies. Indeed, in lieu of testing the mechanical responses of different treatment on two bone samples presenting slight natural variations, subject-specific models would make it possible to test these treatments numerically on a geometry extracted from the same sample, thus preventing some of the uncertainty that stems from inter-individual differences between samples.

Following this idea and in hopes to improve the outcome of veterinary surgical procedures, both ex-vivo and numerical experiments have been conducted previously [9][10] at the LTAS-MN2L department, University of Liège, in collaboration with the Faculty of Veterinary Medicine, University of Liège, with a dual objective. The first objective was the assessment of the mechanical response of canine bone samples subjected to three point bending (for intact samples) and simple compression (for surgically cut and repaired samples). The second objective was to develop a pipeline of numerical tools that are capable of creating a numerical geometry of a bone sample suitable for FE simulations. Then using this numerical equivalent to obtain a simulated response that is validated against the experimental counterpart by reproducing the experimental conditions numerically. The outcomes of these studies have shown a reasonably good fit between numerical and experimental results.

This work continues in the same framework, adding a few more bricks to the subject-specific modeling domain by improving some aspects of the previously developed models and exploring new ways in which this subject-specific geometry can be used. Some key previous developments and works in patient-specific models creation on which this work is based are :

- A tool for the generation of surface meshes using raw Computerized Tomography (CT) volume images, *geniso*, developed by V.d'Otreppe in a thesis titled "*From medical imaging to finite element simulations: a contribution to mesh generation and locking-free formulations for tetrahedra*". [11]
- Works by C.Laurent [9] in the creation of numerical equivalents of canine bone samples suitable for FE simulations and the mechanical validation of such models.
- Works by C.Laurent [10] in the creation of tools to generate and position numerical equivalents of orthopedic implants and surgical operations in bone repair such as plates and screws for implants and cutting and drilling for surgical operations.

Another major contribution to this work is the developments achieved by M.Mengoni in a thesis named "*On the development of an integrated bone remodeling law for orthodontic tooth movements models using the Finite Element Method*", in which a bone remodeling model based on Continuum Damage Mechanics theory is advanced.

In this work, an exploratory study is conducted to assess the capability of a Continuum Damage Mechanics-based remodeling model to represent the healing characteristics found in bone fracture healing when its initial domain of application is extended. Along the way, the full numerical simulation pipeline from initial CT image to Finite Element analysis is didactically presented and improvements made to it are detailed.

# II From bone sample to numerical simulation

## II.1 Image processing pipeline

This section describes the pipeline of image treatment and the operations done to go from a volume CT (Computerized Tomography) scan of a bone sample to a volume mesh usable in the context of FE simulations. Tools and steps involved in the processing of biomedical images will be presented. Going step-by-step from a CT volume image to a volume mesh suitable for use in FE simulations involves quite a few tools, file formats, image processing steps and jargon that can be overwhelming for anyone trying to work on non-CAD generated biomedical geometries. The goal of this chapter is not only to describe the process used in this work, but also to serve as an introductory reference for any newcomer to the subject of creating numerical equivalents of biomedical geometrical models.

Along the way, improvements and changes made to a previously available processing pipeline developed by C. Laurent in [9][10], itself based on mesh manipulation tools created by V. d'Otreppe [11], will be described and are intended for an audience that is already familiar with the subject. These changes hopefully bring tangible improvements to the robustness of the meshing process and the quality of resulting meshes.

A schematic summary of the treatment pipeline is shown in Figure II.1. Intermediate files are displayed in yellow boxes, operations in blue diamonds, and final files used in simulations in green boxes. The following subsections go over each of these steps in more detail and highlight the changes made to each part of the meshing process.

### II.1.1 CT Volume labeling

A raw CT file (volume) is a stack of 2D images(slices) along a direction normal to the 2D images. Each point in the 2D images is attributed a value depending on the density of the material at that point. Upon stacking a sufficient amount of images, one obtains a 3D image in which the magnitude of the density-dependent value at any point in the the CT Volume can be retrieved. A visualization of a CT image in *ParaView*[12] can be found in Figure II.2a, where a slice of the CT volume is shown. Note that the previously mentioned stacking was done along the direction of the bone axis, using images such as the one in Figure II.2b.

Typically, CT scanners do not give a direct value for the density. Instead, they provide a relative quantitative measurement of radio density, expressed in Hounsfield unit (HU). The HU units can then be transformed into density information provided that scans of calibrated

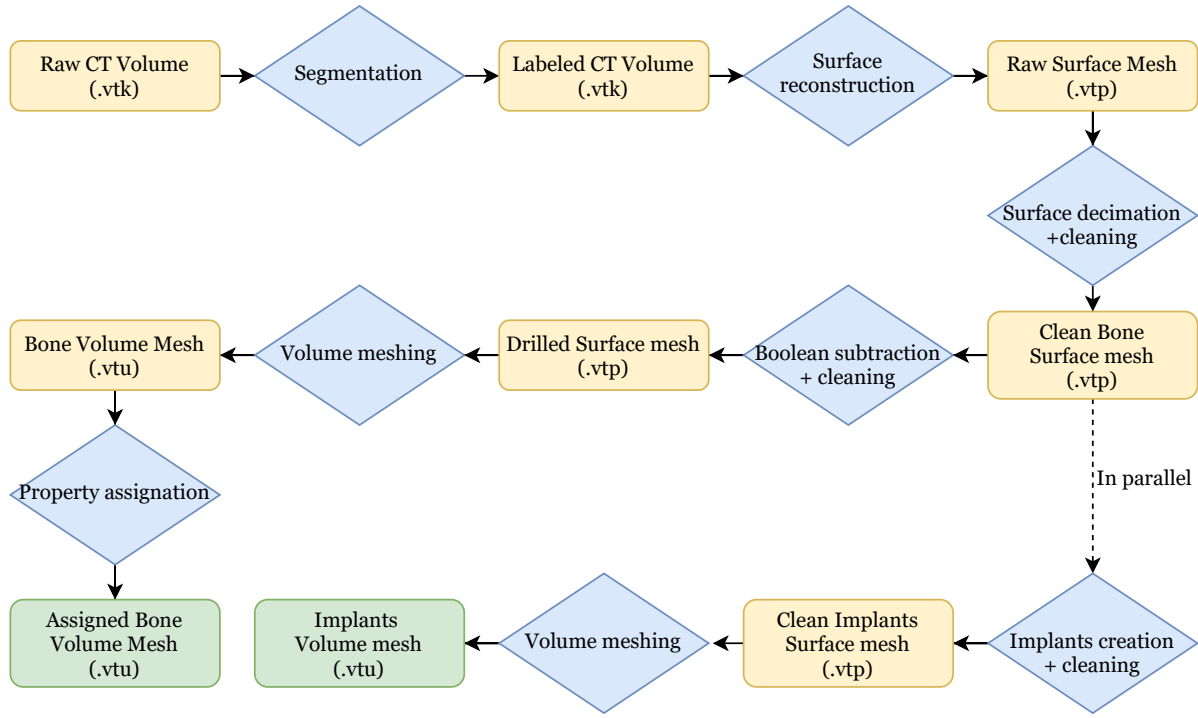


Figure II.1: Image treatment pipeline from a raw CT Volume to a volume mesh suitable for FE simulations.

samples with known densities are made using the same machine. A transformation of the HU unit into workable information such as local tissue density can then be made for each point in the CT image. Additional information about the HU-density relationship can be found in later subsections where the assignation of mechanical properties to the 3D mesh is done.

Before anything else, CT images are labeled to distinguish the areas where there is material from where the CT volume should be empty. Formally, this labeling operation is called 'segmentation' and is done here using a software utility named *3D Slicer* [13]. The segmentation step consists in going over every slice of the CT volume and indicating boundaries around regions of interest, thus assigning a label to every point in the 3D image. In the case of bone samples in the context of this work, the regions of interest are:

- the bone main volume,
- the region occupied by resin blocks used in mechanical testing,
- optionally, the intramedullary canal region inside of the main bone volume.

The segmentation process is a time- and labor-intensive operation in the image treatment pipeline. Most segmentation operations require manual input from a skilled technician and they are not easily automated. Moreover, a well-executed segmentation is critical to the quality of resulting meshes. In this work, no actual segmentation operation has been conducted from scratch but results from previously done segmentation are used in the meshes generated here. A visualization of the resulting labeled CT Volume in *ParaView* is provided in Figure II.3. Again, the same comment is made about the stacking of slices: here the visualization is done using a slice along the bone axis but the segmentation process used the individual slices that were normal to the bone axis to create labeled regions.



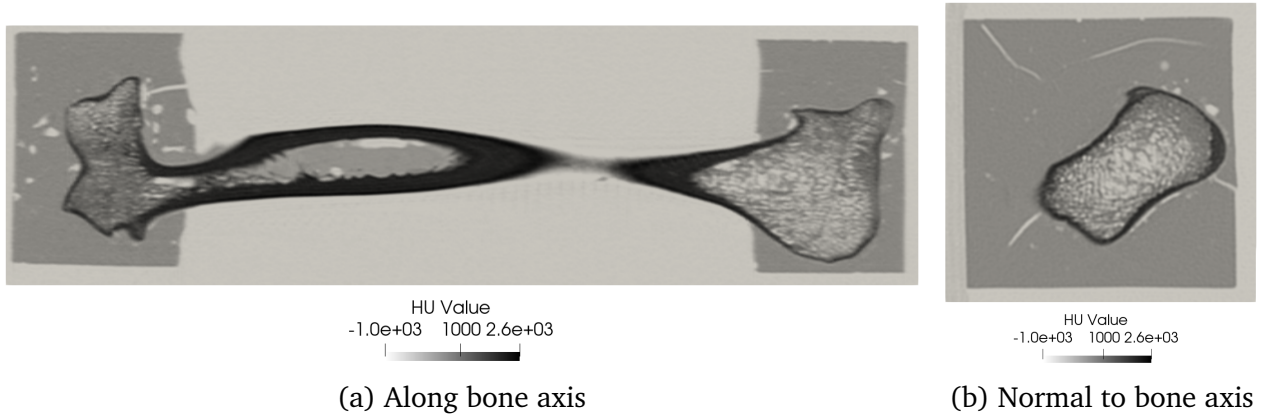


Figure II.2: Visualization of a CT image slices in *ParaView*.

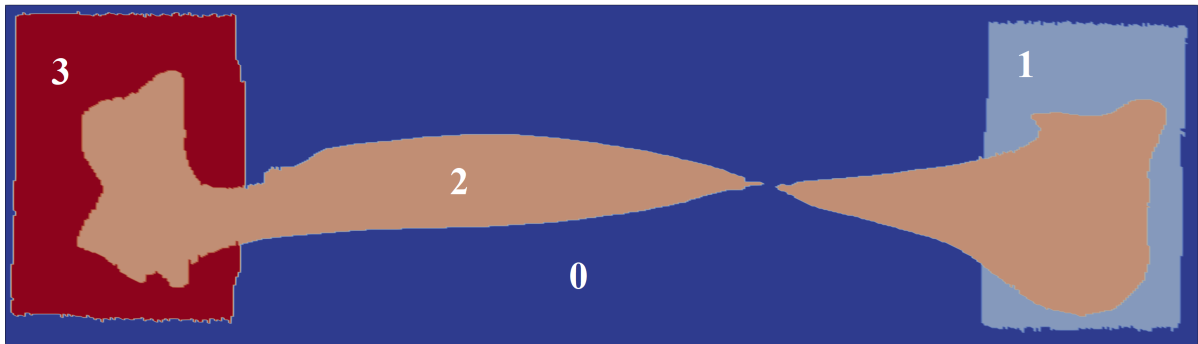


Figure II.3: Visualization of a labeled CT image slice in *ParaView*.

This work makes extensive use of the features provided by the Visualization Toolkit (VTK)[14], be it directly through the use of filters, special information classes and the corresponding file formats; but also indirectly through the use of data visualization software such as *ParaView* that can read and display the file formats from VTK. The Visualization Toolkit is used ubiquitously when dealing with biomedical imaging data, with many scanner(CT or others) file formats providing either a direct conversion kit to VTK file formats or community-sourced conversion kits that reverse engineer proprietary file formats to make use of the powerful data processing capabilities of VTK.

In the case of the CT Volume images used in this work, the original scans were captured in the DICOM file format, then converted into VTK's file formats. For the raw CT Volume image, the extension used is either the legacy format `.vtk`, or the more recent XML-based format `.vti`. The `.vtk` is the legacy file format of VTK and it is capable of holding most types of information that is handled by VTK, they are easier to write by hand or programmatically, but they do not provide newer features made available through the use of XML file formats (e.g. `.vti`) such as random access, parallel I/O, and efficient data compression for smaller file sizes. In general, it is preferred to use the newer XML file formats, but the legacy file format is still supported for compatibility reasons. Moreover, the XML-based file formats (`.vti` here, `.vtp`, `.vtu` later) allow end-users to distinguish the type of data stored in the image at a glance, preventing confusion when manipulating large amounts of files with different internal formats.

The specific class type of image information contained in the `.vtk` file corresponds to the VTK class `'vtkStructuredPoints'`, here called structured points in the text. Structured

points can compactly represent information contained at points with regular geometrical spacing between them. They can have different spacings between points in a single slice and between points from adjacent slices. Knowing the spacing of these points and the number of them in each direction of space, one can retrieve any points using a triplet of indices. The data class also attributes a HU value for each point of space, giving a proxy value of the density at that point.

### II.1.2 Labeled CT Volume image to preliminary surface mesh using *geniso*

The surface reconstruction from labeled CT volume images is done using tools (*geniso*) developed by V. d'Otreppe [15][11]. A sizable advantage of these tools is their capability to generate surface meshes that have low segmentation noise (common when previously labeled boundaries in the 3D volume are fuzzy between slices of a 3D image), resulting in a smooth model boundary. A clip of the resulting surface mesh is shown in II.4. At this point, the resulting surface meshes are very detailed and have a high poly count. Due to this high polygon density, trying to create a volume mesh from this surface mesh would be extremely costly. Moreover, without post-processing, the sheer amount of triangles constituting the surface mesh makes volume meshing an arduous task at this point (see Figure II.5a).

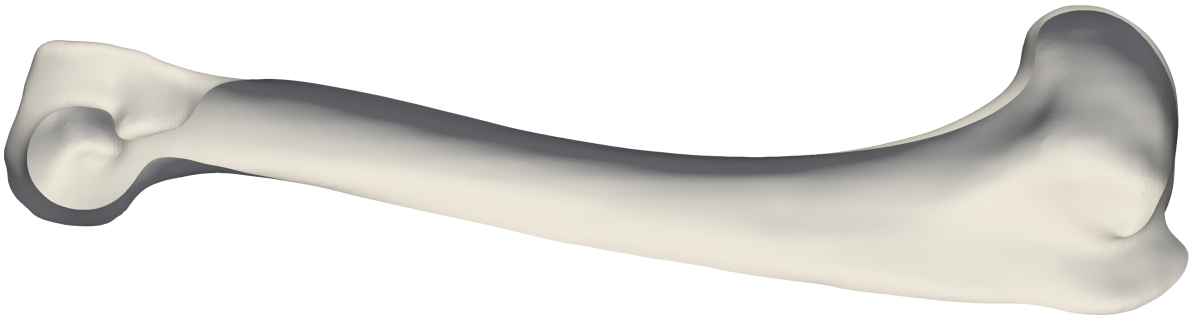


Figure II.4: Clip of the result of a surface reconstruction using *geniso*.

Surface meshes are stored in XML files with the `.vtp` extension. The class type of the image is 'vtkPolyData', here called polydata in the text. Polydatas are used to represent vertices, lines and polygons in 3D space. They are fundamentally different from the previous structured points as there is no regular spacing and points are no longer accessible using a simple triplet of indices. Instead, the format defines vertices(points) and appends connectivity information to join points into lines or polygons(cells). Accessing the points and cells is done through indices or directly from coordinates. The data type can also hold information related to points, or information related to cells. In a hypothetical 2D case, it could for example hold information such as point mass or density and cell information such as Young's modulus to be used in the assignation of properties in a FE code.

### II.1.3 Preliminary surface mesh cleaning

Tools to reduce the number of polygons while keeping the fidelity to the original image are available in the *geniso* toolbox. Reducing the number of polygons while still keeping good geometrical accuracy is formally called 'decimation'. Running this decimation step would gradually reduce the number of triangles in the surface mesh, the operation gradually



removes points on patches of the surface until a certain reduction ratio is achieved. Ideally, node removal would start in regions where the curvature is not high, however, neither *geniso* nor the tools described in the following possess this capability, as evidenced by the random removal seen in Figure II.5b.

At this stage, the previously developed models [9][10] made extensive use of the decimation tools provided by *geniso* to obtain an approximate geometry using a reduced number of polygons before continuing to the cutting and drilling part of the pipeline. A resulting mesh would have looked like the one in Figure II.5b.

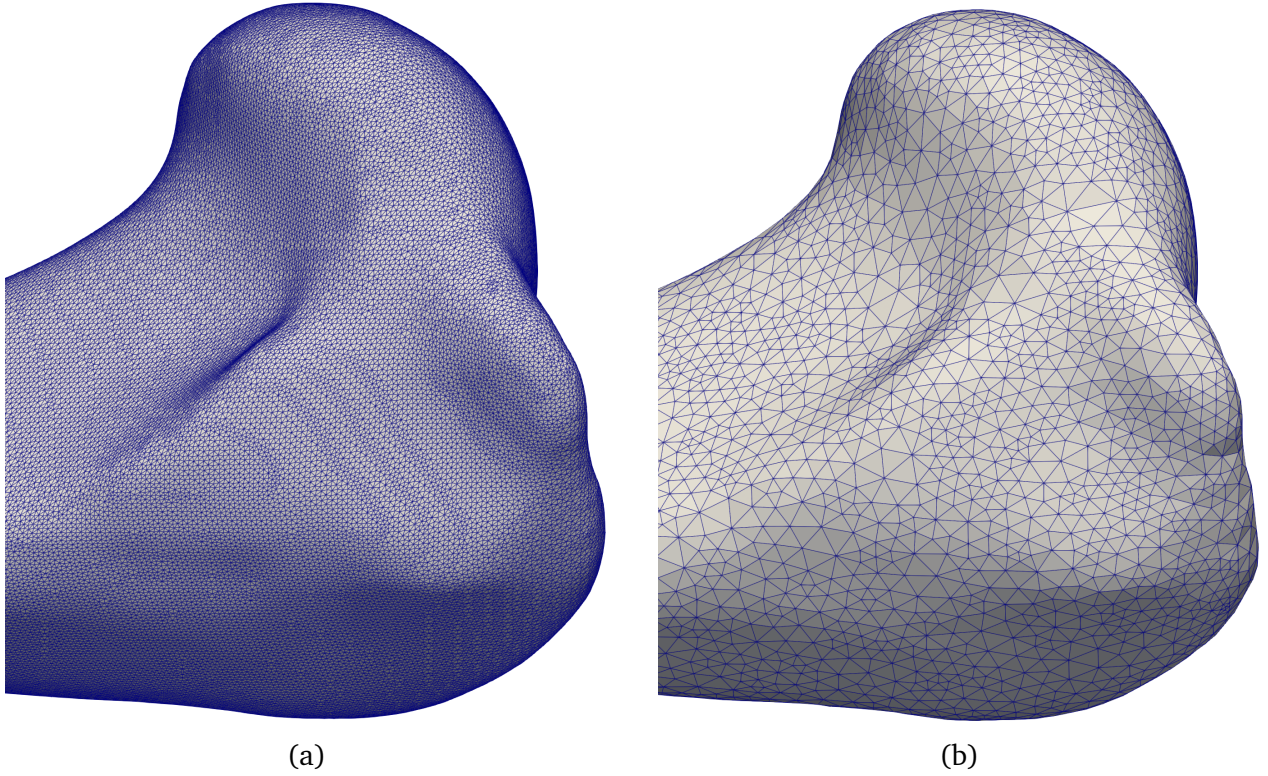


Figure II.5: (a) Close-up of the result of a surface reconstruction using *geniso*, with polygon edges displayed. (b) Result of surface decimation done in [9][10] using *geniso*.

In this work, a suggestion was made to turn to more openly available tools to achieve a similar result. While the previous surface reconstruction required tools from *Metafor* (the *geniso* toolbox is bundled with *Metafor*), the following mesh manipulation operations can be freely done outside of the *Metafor* environment, using open-source software and tools.

As mentioned previously, handling medical data involves many file formats. In the pipeline of manipulation, filtering, cutting and cleaning of meshes, the capabilities of VTK are used quite extensively. While powerful, the syntax and implementation of filters using the base VTK functions can quickly get tedious. Moreover, accessing information fields of VTK files using the Python API results in data objects that are only usable with *vtk*. For example, accessing cell data such as the density of cells in the mesh results in a *'vtkCellData'* object. If we were to limit ourselves to tools provided by VTK, these internal data objects would be sufficient, but as soon as other post-processing (e.g. for re-meshing or filling holes in meshes) tools are included, cross-compatibility of these data objects becomes a significant drawback.

To solve the cross compatibility problems, the Python toolbox *PyVista* [16] is used to

ease the manipulation of VTK files. To ease access to the fields inside the VTK files, *PyVista* creates a wrapper around the different VTK file types and makes the fields of VTK objects accessible as if they were members of a mesh 'object'. The resulting member data objects are standard *NumPy* arrays that are widely supported across processing tools. The toolbox also eases the implementation of filter chains as it provides an 'object-oriented' approach to image management, filters being accessible and usable as if they were the methods of the mesh object. Another advantage of using a toolbox such as *PyVista* is the availability of simpler plotting routines helping in the visualization of these operations. The interested reader should consult the available documentation of the *PyVista* package for the full list of features provided.

Two packages made available through the main *PyVista* package that will be used in the following for, respectively, mesh decimation and cleaning are *pyacvd* [17][18][19] and *pymeshfix* [20].

*pyacvd* aims to provide fast and efficient mesh decimation and resampling for triangular meshes. The method is based on the generation of partitions similar to centroidal Voronoi regions, where each region is similar in size. The results of operations made by *pyacvd* can be seen on the sequence of meshes in Figure II.6. The number of points present in the final surface mesh can be chosen (it is essentially the number of Voronoi clusters that the tool produces), allowing us to create more or less dense meshes from the initially very detailed surface mesh generated by *geniso*. The choice of the surface density of the meshes is chosen here, with the default value being one point per  $\text{mm}^2$ . A mesh density factor  $m$  is then defined to create meshes with varying levels of surface point density. The mesh density factor being such that the total number of points in the surface mesh is the multiplication of the base density, the area of the pre-decimation surface mesh and the mesh density factor  $m$  :  $\text{nb points} = m \times A[\text{mm}^2] \times 1[\text{mm}^{-2}]$ .

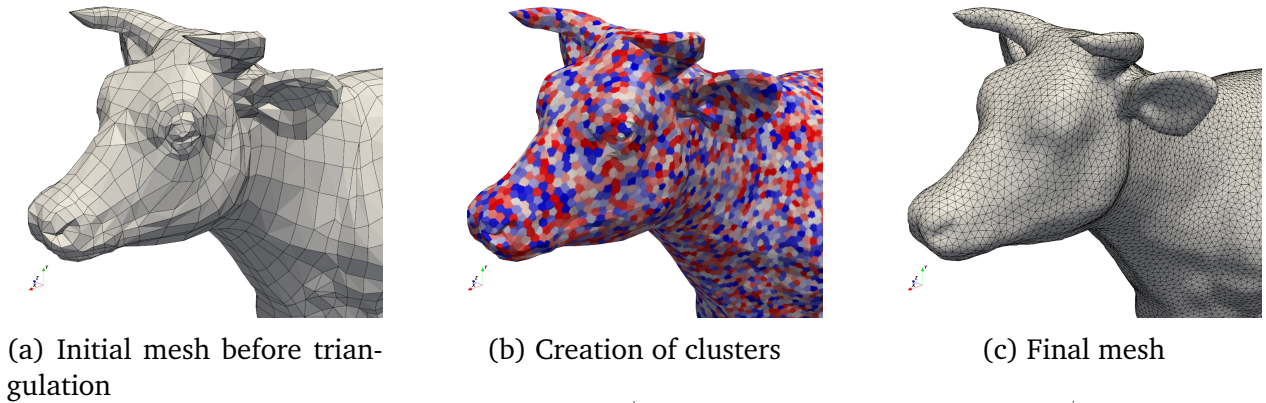


Figure II.6: Process of mesh decimation using *pyacvd*.

*pymeshfix* is the second tool bundled in *PyVista* that will be used extensively in mesh cleaning operations. This tool takes as input a polygon mesh and produces a copy of the input where all the occurrences of a specific set of "defects" are corrected. The resulting meshes are single manifold and watertight triangle meshes without degenerate or intersecting elements. An example of the use of this tool is shown in Figure II.7. Using this tool is required mainly because the cutting and drilling operations presented in the following subsections tend to create extremely distorted elements where the cutting or drilling takes place. In some cases, these operations also produce triangles that are flipped or stacked on top of another existing mesh.



Figure II.7: Mesh fixing using *pymeshfix*.

In the previously developed models [9][10], mesh cleaning operations were done on a mesh by mesh basis. The cleaning consisted in the detection of nodes that were in a region likely to contain distorted elements, and performing a few iterations of node deletion and cell collapsing operations. These operations were tied to certain objects, e.g. entire algorithms that only cleaned the mesh of the plate, but not the drilled bone or the screws, resulting in different treatment pipelines for each geometric object required for the final simulation. In addition, the need to specialize the mesh transformation pipeline to certain geometric objects hinders the potential for reuse of the code as the operations cannot, or only with multiple adaptations, be used on more generic meshes.

For this project, these lengthy specialized operations have been replaced by the use of the two aforementioned tools, resulting in a more streamlined approach to mesh cleaning. Cleaning operations in this work will rely on a combination of the two tools. When necessary, a new surface with homogeneous triangles is defined using *pyacvd*, then it is cleaned using *pymeshfix* to remove any defects on the surface mesh. The result of such a treatment can be seen in Figure II.8, where the number of nodes on the surface of the mesh can be freely chosen through the specification of a mesh density factor. The mesh generated using  $m = 0.7$  has the same number of nodes as the one found in Figure II.5b

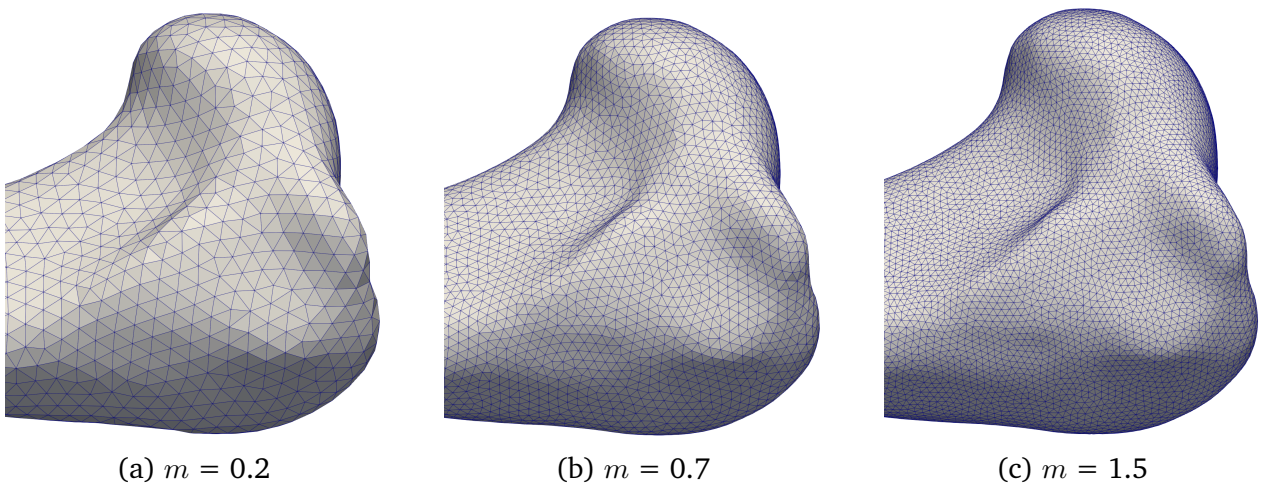


Figure II.8: Result of surface remeshing using *pyacvd* using various mesh density factors.

These cleaning operations yield a uniform triangular surface mesh with overall excel-



lent triangle quality. Histograms showing the scaled Jacobian quality function for meshes decimated using *geniso* and *pyacvd* are shown in Figures II.9a, II.9b and II.9c, for the surface meshes of the bone, the plate and the screws, respectively. Statistical quantities related to the quality function are displayed in Table II.1. A significant improvement on the quality of elements generated using *pyacvd* can be seen over the ones generated by *geniso*. Most notably, one should note the absence of highly distorted elements (see minimum values of scaled Jacobian in II.1).

A major improvement of the quality of the screws and the plate is achieved, they previously contained heavily distorted elements resulting from the decimation with *geniso*. The single value of quality in the meshes of the screw stems from the fact that the previously implemented cleaning pipeline was not compatible with the right triangles in the original mesh of the screw, causing the cleaning operation to do nothing. Using *pyacvd* for remeshing and mesh decimation solves these abnormalities. The new meshes will likely be much easier to handle during the volume meshing part of the pipeline.

A small drawback of the cleaning operations is the slight rounding of edges on the surface mesh. This artifact is a result of the clustering operation, where the Voronoi regions go across the edge, resulting in vertices that make the edges less defined. For the mesh of the bone, the effect is negligible, as edges are not pronounced. These artifacts will be much more visible on the meshes of screws and plates created in the following step.

Scaled Jacobian	Bone		Plate		Screw	
	<i>geniso</i>	<i>pyacvd</i>	<i>geniso</i>	<i>pyacvd</i>	<i>geniso</i>	<i>pyacvd</i>
Mean	0.832	0.916	0.716	0.908	0.408	0.867
Median	0.845	0.925	0.739	0.916	0.411	0.875
Maximum	0.999	0.999	0.999	0.999	0.411	0.998
Minimum	0.169	0.672	0.018	0.664	0.357	0.5206
Area [mm <sup>2</sup> ]	15121	15428	2437	2507	377	417

Table II.1: Statistical quantities for the quality of elements in the surface meshes after decimation using *geniso* or *pyacvd*.

The rounding of edges is a minor issue, as the resulting meshes are still an upgrade compared to previously generated meshes in [9][10]. Indeed, the previous pipeline used multiple iterations of the smoothing filters found in VTK to combat badly conditioned meshes obtained after boolean operations, resulting in significantly a larger loss of volume in the final meshes of the geometrical objects(see Table II.1). Moreover, these artifacts tend to vanish when the mesh is refined; whereas the previous smoothing operations were applied regardless of the mesh size, causing systematic volume loss, even when meshes were extremely detailed.

This new mesh cleaning routine is capable of producing good surface meshes with much less effort than the highly specialized operations used in the previous projects, while maintaining good fidelity to the original geometry of the objects.

#### II.1.4 Creation of implants, drilling and cutting of bone surface mesh

A clean bone surface mesh is now available for further processing, either to drill and cut the bone to prepare it for the attachment of implants, or to create said implants using the geometry of the bone.

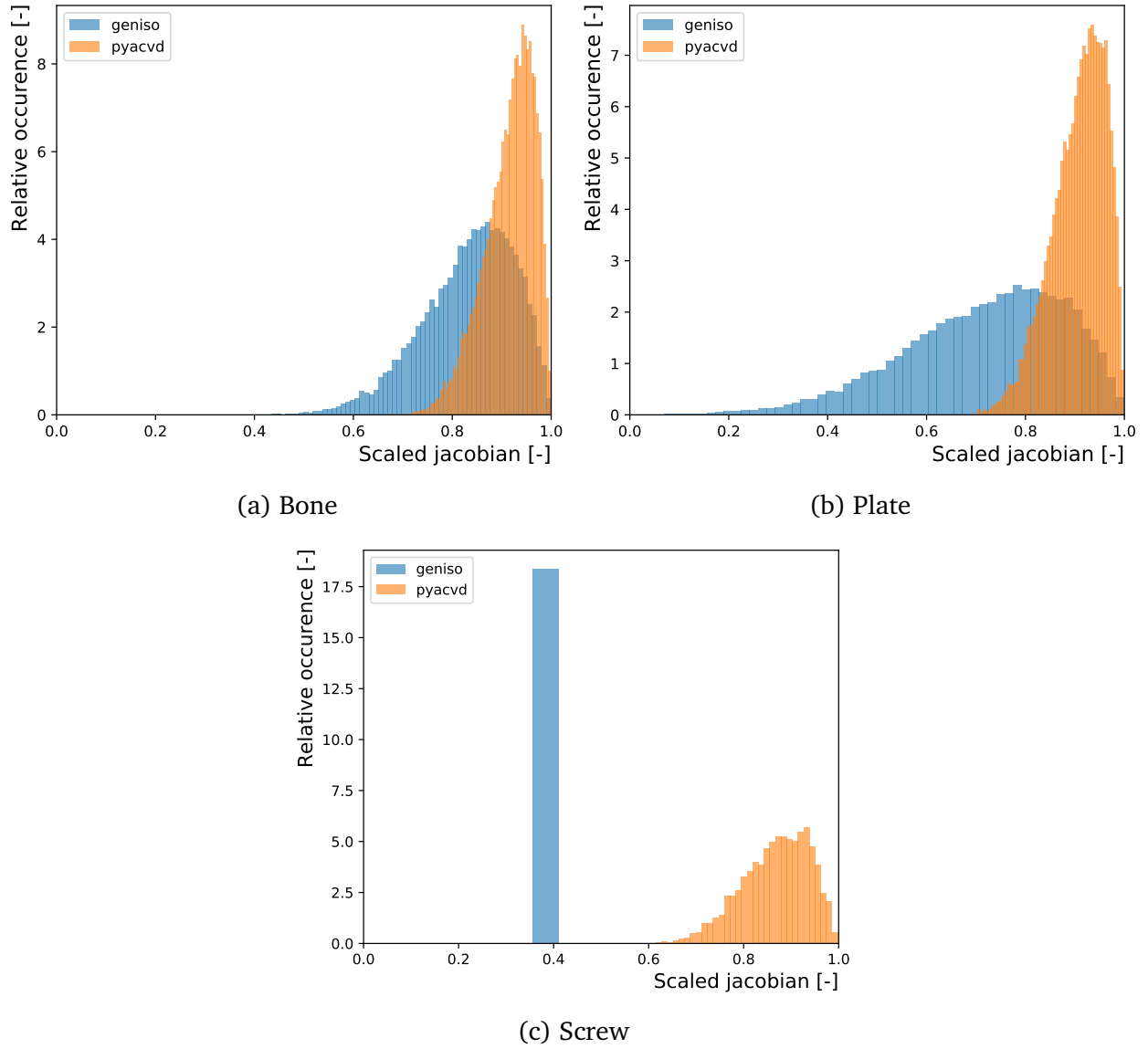


Figure II.9: Histogram of scaled jacobian quality function for surface meshes decimated using *geniso* or *pyacvd*.

### Creation of implants

To create the plate and screws that are adapted to the bone geometry, some geometrical characteristics need to be computed first. The most important characteristics for the following operations are the barycenter, the main orientation, and the length of the bone.

**Orientation of the plate** The plate is created such that the ends are not touching the bone. To this end, the profiles of the bone viewed from two orthogonal directions are needed (see Figure II.10). The 'top-view' is used to determine the orientation of the plate in its length-width plane and the side view is used to determine the offset and the orientation of the plate in its length-thickness plane. Using the offset and the orientation of the plate, one can create a placeholder plate.

In this work, the choice was made to work with a plate drawn using CAD software that better reproduces the geometry of a standard plate used in orthopedic applications. The

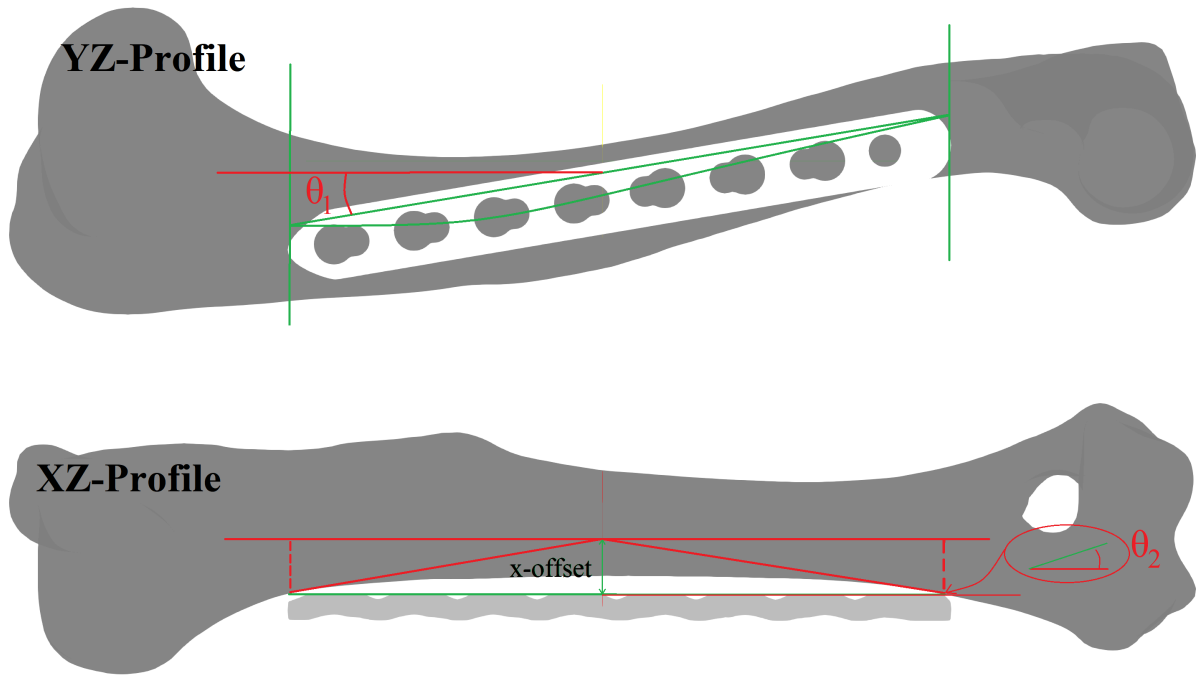


Figure II.10: Using the profiles of the bone to determine the orientation of the orthopedic plate.

orientation of said CAD generated plate is matched to the one of the placeholder plate. This orientation matching allows us to quickly swap between plate designs if needed in the future, as we only need to import a new plate surface mesh and match its orientation to the previously created plate.

**Screws** In the plate designs of the present work, only plates with 8 evenly spaced holes are considered. Knowing the length, which is given, and the orientation of the plate, which we computed in the previous step, one can find the orientation and location of screws that would match the plate design. The location of these screws are specified using a centerpoint and an orientation. The centerpoint is based on the spacing of commonly used plates in the orthopedic field, and the orientation is a normal to the length-width plane of the plate to which the screws will be attached.

An advantage of creating more modularity in the image-to-mesh pipeline is that the number of screws can be set more easily. Implants with the same or different number of screws on each side of a cut or fractured bone can essentially be configured at will, so long as the screws would 'fit' in the bone (in the sense that a free hanging screw at the cutting site would be pointless both from an orthopedic standpoint, as well as from a mechanical testing standpoint).

From the orientation and length of the plate, generated screws are pictured in Figure II.11.

After their creation, the screws and the plate are cleaned using the same combination of tools used for cleaning the bone mesh. Results from mesh cleaning operations is shown in Figure II.12. As mentioned previously, the artifact of rounding the sharp edges is more visible on the better defined edges of the plate and screws.

A note on the issue of having slightly rounded edges after cleaning operations: these

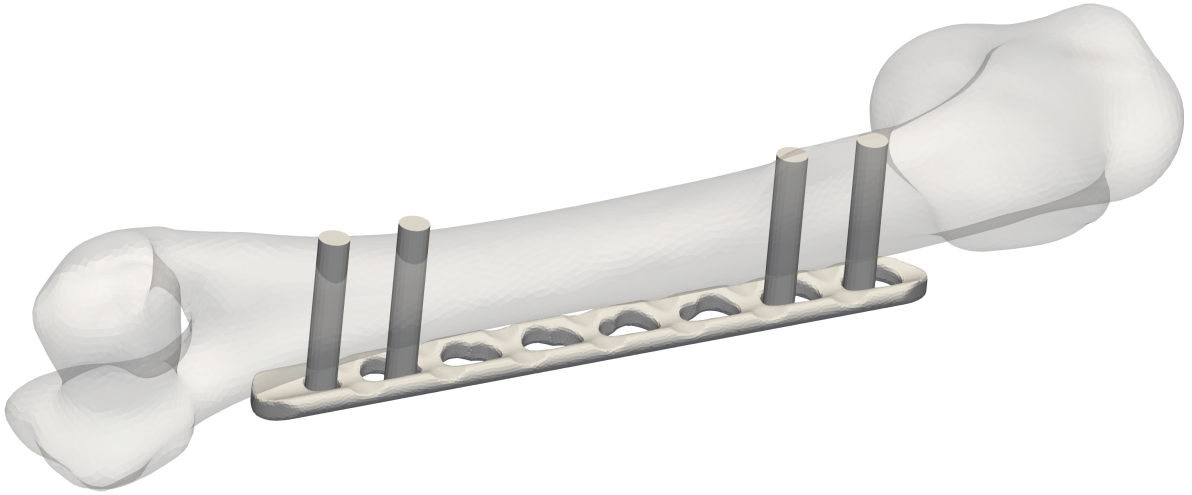


Figure II.11: Screws generated for a plate with given orientation.



Figure II.12: Surface meshes after cleaning.

issues can be entirely avoided on computer generated meshes by drawing the geometry using CAD software and integrating the holes and cuts directly, or generating the geometry parametrically using software such as *gms*. However, for meshes resulting from imaging techniques, such artifacts are quite difficult to avoid. In the framework of this project, as the software and tools list was already long, the choice was made to use previously developed geometry creation routines and applying the new cleaning pipeline to them. To obtain better final geometries for the plate and the screws, generating them directly using CAD software would be an improvement over the meshes shown in this work.

### Drilling and cutting bone

Operations that remove a piece of the mesh rely on boolean operations between two surface meshes. The first is the mesh from which a piece must be removed, and the second is a mesh with the shape of the to be removed part. For example, drilling a hole in the bone that fits a screw requires the bone mesh and the mesh of a screw. VTK's boolean difference operations are then used to remove the second mesh from the first, resulting in a drilled mesh.

**Drilling screw holes** Using the untreated meshes of the screws previously generated, the bone mesh is drilled using boolean operations. A visualization of the operation can be found in Figure [II.13](#).

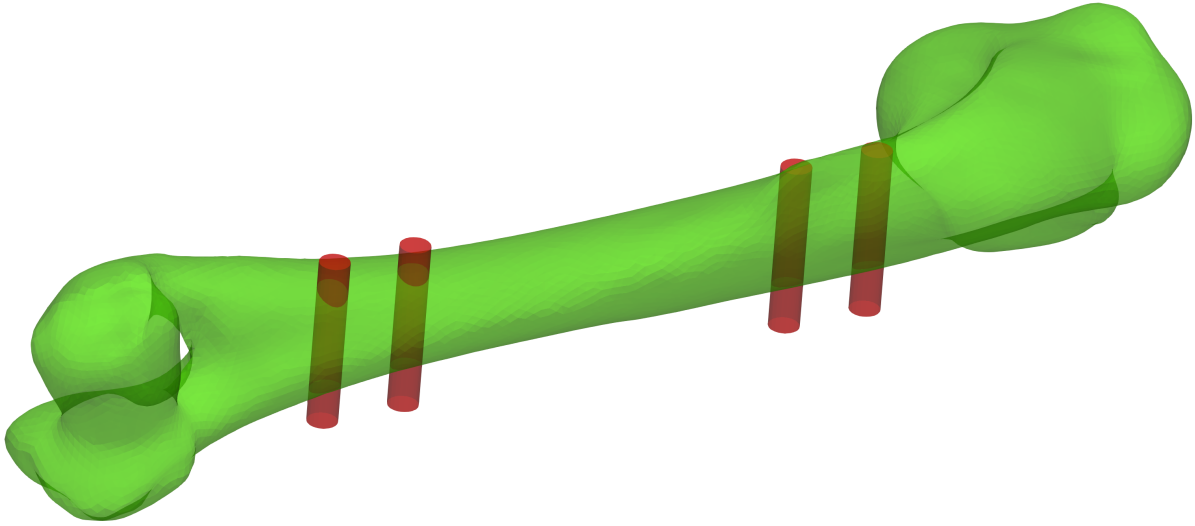
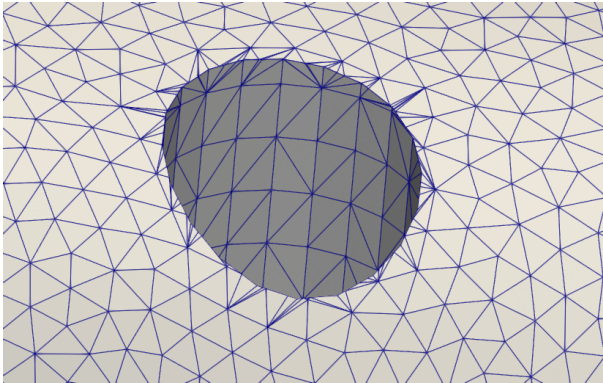
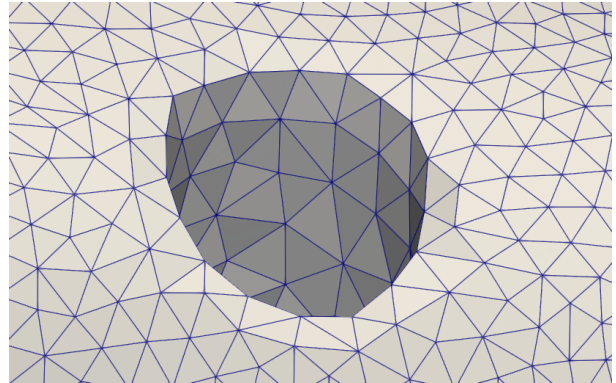


Figure II.13: Drilling bone with mesh of screws.

Cleaning the mesh is critical at this point, as boolean operations tend to leave heavily distorted triangles where the two meshes intersect. For the mesh of the drilled bone, re-meshing and mesh fixing operations using *pyacvd* and *pymeshfix* is performed. Close-ups of the meshes before and after cleaning are shown in Figure II.14.



(a) Before cleaning



(b) After cleaning

Figure II.14: Mesh cleaning of drilled bone

**Cutting bone** In the previously available model, the response of repaired cut bones was investigated; the image to mesh pipeline thus required a cutting operation. The cutting is done using a boolean difference between the bone mesh and a flat box placed normal to the bone's main axis. A visualization of the operation can be found in Figure II.15.

Similarly to the drilling operation, a cleaning pass is applied to the resulting mesh. Close-ups of the meshes before and after are shown in Figure II.16. Similarly to the screws and the plate, edge rounding artifacts are present in the resulting surface meshes.

In this work, as the response of cut bone is not the objective of the simulations, cutting is only performed to produce the meshes compatible with the previous works in [10].

Note that at this point, a file format conversion is made to convert .vtp files into a file format that is readable by *gms*h for the next operation, volume meshing.



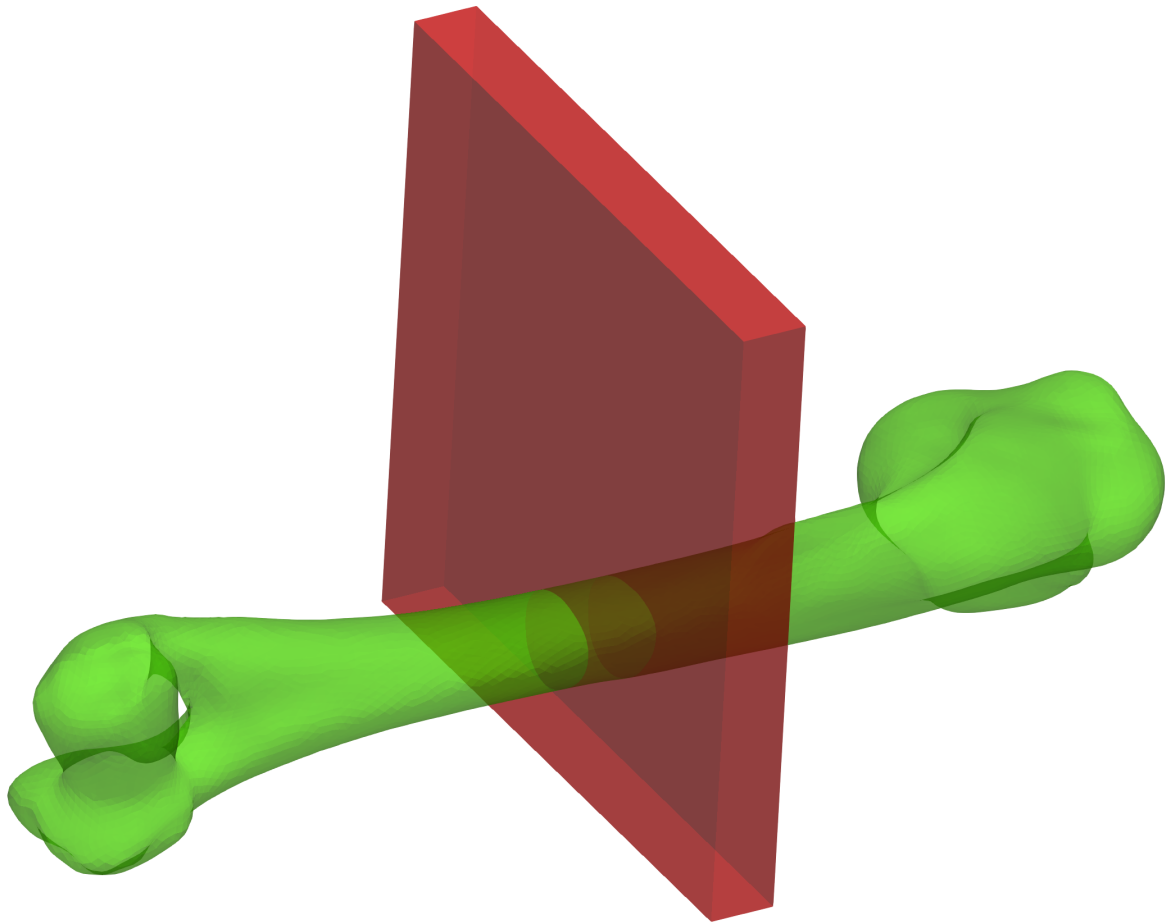


Figure II.15: Cutting bone with mesh of a flat box.

### II.1.5 Volume meshing

Volume meshing is done using *gmsh*[21] through its newly released Python API. Given the power of *gmsh*, the volume meshing operation is completed using very few commands. Requiring only the definition of a closed surface, the volume inside that surface and to specify whether mesh optimization steps should be carried after meshing, the volume mesh is swiftly created by *gmsh*. Naturally, one can exert more control over the meshing operation by specifying mesh sizes and exploiting the full power of *gmsh*, it was here intended more as a drop-in replacement for the previously used *TetGen*[22] tetrahedron generator that was used in [9][10]. For future developments, the use of *gmsh* provides more flexibility as it includes not only the algorithms used by *TetGen*, but also includes other algorithms that might be more suitable for surfaces extracted from medical imaging. At this stage, the inside volumes of the previously generated surface meshes are filled, a cut into the volume mesh of the bone is shown in Figure II.17.

As for the surface meshes, histograms for the quality of the generated volume meshes can be found in Figures II.18a, II.18b and II.18c for the volume meshes of the bone, the plate and the screws, respectively. Statistical quantities related to the quality of the volume meshes can be found in II.2.

The improvement of the meshes generated using *gmsh* can likely be explained by two factors. The first being the badly conditioned meshes obtained during the decimation and

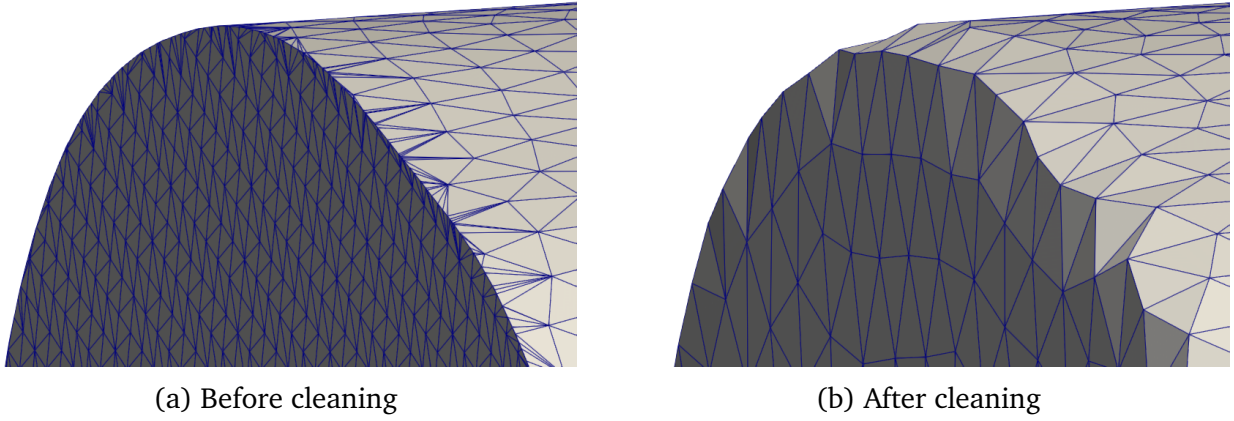


Figure II.16: Mesh cleaning of cut bone

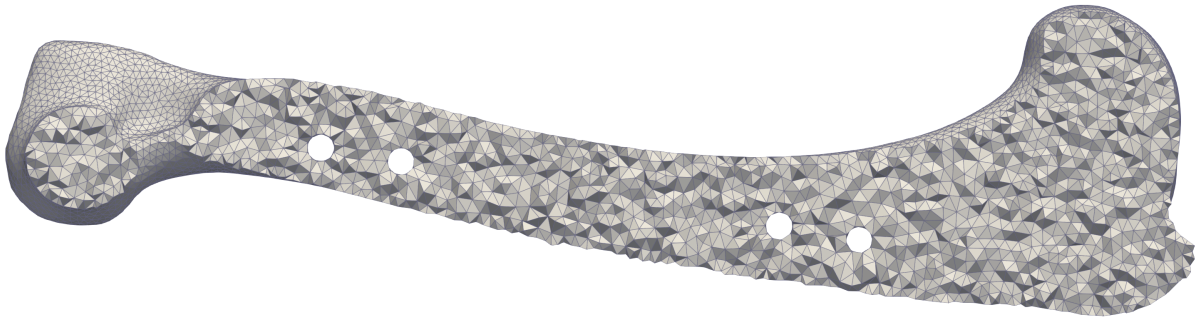


Figure II.17: Clip in the volume mesh, the volume is filled.

cleaning phase in the previous image-to-mesh pipeline. Indeed, the quality of elements on the surface directly influences the quality of volume elements near that surface: if a triangle is badly distorted, the tetrahedron containing that triangle cannot be magically improved during volume meshing without altering the surface in unexpected ways. The second factor is the inherently more powerful mesh generation algorithms implemented in *gmsh*. A more thorough analysis would mix-and-match the decimation and volume meshing algorithms to determine the effects that each step in the image-to-mesh pipeline has on the resulting mesh.

The resulting data type after meshing is stored in XML files with the `.vtu` extension. These files contain VTK's 'vtkUnstructuredGrid' data object, called 'unstructured grids' in the text. Unstructured grids are one of the most versatile data object in VTK, as they can represent any combination of any cell types supported by VTK. This includes 0D (points), 1D (lines), 2D (triangles, polygons) and 3D (tetrahedrons, hexahedrons, polyhedrons) entities. Accessing points or cells in the unstructured grid is done using indices and accessors that return a VTK point object or a VTK cell object depending on the entity that is requested. Similarly to `.vtp`'s polydatas, one can append fields related to the points or the cells of the grid. In this work, the data field that needs to be assigned first is the HU value of the cells, from which other material properties such as density, Young's modulus, yield stress, etc. can be deduced.

At this stage, the `.vtu` files only contain geometrical data, i.e. point and cell information. The following subsection covers the assignation of material properties to the unstructured grid.

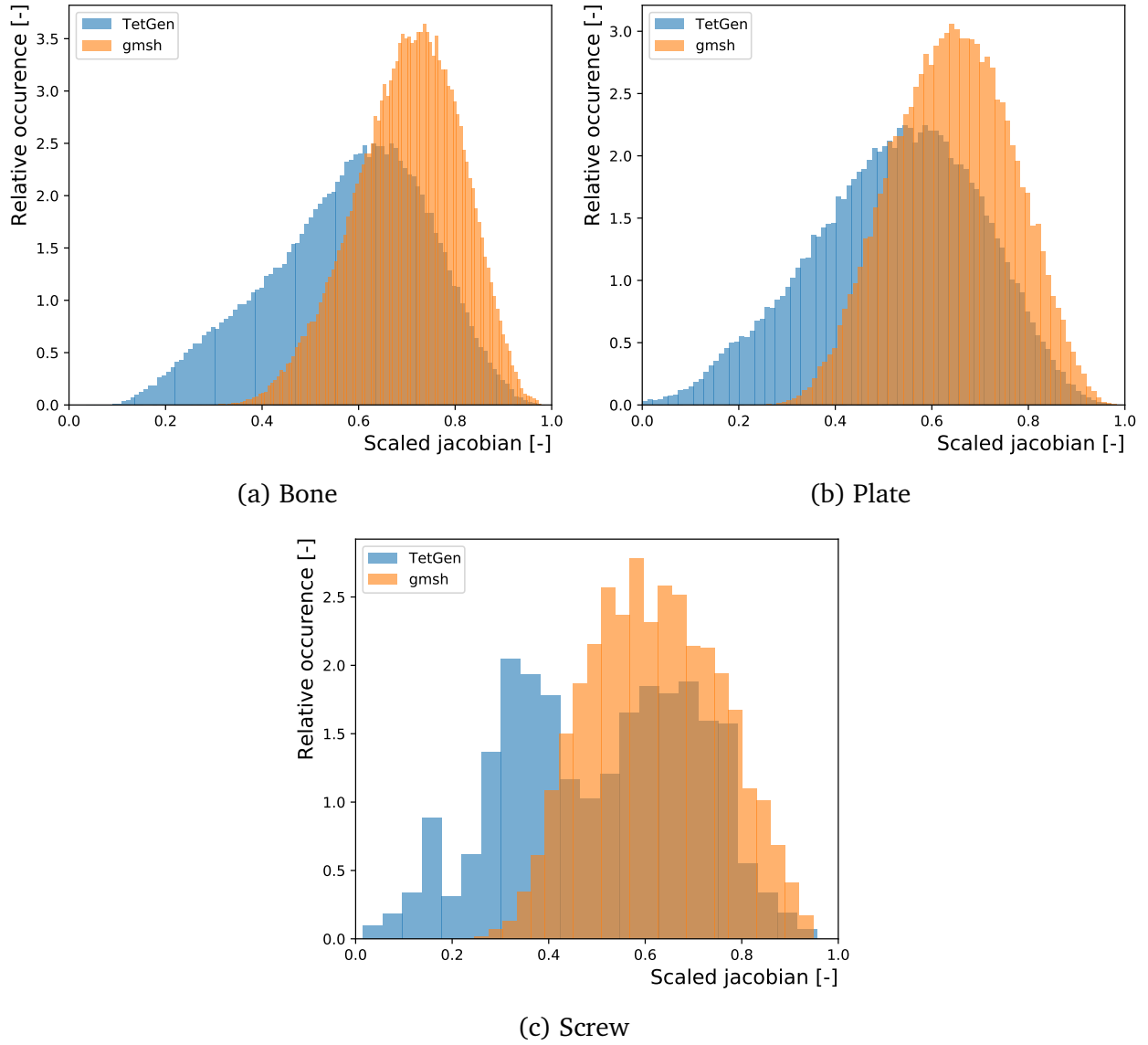


Figure II.18: Histogram of scaled jacobian quality function for volume meshes generated using *TetGen* or *gmsh*.

### II.1.6 Material property assignation to the volume mesh

The main idea behind the assignation of material properties is to transfer the HU values from the initial CT volume to the cells that have been created during the volume meshing process. After the information transfer, transforming the HU value of a cell into another cell field such as bone density is a matter of going through all the cells and applying a simple function, assuming the function linking the fields is given.

In other words, the information found at the points of a regularly spaced grid (points in the initial structured grid - CT volume) must be linked to cells that occupy a certain position in the volume (cells in the **UN**structured grid - unassigned volume mesh). The way it is done in this work starts by going through each cell of the volume mesh and computing the bounding box of said cell. Knowing the boundaries of the bounding box, the points of the CT volume that are situated in the volume of the cell bounding box are selected. The HU values of the points encompassed by the cell bounding box are averaged and this average

Scaled Jacobian	Bone		Plate		Screw	
	<i>TetGen</i>	<i>gmsht</i>	<i>TetGen</i>	<i>gmsht</i>	<i>TetGen</i>	<i>gmsht</i>
Mean	0.577	0.702	0.525	0.643	0.507	0.621
Median	0.595	0.709	0.536	0.544	0.521	0.617
Maximum	0.987	0.985	0.983	0.993	0.956	0.948
Minimum	0.034	0.254	0.0004	0.257	0.014	0.246
Volume [mm <sup>3</sup> ]	69862	71050	2049	2141	378	411

Table II.2: Statistical quantities for the quality of elements in the volume meshes generated by *TetGen* or *gmsht*.

value is then assigned to the current cell.

An observation to be made at this point is that this averaging process can induce a slight smoothing of the information in the CT volume. Indeed, as the bounding boxes in which the points of the CT volume are selected have a larger volume than the tetrahedral cell from which they are computed, information that should be outside of the cell is brought inside the cell, ultimately creating a slight smoothing the HU field when going from the CT volume to the volume mesh. This effect can be even more significant when the initial volume itself has blurring at regions where density varies intensely, e.g. at the surface of the bone where there is the bone-air interface. Blurring effect that occur due to this smoothing at the surface of the bone are called partial volume artifacts. In the case where meshes are coarse, partial volume artifacts can negatively affect the resulting volume mesh by creating a layer of spuriously weak material at the surface of the bone.

Another note is that this information transfer imposes an upper bound to the fineness of the volume mesh, as elements can only be as small as the largest resolution between two adjacent points in the initial CT volume. However, this upper limit is never reached in the context of this work, as the required number of elements would be approximately 45 million (one per point in the CT volume).

After assigning a HU value to each cell, derived properties are computed, first using the HU-density relationship developed in [9] in equation II.1, where  $\rho$  is in [g/cc]. From the bone density, a relationship can be found with Young's modulus in II.2, this relationship is derived in [23], where  $E$  is in [MPa] and  $\rho$  is in [g/cc].

The yield stress of bone is also extracted from [23] and is found in II.3

$$\rho = f(\text{HU}) = 4.93323 \times 10^4 \text{HU} + 0.9839 \quad (\text{II.1})$$

$$E = f(\rho) = 2065 \rho^{3.09} \quad (\text{II.2})$$

$$\sigma_y = f(E) = 0.0073E \quad (\text{II.3})$$

In this work, only isotropic material properties are considered due to limitations in the determination of local orthotropy axes in the available toolboxes, even though bone has widely been described in the literature as a locally orthotropic material[24]. Due to the exploratory context of this work and given the complexity of material laws discussed in the next chapter, the choice was made to limit ourselves to isotropic materials. Naturally, introducing a proper computation of local orthotropy axes and assigning locally orthotropic material properties would be an upgrade over the current model.

These material properties are then assigned to the unstructured grid to the corresponding cell from which the HU value was selected. A cut into the volume mesh is displayed in Figure II.19, using Young's modulus to color the cells.

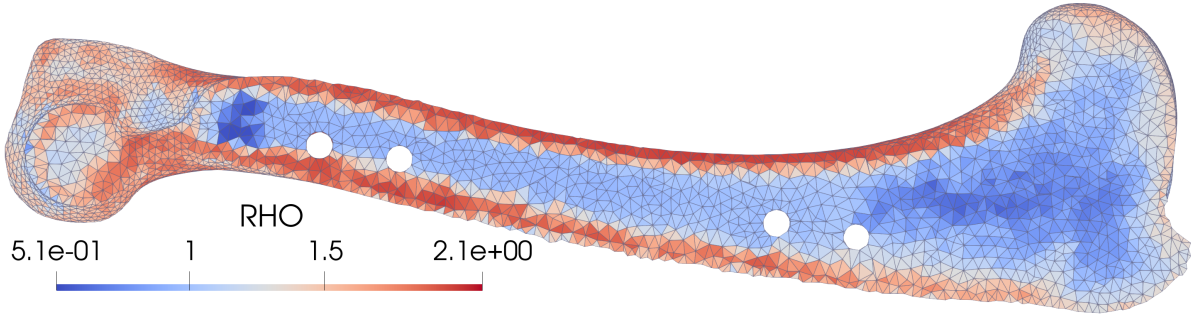


Figure II.19: Clip in the volume mesh, coloring based on local density in [g/cc].

Another specificity to this work that was not present in the previous models ([9][10]) is the presence of a defect in material property where the cut was made in [10], modeling a hypothetical blood clot that would have a stiffness close to zero. This defect is created by selecting all cells present in a region close to the midpoint of the bone, and selectively assigning an extremely low value for the Young's modulus to these cells. The goal of this assignation of extremely low material property instead of making a cut is to gain the possibility of selecting special material laws to model the healing behavior of the clot. Materials being assignable only where there is a geometric support, a region of low material property is chosen over a geometric defect. A visualization after the creation of the clot is shown in II.20.

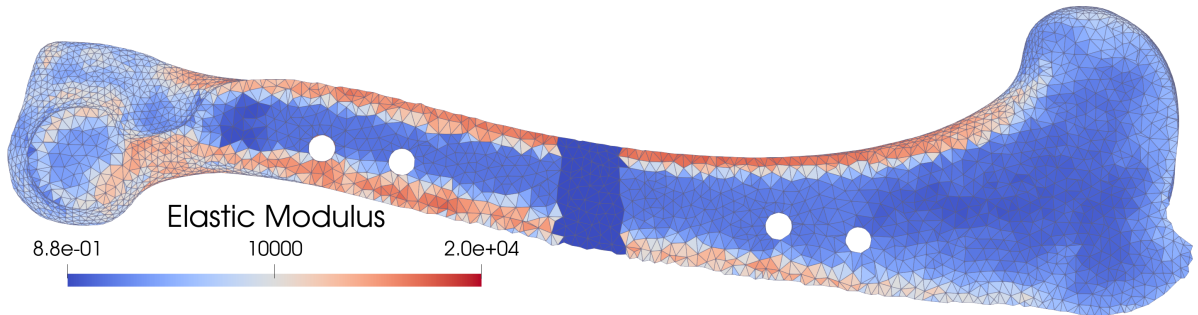


Figure II.20: Clip in the volume mesh, with defect in material properties, coloring based on Young's modulus in [MPa].

With these material properties assigned, the volume meshes of the bones is considered complete. A summary of the mesh characteristics can be found in Tables II.3 and II.4 for the geometrical and material characteristics, respectively.

$m$	Bone			Plate			Screws		
	N. nodes	N. cells	Volume	N. nodes	N. cells	Volume	N. nodes	N. cells	Volume
0.1	7255	2104	67004	1968	680	1549	250	99	313
0.2	18157	4789	68640	4050	1308	1866	439	181	368
0.3	31559	7881	69204	5385	1836	1967	716	278	388
0.4	47127	11324	69485	6955	2381	2022	1152	396	397
0.5	63786	14967	69652	8770	2966	2054	1607	519	404
0.7	100375	22807	69846	13527	4297	2091	2389	745	411
0.9	142548	31543	69953	18807	5728	2111	3316	995	415
1.1	188237	40894	70018	24677	7252	2124	4308	1248	417
1.5	296314	62360	70098	36884	10379	2140	6782	1848	420
2.0	428401	88756	70161	53560	14514	2151	9820	2583	422

Table II.3: Summary of mesh characteristics at the end of image-to-mesh pipeline. Volumes in  $[\text{mm}^3]$ .

$m$	HU			Density			Young's Modulus		
	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
0.1	-811	666	2011	0.58	1.31	1.97	467	6356	17066
0.2	-890	730	2057	0.54	1.34	1.99	346	6786	17653
0.3	-921	738	2096	0.52	1.34	2.01	301	6836	18143
0.4	-950	739	2096	0.51	1.34	2.01	267	6848	18115
0.5	-958	734	2116	0.51	1.34	2.02	262	6820	18412
0.7	-975	737	2151	0.50	1.34	2.04	247	6830	18919
0.9	-972	717	2199	0.50	1.33	2.06	250	6710	19542
1.1	-988	716	2187	0.49	1.33	2.06	237	6696	19392
1.5	-991	696	2221	0.49	1.32	2.07	235	6559	19922
2.0	-1000	682	2309	0.49	1.32	2.12	228	6473	21143

Table II.4: Summary of material properties contained in the intact bone meshes at the end of image-to-mesh pipeline. Densities in  $[\text{g/cc}]$ , Young's modulus in  $[\text{MPa}]$ .



## II.2 Numerical Simulation in *Metafor*

This section presents details and results from numerical simulations in *Metafor*. Specific considerations related to the use of previously generated meshes are mentioned; e.g. the importation of VTK files into *Metafor* or the use of groups to emulate continuously varying material properties in the bone. Modeling assumptions regarding materials and boundary conditions are briefly described.

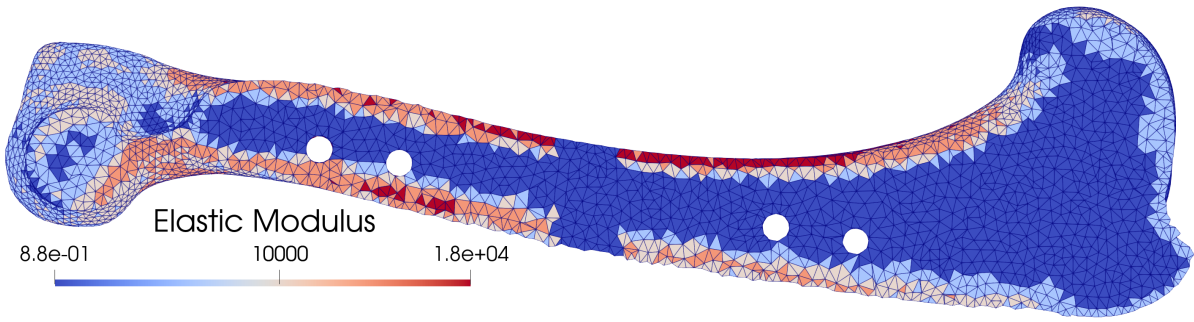
A special consideration will be given to the management of plate-screw and screw-bone contact. An alternative to the previously implemented over-constraining spring-based method is developed and a study is done to determine adequate parameter values. This alternative method should be seen as a temporary solution before proper contact management can be done using sticking contact conditions. Unfortunately, these sticking contact conditions could not be implemented due to numerical challenges in the context of this work.

### II.2.1 Importing VTK meshes into *Metafor*

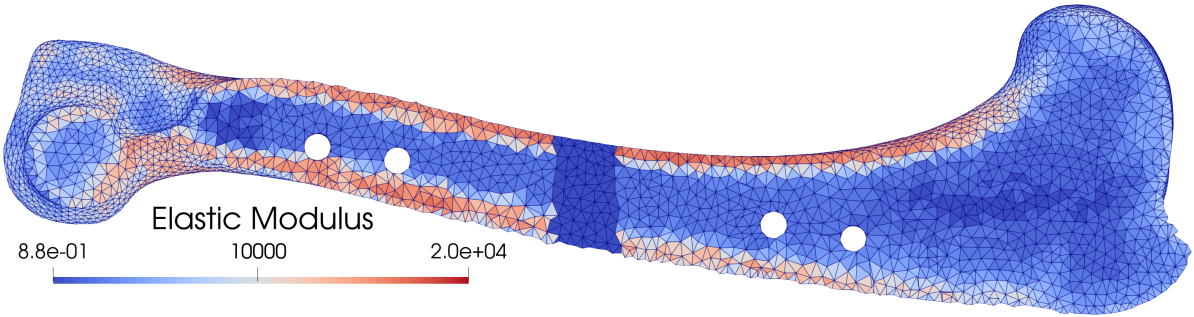
Importing purely geometrical VTK volume meshes (i.e. with no data arrays appended, in this project, the screws and the plate) is straightforward, the points and cells of the mesh are read one by one and an equivalent copy is created within *Metafor* using point coordinates for the points and the indices of the points composing a cell for the cells.

For meshes with appended data arrays (in this project, the different bone meshes), the import process is more complex. The ultimate goal after importing the mesh is to create materials in *Metafor* that mimic the geometric distribution of material properties found in the VTK volume mesh as closely as possible. In *Metafor*, it is not feasible to define a material with varying properties but one can instead create groups of cells that have different properties. The idea behind the creation of a bone with "continuously" varying material properties is to create a large number of groups of cells that have the same properties within a group, but different properties between groups.

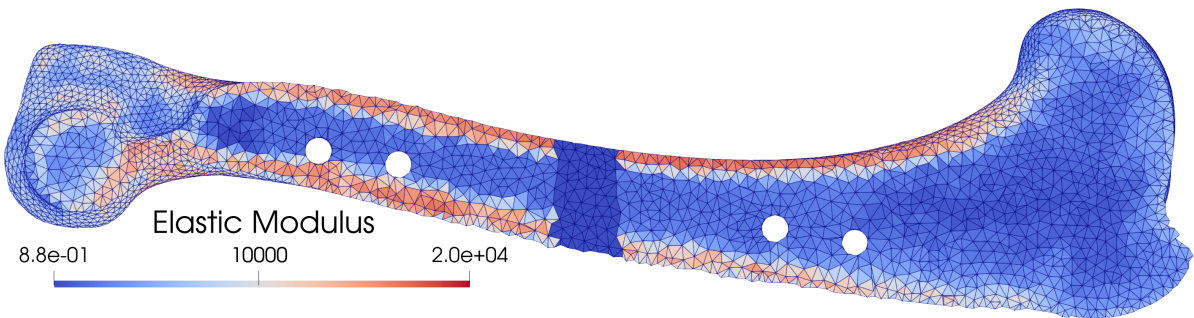
Following this intention of binning the material properties, the data arrays appended to the volume mesh containing the material property fields are read first. These properties are then separated into  $n$  groups, each encompassing an interval of property values (i.e. the width of each interval is  $\frac{\max \text{prop} - \min \text{prop}}{n}$ ). A loop over the cells then assigns each cell a region number (between 1 and  $n$ ) indicating which group they belong to depending on the value of the property in the cell. After separation, each of these regions will be assigned the center value of the property interval as their property value. At this stage, the proper import starts by going over all points in the volume mesh and adding them to the *Metafor* point set, followed by the cells, which are created and added to their corresponding group in *Metafor* depending on their assigned region number. After the import, it is now possible to assign different properties to different selections of cells, thus mimicking the continuously varying material properties of a bone using a discrete number of groups. Note that at this point, the number of bins ( $n$ ) is the conditioning factor behind the 'smoothness' of the properties in the bone. The results of the import using different numbers of bins are shown in Figures II.21a, II.21b and II.21c, where 5, 25 and 50 bins are used, respectively. A clip into the full volume mesh is displayed in Figure II.21d. Using 50 bins, virtually no difference can be spotted. In subsequent simulations, the number of bins will be fixed at 50.



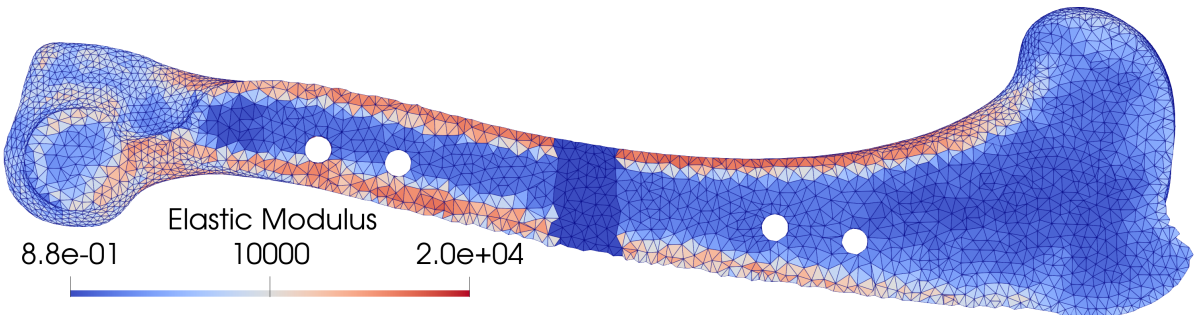
(a) 5 bins



(b) 25 bins



(c) 50 bins



(d) Original volume mesh

Figure II.21: Resulting distribution of material properties in the mesh after import with different number of bins.



## II.2.2 Modeling assumptions

### Material Laws

Following the models developed in [9][10], for the most part, the choice of material laws has not changed. Most of the bone is still modeled using an elasto-plastic material exhibiting isotropic linear hardening after yielding, with a post-yield modulus set at 5% of the local elastic modulus. The elastic modulus values can be deduced from bone density values using equation II.2.

The material chosen for the plate and screws is orthopedic stainless steel, modeled using linear elasto-plastic material with an elastic modulus of 180 [GPa] and a Poisson's ratio of 0.3, the yield stress being set at 290 [MPa] and post-yield hardening set to 5% of the initial elastic modulus.

### Boundary conditions

The boundary conditions have been kept similar to the ones used in [10], i.e. the main loading scenario is still bone compression along its axis. However, instead of piloting the compression using planes in contact with resin cubes capturing the epiphysis (i.e. the extremities) of the bones, the choice was made to impose the displacements directly on the epiphysis. The motivation behind this choice was two-fold. Firstly, the handling of contact between the resin blocks and the bone was done using a set of virtual springs, as it will appear in the following sections, the handling of contact using springs is a delicate balance that depends both on the number of nodes to which those springs are attached to and on the stiffness of said springs. Refining the bone mesh increases the number of springs attaching the bone to the resin blocks, causing an over-constrained situation where the bone is virtually undeformable where it is captured by the resin blocks. Secondly, these springs connect each node of the resin blocks to every other nodes of the bone epiphysis. In total, the number of springs almost matches the number of nodes on the entire bone surface itself, causing large computational efforts to be made to compute the deformation of the springs. Imposing the displacement directly on the epiphysis saves the computational load and frees the epiphysis from an over-constrained grip.

### Contact management

While the spring-based "contact" between the resin blocks and the bones (see Figure 1 in [10]) have been removed, screw to plate and screw to bone contact is still managed using springs connecting each node of one surface to every node of the second surface. This makes the contact management mesh-dependent, as the rigidity of the contact thus depends on the number of nodes defining the surfaces of the geometric objects. Moreover, these springs tend to over-constrain the objects around which they are wrapped. This results in screw heads in contact with the plate and screw bodies in contact with the bone that apparently carry no load, giving questionable trust in the stress and strain fields reached within those objects. Ideally, the contact should be managed using sticking contact conditions, to which penalties are attached to normal and tangential penetrations, blocking the screw head inside the plate or the screw body inside the bone. Unfortunately, these sticking contact conditions could not be implemented in the context of this work due to numerical challenges. However, an improvement to the spring-based solution is proposed

in the following sections, which tries to address the issue of the over-constraining of objects wrapped in springs.

### Plate and screws number choice

Like in the previously developed models, the base model of the bone is repaired using two screws on each side of the defect. The same CAD generated plate design as in [10] is selected and placed with a 2 [mm] offset to the bone. In the chapter exploring fracture healing, different combinations of screw numbers and plate offsets will be tested to see if the healing response can be improved by modifying the screw number and arrangement.

## II.2.3 Simulation results

Using the modeling assumptions and the previously generated volume meshes, the results of a series of numerical simulations are shown here. A typical end result of a simulation is displayed in Figure II.22, with coloring based on the equivalent Von Mises stress (EVMS). In this chapter, only 'mechanical' simulations are performed, in the sense that the bone is not allowed to remodel. The time dependence of load application is thus irrelevant here as long as the numerical time steps are chosen sufficiently small to capture the loading curve (which itself is a simple ramp from 0 to the prescribed displacement). A first round of

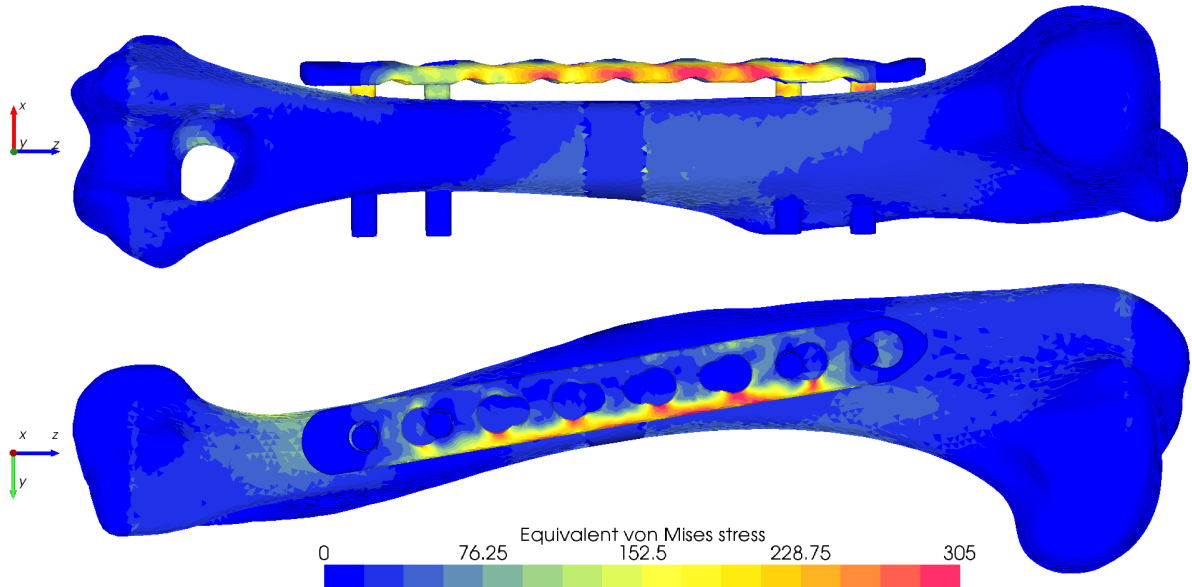


Figure II.22: Equivalent Von Mises stress field.  $m = 0.7$ . End-to-end displacement 0.5 [mm], springs stiffness  $K = 10^4$  [N/mm], other parameters as described in modeling assumptions.

simulations is performed to determine the required mesh density factor in order to obtain reliable simulation results. The total force required to create a 0.5 [mm] end-to-end compression of the bone is shown in Figure II.23. Results for the applied force stabilize when the mesh density factor reaches 0.7-0.8.

In regard to computation times, simulations done to create Figure II.23 can take from 30 seconds for low  $m$  to almost 2 hours for higher  $m$  on a 4-core laptop-class machine. Computation times for each simulation are shown in Table II.5.

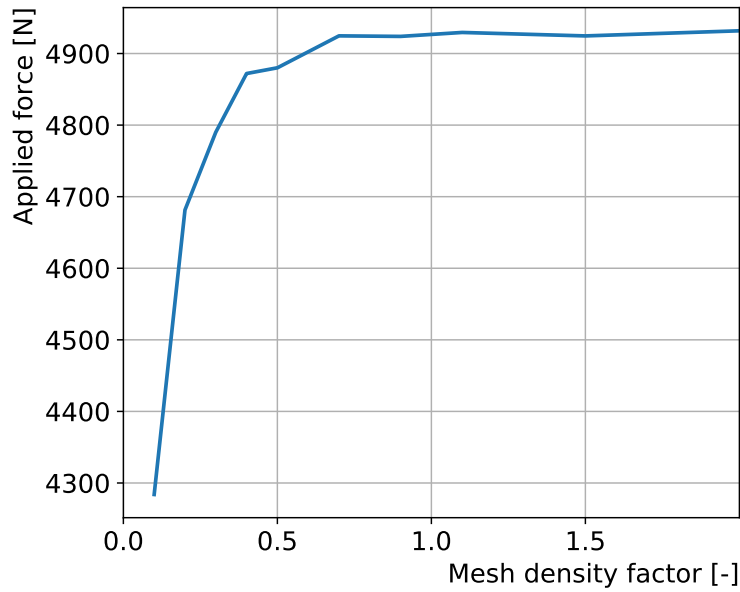


Figure II.23: Applied force at epiphysis for varying mesh density factors  $m$ .

$m$	0.1	0.2	0.3	0.4	0.5	0.7	0.9	1.1	1.5	2
Time (in s)	27	63	111	198	284	741	989	1452	3693	7012

Table II.5: Computation times for different mesh density factors

Maximum equivalent Von Mises stress reached within the objects are displayed for each simulation in Figures II.24a, II.24b and II.24c.

Equivalent Von Mises stresses follow an expected progression through the use of more elements. The only exception is the high stresses reached in the first simulation in Figure II.24b, which upon closer inspection(see Figure II.25), were due to the mesh having a too low polygon count and thus creating a region of high stress concentration.

Additionally, differences may be explained by the finer attribution of material properties when using finer meshes during the process described in section II.1.6. A finer mesh displaying less blurring of the original properties found in the CT volume. A consequence of that is that peak properties (e.g. maximum and minimum density) in the CT volume are reproduced with more fidelity, often resulting in regions with higher material properties.

Another factor amplifying the potential error in that simulation is the interpolation error induced by larger elements, as values from integration points must be interpolated further to reach the nodes, where the stresses are extracted for plotting. The EVMS results also indicate that a mesh density factor around 1.0 is sufficient to capture the stress field in the structure, with the exception of the plate mesh, where additional refinement might be necessary due to the presence of more significant hardening effects.

While maximum equivalent Von Mises stresses show expected progressions through the refinement of the mesh, a closer look at the distribution of the stress field on some of the objects reveals some issues. For example, looking at the stress field in one of the screws (see Figure II.26) reveals that some parts of the screw seem entirely stress free.

This is in stark contradiction with expectations, the screws would be expected to show a non-zero stress/strain field everywhere they are in contact with the bone and the plate, not only where they are free to shear.

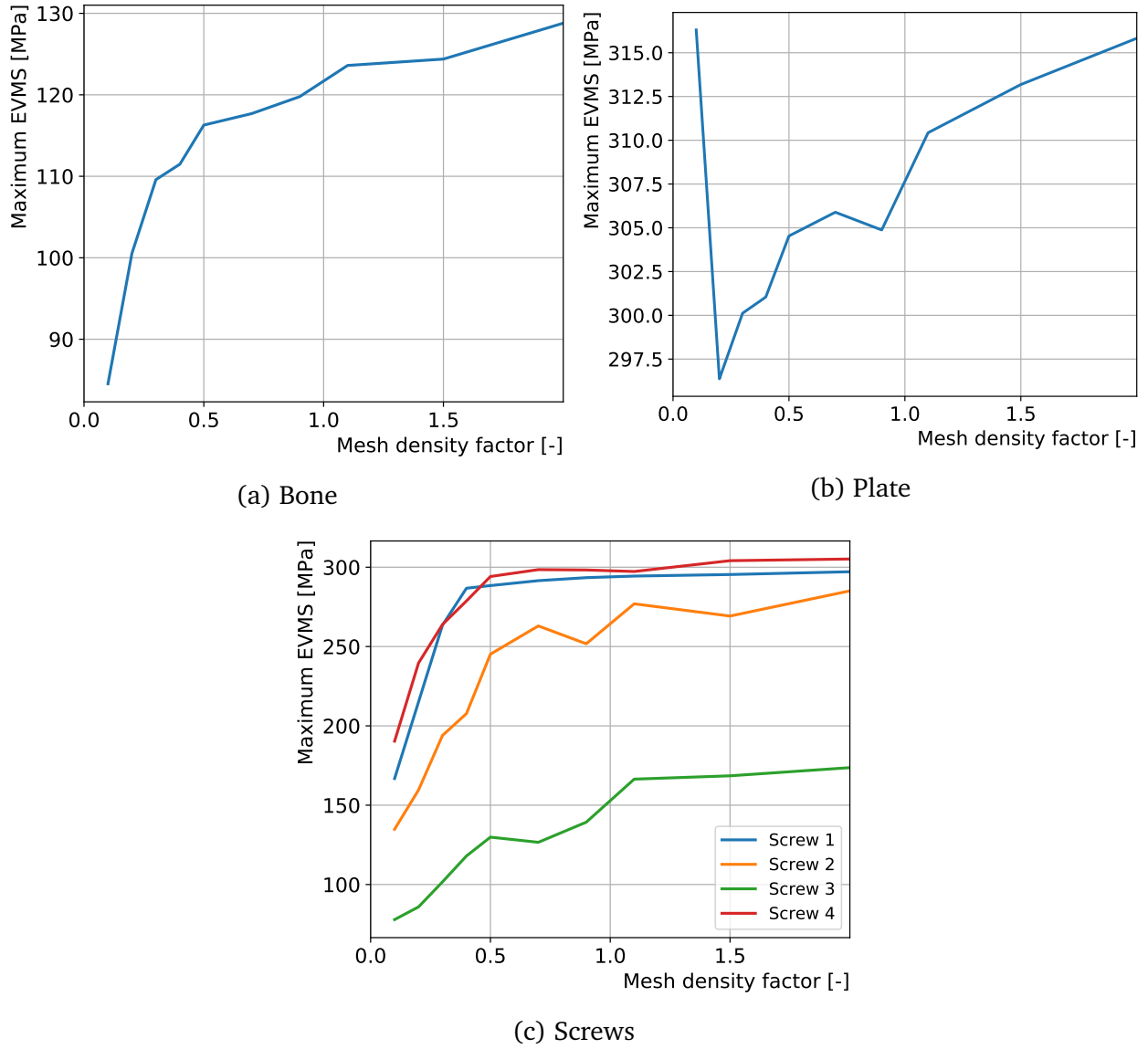


Figure II.24: Maximum equivalent Von Mises stress reached in each object, varying the mesh density factor  $m$ .

The distribution of the stress field of the screws displays exceptionally low values in the regions where they are in contact with the bone volume or the plate, as if they were prevented from deforming. These distributions are unexpected, as the contact surface between the screws and the other bodies should be one where stresses are present due to the screws being captured by the contacting bodies. Physically, the binding mechanism between the screw and the plate is done using locking screws, meaning the screw head has a thread that screws into the plate to prevent the screw from sliding out of the plate. This results in a geometrical configuration where the screws are 'locked' into the surgical plate. This locking should induce contact stresses at the regions where the screw is slotted into the plate, which are absent in the current simulations. This absence is likely explained by the management of contact that is implemented in the previous models [10].

When looking at how the contact is defined between bone and screws or the screws and the plate, one can infer that the current spring-based contact method might be the

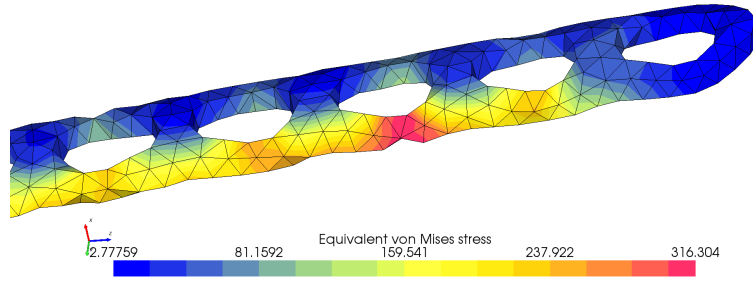


Figure II.25: Equivalent Von Mises stress field of plate, mesh density factor  $m = 0.1$ .



Figure II.26: Equivalent Von Mises stress field of screw 3, mesh density factor  $m = 0.7$ .

cause. Indeed, the sticking contact between surfaces is currently done by creating springs connecting the surface nodes of one object (e.g. a screw) to the socket where the screw is slotted in (whether it is the plate or the bone). These springs connect every surface node of one object to every surface node of the second object. While such a contact management could work in a situation where sticking contact is needed between two flat planes, even though there are now better ways to manage contact in such cases (see the defo-defo contact interactions in *Metafor* for example), it creates a surprising constriction effect when used on concave-to-convex contact situation. This situation arises here, where a cylindrical screw sits in a hole in the bone or plate.

The problem of a spring-based solution connecting each node to every other creates unwanted interactions between nodes that should never enter in contact with each other. Schematically (see Figure II.27), a screw node situated at one edge of the screw cross-section is virtually linked to nodes on the surface of the socket that are on the edge diametrically opposite to the screw node, spuriously increasing the rigidity of the contact. The problem is that the rigidity is not based on physical parameters such as the spring stiffness, but simply on the presence of long range spurious interactions between each screw node and every node of the socket surface. Note that in the schematic view, only a cross-section of the screw is shown, the nodes of the surface of the screw are also linked to surface nodes of the socket that are outside the plane of the schematic view.

Moreover, an additional problem arises when more elements are used when performing mesh refinement. Due to additional nodes being selected in the regions of contact, more springs are virtually created, and the rigidity of the contact increases due to mesh refinement.

## II.2.4 Improving the management of contact

As previously evidenced, the current spring-based contact management displays spurious stress and strain fields where the geometrical objects are wrapped in springs. In this sub-

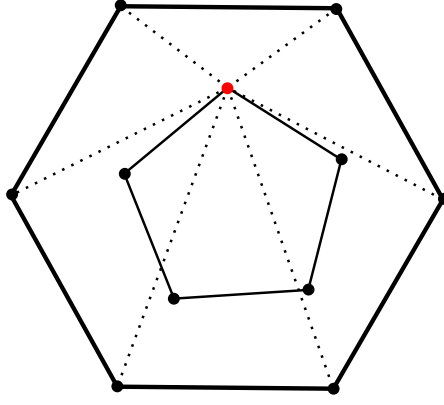


Figure II.27: Schematic representation of the spring-based contact. Plain lines indicate the surface of an object. Dotted lines indicate linking springs. The linking springs are only shown for the node in red for clarity.

section, an attempt at creating an improved spring-based contact management system is made. To combat the over-constraining problem, instead of having each surface node of the first object being virtually linked to every other surface node of the second object, each node on the first object is only linked to the  $n_S \in \mathbb{N}^+$  closest nodes in the second object. This way, the linking of nodes diametrically opposite of each other is hopefully prevented, and contact is once again defined using localized interactions.

The new contact management model is schematically depicted in Figure II.28, for  $n_S = 1$  and  $n_S = 3$ . The linking springs are here shown for every node, as there is now a reduced number of them.

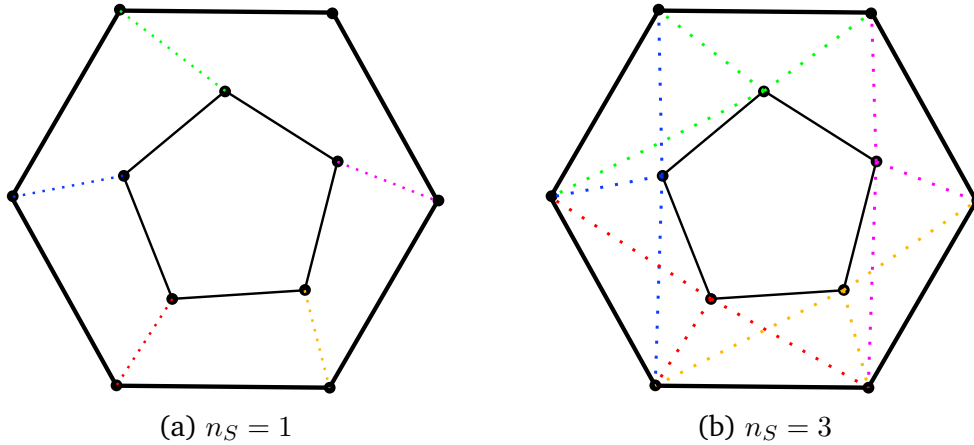


Figure II.28: Schematic representation of the newly implemented improved spring-based contact. Plain lines indicate the surface of an object. Dotted lines indicate linking springs. The springs are colored based on the node of the first object they are attached to.

In the previous contact model, the only parameter that could influence the results of contact was the stiffness of springs  $K$  linking nodes to each other. However, as the number of springs linking the two objects was extremely high, even a low spring stiffness had the effect of constraining the screws such that the stress and strain fields were close to zero where contact was set to occur. In other words, the value of the spring constant did not matter.

Using this improved version, the spring stiffness hopefully becomes a meaningful parameter again, setting the rigidity of the sticking between points close to each other but belonging to different geometrical objects. In addition to the stiffness of springs, the number of nodes of the second object to which a node from the first object is attached to is the other main parameter controlling the contact interaction.

In the next subsection, a parametric study on a simplified geometry is made to determine adequate values for the newly introduced parameters.

### II.2.5 Sensitivity study of parameters in contact management

To determine adequate values for the spring stiffness and the number of springs attached to each mesh point on the surface, a reduced geometry of the contact at the plate is created. The geometry is created and then meshed using *gms**h*, with the element edge length  $l_e$  being given as a parameter to the geometry generation script. The geometry is then imported into *Metafor* for simulation. The geometry outline is shown in Figure II.29, it is supposed to represent a single screw in contact with the plate.

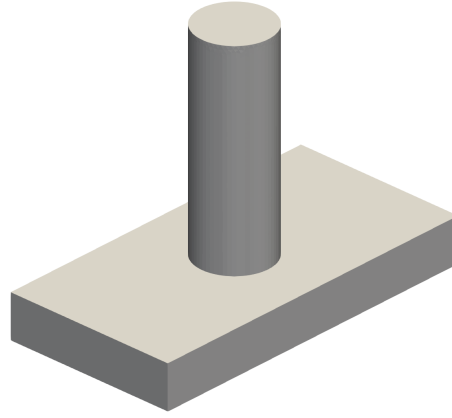


Figure II.29: Geometry used for contact sensitivity study.

The materials assigned to the objects are the same as in previous sections, i.e. orthopedic stainless steel, modeled using a linear elasto-plastic material with an elastic modulus of 180 [GPa] and a Poisson's ratio of 0.3, the yield stress being set at 290 [MPa] and post yield hardening set to 5% of the initial elastic modulus.

Boundary conditions consist in constraining the plate along its thickness axis, and blocking lateral movements along its length and width axes. The screw is in contact with the plate using the aforementioned improved contact condition. The simulations made in *Metafor* consist of pushing the screw inside the plate by 0.01% of the length of the screw, this low displacement value is set to prevent any hardening effects.

The parameters being tested here are the mesh density, the stiffness of springs  $K$  and the number of springs  $n_S$  linking each node of the first object to nodes of the second object.

Results from a battery of tests varying each parameter can be found in Figures II.30, II.31 and II.32. The element edge length  $l_e$  takes values in  $\{0.25, 0.50, 1.0, 1.5, 2.0\}$  [mm], the spring stiffness  $K$  takes values in  $\{10^2, 10^4, 10^6, 10^8\}$  [N/mm] and the number of springs per node  $n_S$  takes values in  $\{1, 3, 5\}$  [-].

Results from simulations using the over-constraining contact management method for the same geometry are displayed in Figure II.33 for comparison purposes. In these sim-



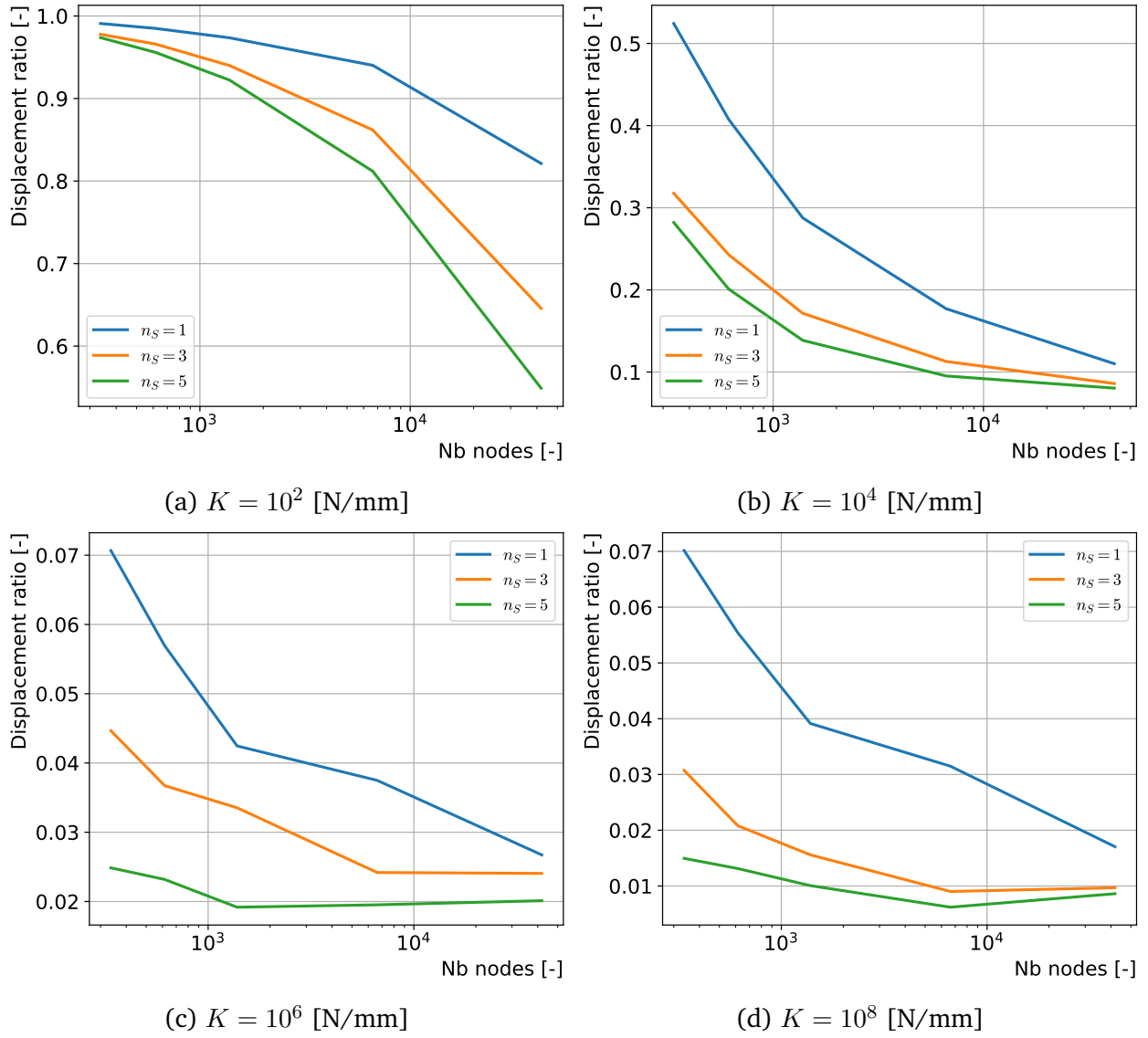


Figure II.30: Displacement ratio between the top and bottom surface of the screw. Element edge length  $l_e$  in  $\{0.25, 0.50, 1.0, 1.5, 2.0\}$  [mm], spring stiffness  $K$  in  $\{10^2, 10^4, 10^6, 10^8\}$  [N/mm], number of springs per node  $n_S$  in  $\{1, 3, 5\}$  [-].

ulations, the element edge length  $l_e$  takes values in  $\{0.50, 1.0, 2.0\}$  [mm] and the spring stiffness  $K$  takes values in  $\{10^2, 10^4, 10^6, 10^7\}$  [N/mm].

The displacement ratio is defined here as the ratio between the displacement of the bottom surface of the screw and the displacement of the top surface of the screw. The reasoning behind the choice of this indicator stems from the need of a sufficiently rigid contact to accurately represent the locking screws used in orthopedic plates. In these situations, the screw should display minimal movement inside its socket when loads are applied it.

The applied force on the top of the screw is chosen as a second indicator gauging the rigidity of the contact, while also providing information on the convergence of simulations.

Finally, the equivalent Von Mises stress inside the structure is also selected to gauge the convergence of the simulations.



**Mesh density** The mesh density is chosen using the length of elements  $l_e$  passed to *gmsh* during the meshing part of the simulation. The set of values used for this test battery is representative of the lengths of elements observed in the simulations with the model of the bone. Overall, an increase in the rigidity of the contact due to the refinement of the mesh is observed in all tested cases. This increase is however much less pronounced in cases where the spring stiffness is high (Figure II.30c, II.30d, II.31c and II.31d). Some stabilization is observed on the applied force when going above a certain level of refinement, which is quickly reached in the real geometries of plates and screws for simulations with a mesh density factor above 0.6 (where screws have  $\sim 1000$  nodes)

**Spring Stiffness** As the value for the spring stiffness is totally unknown using this new method of contact management, it is varied through 6 orders of magnitude to determine the proper range to select in further simulations.

The results obtained by varying this stiffness are as expected, with a larger spring stiffness bringing a more rigid contact. Looking at the displacement ratio, the penetration of the screw goes from almost 100% with  $K = 10^2$  [N/mm] to under 7% across parameters with  $K > 10^6$  [N/mm].

However a stagnation above values situated between  $10^4 - 10^6$  [N/mm] is observed and increasing the stiffness above  $10^6$  [N/mm] brings virtually no change in the results (see the similarity between II.30c and II.30d, II.31c and II.31d, and II.32c and II.32d)

Moreover, having extremely high values of the stiffness is detrimental to the progress of simulations, as they tend to introduce locally extreme fields even with small displacements, requiring the simulation to have tighter numerical tolerances to obtain valid results.

**Number of springs per node** The number of springs connecting each node influences the rigidity of the contact, virtually constraining the node between those to which it is attached. In the case of a perfect screw attached to a perfect socket (conforming meshes), meaning each node of the screw is at the same position as the corresponding node in the socket, only one connecting spring would be required. This single spring would get 'engaged' as soon as the screw moves in the socket. In the practical cases considered in this work, where the meshes are non-conforming, nodes are generally not superposed, and it is possible that nodes move without engaging the spring (hypothetical case where the node moves in a circle around the node to which it is attached). To ensure that contact is engaged immediately when screw nodes move relative to nodes in the socket, multiple springs are used.

Overall, responses differ quite significantly when going from  $n_S = 1$  to  $n_S = 3$ , and much less when going from  $n_S = 3$  to  $n_S = 5$ . Looking at the maximum equivalent Von Mises stresses (e.g. in Figure II.32c), the difference between  $n_S = 3$  and  $n_S = 5$  is insignificant, suggesting that increasing the number of springs above 3 is not required for proper contact engagement.

**Summary of parameter value selection** The mesh density analysis was done to show that the new contact solution is unfortunately still mesh-dependent, albeit less so when higher values of the spring stiffness are used. To balance the trade-off between the need of a rigid contact and reasonable (not over-tight) numerical tolerances, the spring stiffness will be chosen as  $K = 10^5$  in further simulations. The number of springs per node  $n_S$  will

be set at 3 to engage the contact interaction as soon as relative movement is detected.

## II.2.6 Simulation results with updated contact management

Using the newly implemented contact management and the chosen parameter values, results for bone compression simulations are displayed in Figure II.34.

Compared to results obtained using the over-constraining contact management method, applied forces are reduced in magnitude. The change is explained by the removal of the extremely rigid contact regions, allowing contacting objects to slightly move relatively to each other.

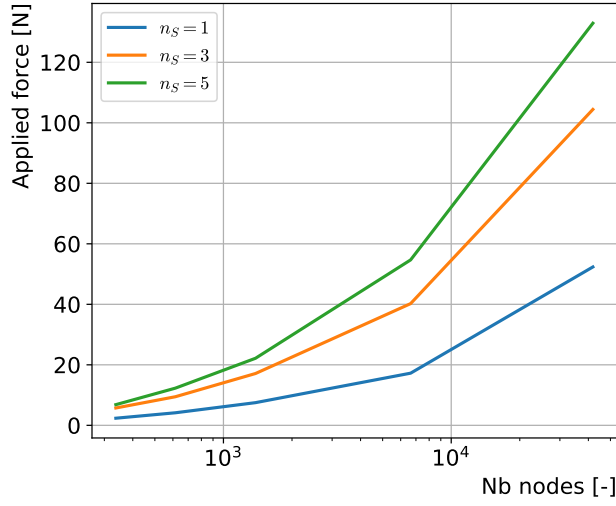
Stresses in the plate seem to reduce in magnitude, while stresses in the bone show higher magnitudes. The explanation behind this is likely that the contacts between the screws and the bone are no longer over-constrained regions with zero-stress/strains, and thus showing stress fields that are more akin to what is physically expected. The reduction in maximum EVMS in the plate is likely also due to the same reason, with the increased flexibility of the screw to plate contacts, the plate no longer has to sustain as high of an effort as before. The maximum EVMS in screws do not show major differences. However, their distributions change dramatically.

Using the new contact management method, the equivalent Von Mises stress field of screw 3 is displayed in Figure II.35. The stress field is now non-zero on the screw inside of the bone and the plate. Moreover, a slight concentration of the stress field can be seen in the cortical regions of the bone, where material properties are typically higher, indicating a more physically accurate transfer of efforts that was not present in the previously implemented models[10]. Regarding absolute values of stresses reached within the screw, a better distribution of stresses can be observed. The previously very concentrated shearing efforts seen between the plate and the bone are much less pronounced due to the part of the screw inside the bone and the plate being able to slightly deform.

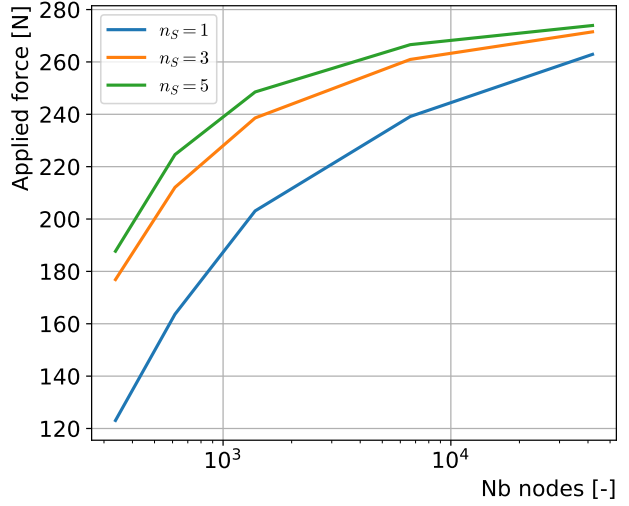
## II.3 Chapter summary

In the first part of this chapter, a complete and didactic image-to-mesh pipeline was presented. An effort was undertaken in order to make the meshing process more modular, easing the application of the pipeline for more generic meshes and the creation of various implant configurations. The pipeline showed significant improvements in the quality of generated meshes compared to its predecessor found in [10].

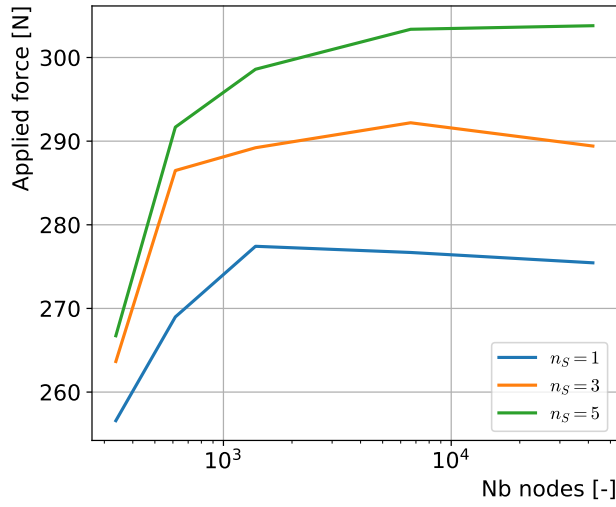
The second part of the chapter described the numerical simulations in *Metafor*. Simulations using the updated meshes were made to determine an appropriate mesh density factor. These simulations outlined issues with the previous contact management method. An improved version of contact management was proposed and a sensitivity study on the parameters involved in the new method was conducted. After selecting values for the parameters involved in the new contact management method, another series of simulations was performed and showed improved results.



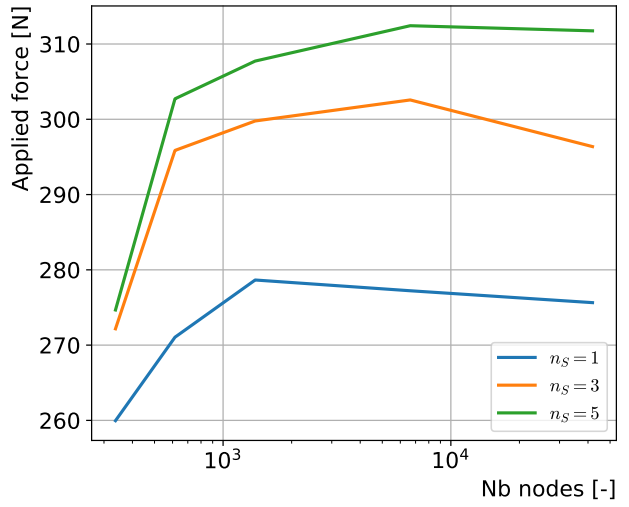
(a)  $K = 10^2$  [N/mm]



(b)  $K = 10^4$  [N/mm]

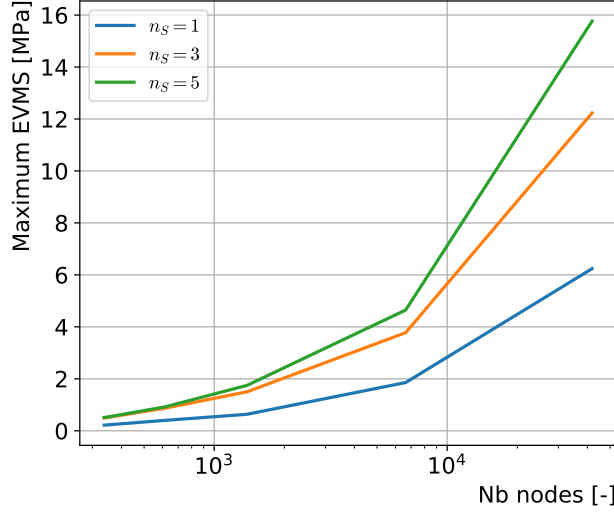


(c)  $K = 10^6$  [N/mm]

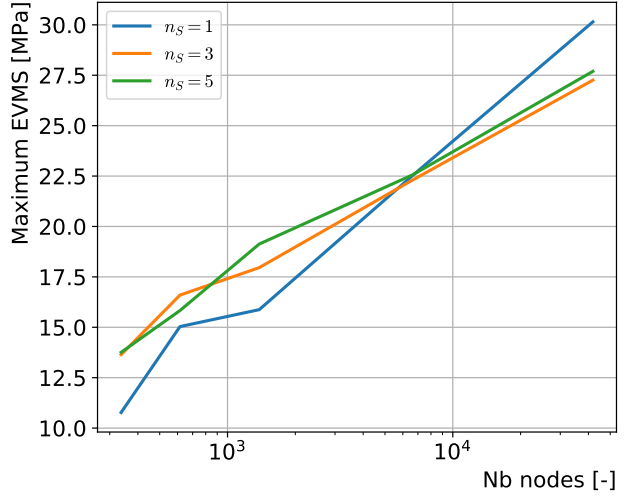


(d)  $K = 10^8$  [N/mm]

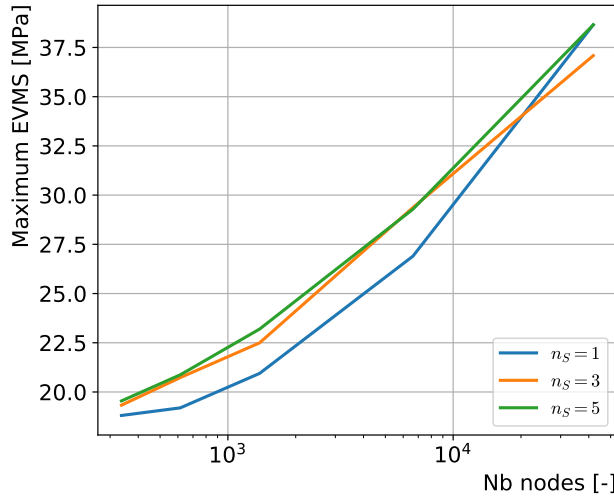
Figure II.31: Applied force on the top surface of the screw. Element edge length  $l_e$  in  $\{0.25, 0.50, 1.0, 1.5, 2.0\}$  [mm], spring stiffness  $K$  in  $\{10^2, 10^4, 10^6, 10^8\}$  [N/mm], number of springs per node  $n_S$  in  $\{1, 3, 5\}$  [-].



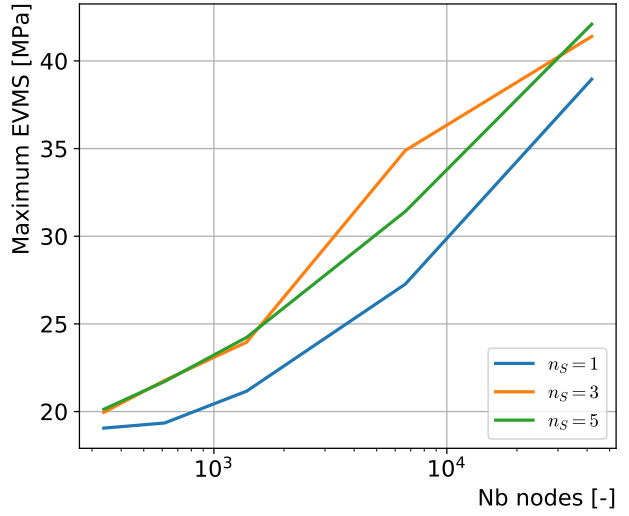
(a)  $K = 10^2$  [N/mm]



(b)  $K = 10^4$  [N/mm]

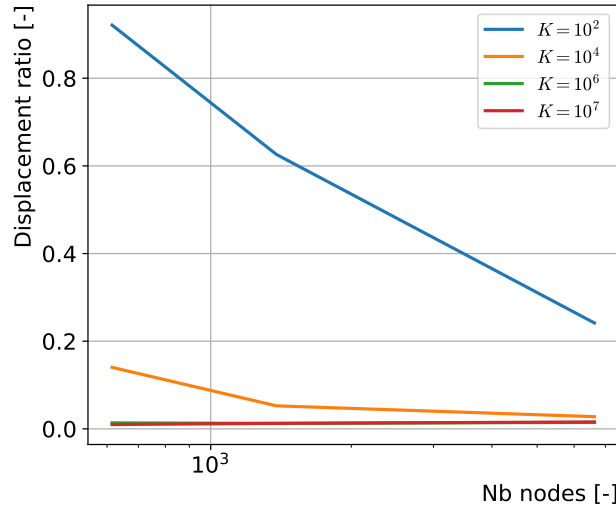


(c)  $K = 10^6$  [N/mm]

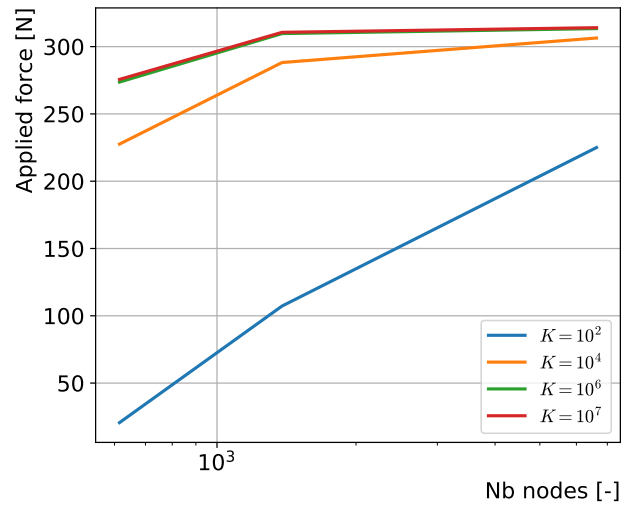


(d)  $K = 10^8$  [N/mm]

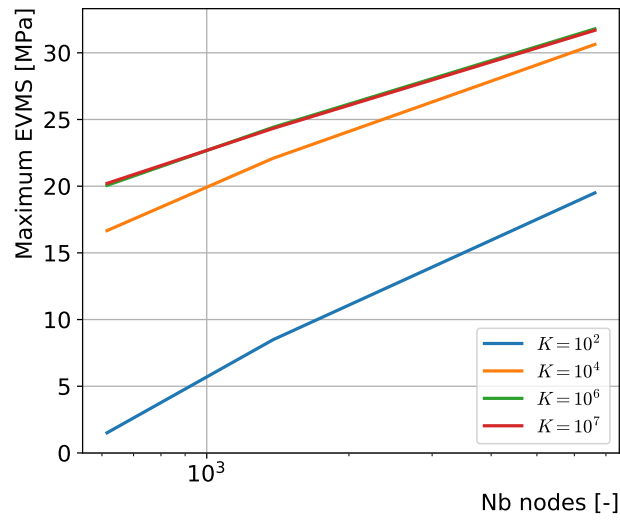
Figure II.32: Maximum equivalent Von Mises stress in the whole geometry. Element edge length  $l_e$  in  $\{0.25, 0.50, 1.0, 1.5, 2.0\}$  [mm], spring stiffness  $K$  in  $\{10^2, 10^4, 10^6, 10^8\}$  [N/mm], number of springs per node  $n_s$  in  $\{1, 3, 5\}$  [-].



(a) Displacement ratio



(b) Applied force



(c) Maximum EVMS

Figure II.33: Results using the reduced geometry and the over-constraining contact management method. Element edge length  $l_e$  in  $\{0.50, 1.0, 2.0\}$  [mm], spring stiffness  $K$  in  $\{10^2, 10^4, 10^6, 10^7\}$  [N/mm].

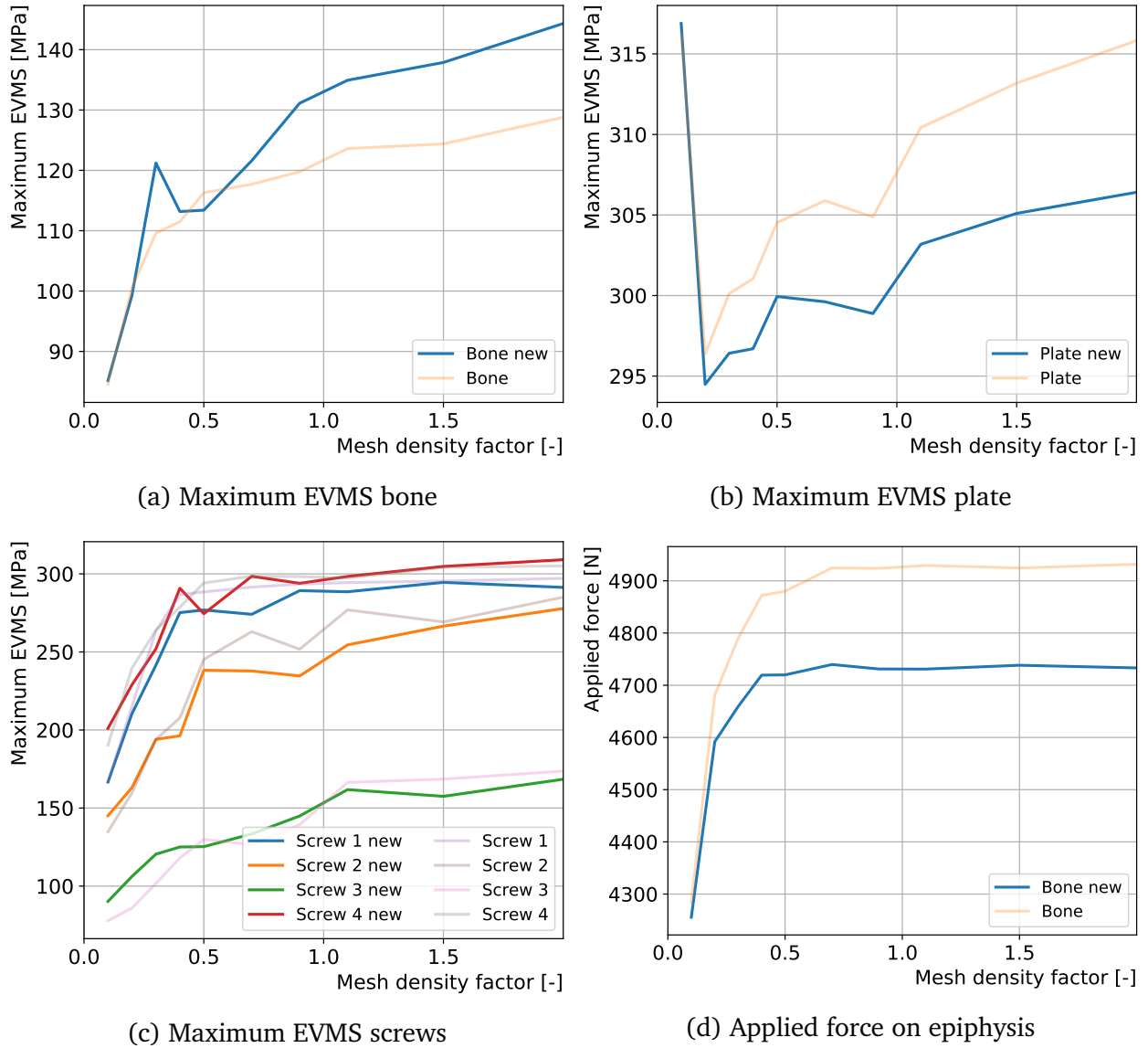


Figure II.34: Maximum equivalent Von Mises stress reached in each object and total applied force on the bone, varying the mesh density factor  $m$ . Contact management is done using the newly implemented springs generation method. Previous results using the over-constraining method are shown with transparency.

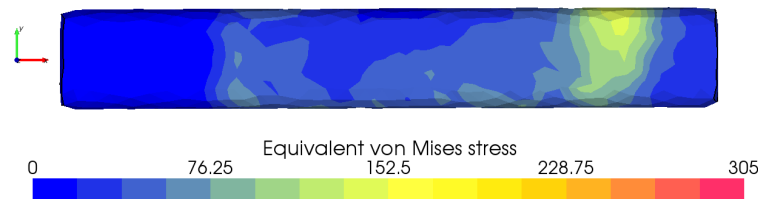


Figure II.35: Equivalent Von Mises stress field of screw 3, mesh density factor  $m = 0.7$ . Contact management is done using the newly implemented method.

### III Continuum Damage Mechanics as a tool to study bone fracture healing

Continuum Damage Mechanics (CDM) has been used previously in the context of modeling remodeling processes in orthodontic tissues [25][26], and more broadly in bone remodeling [27][28][29][30].

Damage mechanics initially studied material deterioration or failure when exposed to a large number of loading cycles or large loads. The challenge not being in the representation of the state of the material within a loading cycle, but in the long term behavior when material properties degrade due to the repeated load application. To describe this evolution in time, CDM generally requires three components[31]:

- A definition or a measure of damage, i.e. a damage variable, or multiple damage variables in the case of more exotic models, possibly describing the microstructure,
- A description of the mechanical behavior when accounting for the damage, i.e. a description of how material properties change with damage,
- A description of how the damage variable(s) evolve(s) with the loading, i.e. a set of equations governing the damage rate.

The motivation behind the use of CDM in this work is that multiple parallels can be drawn between the fields of damage mechanics and the phenomenon of bone remodeling. In both cases, the material is subjected to a large number of loading cycles over a long period of time and the mechanical behavior in one loading cycle is obtainable using available constitutive laws (elasto-plasticity for metals, orthotropic elasticity for bone). It is thus, in both cases, the deterioration (or improvement in the case of bone) of material behavior over the cycles that is of interest.

The difference between traditional materials and bone is that the first lose stiffness in overloaded cases and are unaffected by underloading, whereas bone is able to improve the material properties when subjected to overloading (negative damage evolution), and loses stiffness when it is underloaded (positive damage evolution). In some sense, this positive remodeling behavior can be seen as a material able to display inverted creep for an imposed displacement, or reversed relaxation for some imposed force.

Damage variables were first introduced for traditional materials as a representation of the effective surface density of microdamage per unit of volume.

**Example of isotropic damage** To introduce isotropic CDM damage variables, let us consider a uniformly damaged bar that is subjected to a uniaxial tensile stress  $\sigma$  with a cross-sectional area  $A$  (in the damaged configuration), the force acting on the bar is thus expressed with  $f = \sigma A$ . CDM proposes to introduce a fictitious undamaged configuration,

called effective configuration, where all the damage is removed. In this effective configuration, the cross-sectional area is now  $\tilde{A}$  and the effective uniaxial tensile stress is  $\tilde{\sigma}$ . Both bars (in the damaged and effective configurations) are subjected to the same tensile force  $f = \sigma A = \tilde{\sigma} \tilde{A}$ , therefore,

$$\tilde{\sigma} = \sigma \frac{A}{\tilde{A}} \quad (\text{III.1})$$

$$\Rightarrow \tilde{\sigma} = \frac{\sigma}{1 - \phi} \quad \text{where } \phi = \frac{A - \tilde{A}}{A}. \quad (\text{III.2})$$

The effective stress in [III.2](#) can be interpreted as an average stress acting on an effective (the actual area of the material that is capable of sustaining a load) area of the material. A virtual damage-free material has been introduced to represent the effective stress in a damaged material.

To generalize this illustrative case, isotropic damage variables are defined in a similar way using a direction-independent damage variable  $d$  such that

$$\tilde{\sigma} = \frac{\sigma}{1 - d}. \quad (\text{III.3})$$

The damage variable is now an internal variable representing the effective density of cracks (or cavities) at each point in space. This damage variable can be generalized to an anisotropic case by adding directional information to the damage variable (we would thus have a damage tensor). Continuum damage theory is a phenomenological approach to damage, as it does not use physical quantities related to the actual damage mechanism inside the material, whether it be crazing, cavitation, cracks, etc...

An important note here is that the damage variable is normalized,  $d = 0$  represents the undamaged virgin material, whereas  $d = 1$  represents the local absence of material, i.e. local failure.

To relate the constitutive laws in the effective configuration to the one in the damaged configuration, one can relate the stress level in the damaged material with the stress level in the undamaged material that leads to the same strain. This approach of describing how material properties change with damage is called *strain equivalence*. For small strain elasticity:

$$\tilde{\sigma} = \mathbb{H}^0 : \varepsilon \quad \text{in the effective configuration,} \quad (\text{III.4})$$

$$\sigma = \mathbb{H}_{se} : \varepsilon \quad \text{in the actual(damaged) configuration,} \quad (\text{III.5})$$

Using equations [III.4](#) and [III.5](#) in [III.3](#), one obtains  $\mathbb{H}_{se} = (1 - d_{se}) \mathbb{H}^0$ , which is Hooke's fourth order tensor for the damaged material.

Finally, the stiffness variation (i.e. the evolution of the mechanical behavior with damage) is

$$\dot{\mathbb{H}}_{se} = -\dot{d}_{se} \mathbb{H}^0. \quad (\text{III.6})$$

For an 1D problem, damage can be related to the elastic modulus with  $d_{se} = 1 - \frac{E}{E_0}$ .

Thermodynamic relations enable the writing of damage evolution laws. Using an additive decomposition of strains

$$\varepsilon = \varepsilon^e + \varepsilon^p \quad (\text{III.7})$$



and defining the following state variables: the strain field ( $\varepsilon$ ), the plastic strain ( $\varepsilon^p$ ), the damage ( $d$ ) and the variables associated with hardening ( $q$ ). The free energy is written  $\Psi(\varepsilon, \varepsilon^p, d, q)$  or  $\Psi(\varepsilon^e, d, q)$  by taking into account the additive strain decomposition. Taking the variation of the free energy:

$$\dot{\Psi} = \frac{\partial \Psi}{\partial \varepsilon^e} : \dot{\varepsilon}^e + \frac{\partial \Psi}{\partial d} \dot{d} + \frac{\partial \Psi}{\partial q} \dot{q}. \quad (\text{III.8})$$

The Clausius-Duhem inequality ( $\sigma : \dot{\varepsilon} - \dot{\Psi} \geq 0$ ) is then written

$$\left( \sigma - \frac{\partial \Psi}{\partial \varepsilon^e} \right) : \dot{\varepsilon}^e + \sigma : \dot{\varepsilon}^p - \frac{\partial \Psi}{\partial d} \dot{d} - \frac{\partial \Psi}{\partial q} \dot{q} \geq 0. \quad (\text{III.9})$$

Given that the inequality has to be verified  $\forall \dot{\varepsilon}^e$ , we thus have

$$\sigma = \frac{\partial \Psi}{\partial \varepsilon^e}, \quad (\text{III.10})$$

i.e. the Cauchy stress tensor is the variable thermodynamically conjugated to the elastic strain field.

The other derivatives define the energy conjugated variables related to damage and hardening, respectively,

$$Y = \frac{\partial \Psi}{\partial d}; \quad Q = \frac{\partial \Psi}{\partial q}. \quad (\text{III.11})$$

The Clausius-Duhem inequality becomes

$$\sigma : \dot{\varepsilon}^p - Y \dot{d} - Q \dot{q} \geq 0. \quad (\text{III.12})$$

A dissipation potential, i.e. a convex scalar function of the conjugated variables  $\varphi(\sigma, Y, Q)$  is then used to define evolution laws for the variables. The normality rule of generalized standard materials then allows us to write:

$$\dot{\varepsilon}^p = \lambda \frac{\partial \varphi}{\partial \sigma}; \quad \dot{d} = \lambda \frac{\partial \varphi}{\partial Y}; \quad \dot{q} = -\lambda \frac{\partial \varphi}{\partial Q} \quad (\text{III.13})$$

For simplicity, the dissipation potential is decomposed into a plastic dissipation potential related to the yield criterion  $f$ , and a damage dissipation potential  $\varphi_d$ . Additionally, considering the damaged equivalent plastic strain ( $q = \bar{\varepsilon}^{p,d}$ ) as the only internal variable associated to the yield limit of the material ( $Q = \sigma_y$ ). The normality rule finally becomes

$$\dot{\varepsilon}^p = \lambda \frac{\partial f}{\partial \sigma}; \quad \dot{d} = \lambda \frac{\partial \varphi_d}{\partial Y}; \quad \dot{q} = -\lambda \frac{\partial f}{\partial Q} = \lambda \quad (\text{III.14})$$

With these evolution laws, assuming the yield function and the damage dissipation potential can be defined, the 3 ingredients necessary in CDM models are present. Theoretically, one could obtain the long-term response of a material by integrating the evolution laws.

Extensions to anisotropic damage exist, the details may be found in more complete references such as [30].

In the following sections, an application of the principles of CDM to bone remodelling will be presented.

### III.1 Stanford Model for isotropic bone remodeling

Before presenting a remodeling model based on CDM theory in the next section, a remodeling model is first presented as an introduction to the mathematical and physical representation of bone remodeling.

As in damage mechanics models, two main approaches exist for bone remodeling models. The first being phenomenological models, where the detailed biological activity within the bone is not considered and whose goal is to predict global mechanical behavior of the bone when subjected to loads, while taking into account the microstructure through internal variables of the model. The second being mechanobiological models, whose goal is to determine the evolution of the microstructure and biological parameters when the bone is subjected to mechanical loading. In this work, mechanobiological models are not considered. Both presented models for bone remodeling follow the phenomenological approach.

Phenomenological bone remodeling models use a notion of *mechanical stimulus* that either causes bone apposition or resorption depending on whether this input stimulus is higher or lower than a tissue homeostatic stimulus level, respectively. In a sense, these apposition and resorption phenomena follow the rules advanced in Frost's mechanostat theory. The mechanostat rules state, broadly, that overloaded parts of the bone tend to remodel to better sustain the increased loading, whereas underloaded parts of the bone tend to resorb, as their role in sustaining the loading is not critical. These structural changes in the bone lead to a material that ideally resists usual loading with the most economical amount of material. Complementing these functional adaptation rules, a lazy zone in which the bone is considered neither overloaded nor underloaded, i.e. in which the bone is adapted to the local loading, is usually considered in phenomenological models. The existence of a lazy zone explains why bone material in a localized region submitted to the same loading may display slight variations in material properties. The definition of the stimulus itself can use a wide range of mechanical quantities such as stresses, strains, strain energy density, strain rate or damage.

In these phenomenological models, the remodeling is usually implemented through constitutive material models (stress/strain relationships) that take into account internal variables that represent the evolving local bone structure. The most common choice is having a Young's modulus that is dependent on the bone density, for which an evolution law is written.

In other words, remodeling occurs in the bone tissue to bring a tissue-level (subscript  $t$ ) stimulus  $\psi_t$  close to a homeostatic tissue level value,  $\psi_t^*$ . This stimulus is a scalar that is representative of the locally applied mechanical loading.

A *remodeling rate*  $\dot{r}$  is then defined as a function of the difference between the current ( $\psi_t$ ) and homeostatic ( $\psi_t^*$ ) stimulus. In the models presented in this work, a lazy zone is assumed to exist. This lazy zone is defined as a interval of tissue stimulus around the homeostatic stimulus for which no remodeling occurs. Usually, this interval is centered on the homeostatic stimulus level and has a half-width  $\omega$ . The remodeling rate needs to take into account this lazy zone, the remodeling rate's effect must be that resorption of the bone takes place in underloaded bone tissue (i.e. where  $\psi_t < \psi_t^* - \omega$ ); on the other side of the stimulus interval, apposition of bone tissue takes place in overloaded bone tissue (i.e. where  $\psi_t^* + \omega < \psi_t$ ). These last two inequalities are referred to as *remodeling criteria*.

For the models presented in this work, it is assumed that only the density is affected by remodeling and mechanical properties are affected by remodeling only if they explicitly

depend on bone density. Another function linking the remodeling rate to the *density rate* in the tissue is thus needed, enabling the mechanical properties of the tissue to evolve in time. Note that the density variation only has an effect on the material properties, it does not actually produce new mass in the model that would require updates to the mass matrix in FE computations.

In short, phenomenological models need to define three main functions:

- A stimulus, defined using mechanical quantities representative of the local tissue loading, which is responsible for causing remodeling,
- A remodeling rate function that is responsible for determining how much the tissue is overloaded or underloaded (usually a function of the remodeling criteria),
- A function relating the density rate to the remodeling rate to determine how much the density changes for a certain remodeling rate, ultimately allowing to compute mechanical properties changes.

The first complete phenomenological remodeling model developed in [32] [33], also called the Stanford model, is presented here as an introduction to the subject. This model is written for small strains isotropic linear elasticity, where materials follow Hooke's law:

$$\boldsymbol{\sigma} = \mathbb{H} : \boldsymbol{\varepsilon} \quad (\text{III.15})$$

In the Stanford model, the stimulus at a continuum level is written

$$\psi_t = N^{1/m} \left( \frac{\rho_0}{\rho} \right)^2 \sqrt{E \boldsymbol{\sigma} : \boldsymbol{\varepsilon}} \quad (\text{III.16})$$

where  $N^{1/m}$  is a proportionality coefficient accounting for the repeated application of the loads leading to the observed stresses and strains in which  $m = 4$  and  $N$  is the number of cycles per time frame considered,  $\rho_0$  is the density of fully calcified bone,  $\rho$  is the apparent density,  $E$  is the density dependent Young's modulus,  $\boldsymbol{\sigma}$  is the Cauchy stress tensor and  $\boldsymbol{\varepsilon}$  is the Cauchy strain tensor.

The mechanical parameters depend on the density following:

$$E = B(\rho) \rho^{\beta(\rho)} \quad (\text{III.17})$$

$$\nu = \nu(\rho) \quad (\text{III.18})$$

In this work, a constant Poisson's ratio is selected and the Young's modulus follows

$$E = \begin{cases} 2014 \rho^{2.5} [\text{MPa}] & \text{if } \rho \leq 1.2 \text{g/cc} \\ 1763 \rho^{3.2} [\text{MPa}] & \text{if } \rho \geq 1.2 \text{g/cc} \end{cases} \quad (\text{III.19})$$

Injecting equation III.17 in the stimulus definition yields

$$\psi_t = \underbrace{N^{1/m} \rho_0^2 \sqrt{B}}_C \frac{\sqrt{\boldsymbol{\sigma} : \boldsymbol{\varepsilon}}}{\rho^{2-\beta/2}} \quad (\text{III.20})$$

The remodeling rate is written in terms of the remodeling criteria,  $g_f$  and  $g_r$ , the out-of-bounds amplitudes of the stimulus, measuring the signed distance between the value of the stimulus and the borders of the lazy zone.

$$g_f = \psi_t - (\psi_t^* + \omega) \quad \text{for formation conditions} \quad (\text{III.21})$$

$$g_r = \psi_t - (\psi_t^* - \omega) \quad \text{for resorption conditions} \quad (\text{III.22})$$

	$g_f$	$g_r$	$\dot{r}$	remodeling	density
$\psi_t < \psi_t^* - \omega$	$<0$	$<0$	$<0$	resorption	decreases
$\psi_t^* + \omega < \psi_t$	$>0$	$<0$	$>0$	formation	increases
$\psi_t^* - \omega < \psi_t < \psi_t^* + \omega$	/	/	0	lazy zone	stays the same

Table III.1: Summary of density evolution cases.

The remodeling rate is then simply written as a piecewise linear function of these remodeling criteria

$$\dot{r} = \begin{cases} c_f g_f & \text{for } g_f > 0, \\ 0 & \text{for } \psi_t^* - \omega \leq \psi_t \leq \psi_t^* + \omega, \\ c_r g_r & \text{for } g_r < 0 \end{cases} \quad (\text{III.23})$$

where  $c_f$  and  $c_r$  are remodeling constants, affecting the speed of remodeling, usually expressed in  $[\mu\text{m}/(\text{day MPa})]$ . They are initially deduced from mechanobiological studies looking at the evolution of trabecular thickness on a microscopic scale. The general aspect of the remodeling rate is depicted in Figure III.1.

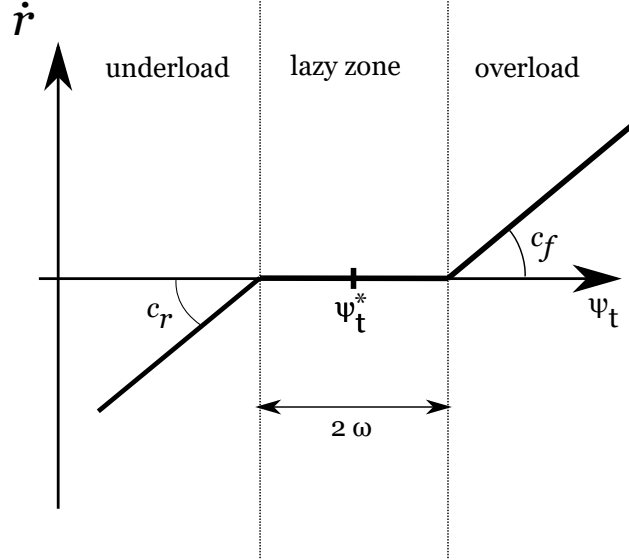


Figure III.1: Aspect of the remodeling rate in the Stanford model.

Finally, the density rate is written

$$\dot{\rho} = k S_v \rho_0 \dot{r} \quad (\text{III.24})$$

where  $k S_v$ ,  $k \in [0, 1]$  indicates the available surface area for bone apposition/resorption to occur. Their presence stems from the necessity for a bone surface to exist inside the bone tissue for osteoblasts and osteoclasts to deposit and remove bone matter, respectively.  $S_v$  is a function of the porosity  $f = 1 - \frac{\rho}{\rho_0}$ , the curves looks like a bell with the bell tips at  $f = 0$  and  $f = 1$ . This available surface is, expectedly, null for  $f = 0$  (fully compact bone) and  $f = 1$  (absence of bone).

In summary the density evolves following the cases in Table III.1.

Using the previous equations for the stimulus, the remodeling rate, and the density rate allows to compute an update on the density according to the stimulus at each time step.

The mechanical parameters are also updated using the current density after each time step of a simulation. This completes the Stanford model for bone remodeling.

## III.2 CDM for isotropic bone remodeling

In the following, an application of CDM to the modeling of bone remodeling is presented. The constitutive model recalled in this section can be found in more detail in the works of M.Mengoni [30]. This model is an updated version of a model developed by M.Doblaré and J-M.Garcia [28][27], improved by M.Mengoni in [29], suitable for modeling anisotropic remodeling in a finite strains framework, here particularized to the isotropic case.

Being a CDM model, it needs to define the three required components of a damage model, briefly reminded here:

- A damage variable definition,
- A description of the material property change with damage. In this case, it is integrated to the material property changes in the remodeling part of the model,
- And an evolution law for the damage variable, that is assimilated by the density rate function in the remodeling part of the model.

Additionally, being a phenomenological remodeling model, it also needs to define the three aforementioned characteristic functions:

- A stimulus definition,
- A remodeling rate function,
- And a density rate function.

On the CDM side, the damage variable representative of the local bone morphology that is chosen in [30] is defined in the following way:

$$\mathbf{d} = \mathbf{I} - \bar{\rho}^\beta A \mathbf{T} \quad (\text{III.25})$$

where  $\bar{\rho} = \frac{\rho}{\rho_0}$  is the bone volume fraction, commonly denoted BV/TV in the literature,  $A$  is a parameter stemming from the specialization from the general anisotropic case to the isotropic case:

$$A = \frac{B(\rho)}{B(\rho_0)} \rho_0^{\beta(\rho) - \beta(\rho_0)}, \quad (\text{III.26})$$

in this work, using equation III.17, it results that  $A = 1$  for  $\rho > 1.2$  [g/cc] and  $A = 0.679$  for  $\rho < 1.2$  [g/cc].

$\mathbf{T}$  is a fabric tensor, describing the local anisotropy in the arrangement of the bone density (local bone morphology), this tensor is normalized such that  $\text{tr}(\mathbf{T}) = 3$ .

This definition of the damage fulfills the requirements to be a damage variable:  $\mathbf{d} = \mathbf{0}$  for  $\rho = \rho_0$  and  $\mathbf{T} = \mathbf{I}$ , corresponding to fully calcified bone with isotropic properties, i.e. an undamaged state and  $\mathbf{d} = \mathbf{I}$  for  $\rho = 0 \ \forall \ \mathbf{T}$ , corresponding to an empty bone.

The damage is sometimes written

$$\mathbf{d} = \mathbf{I} - \mathbf{H}^{-2} \quad \text{where } \mathbf{H}^{-2} = \bar{\rho}^\beta \mathbf{A} \mathbf{T} \quad (\text{III.27})$$

where  $\mathbf{H}$ , the effective damage tensor, is a second order symmetric tensor whose principal directions are aligned with the fabric tensor  $\mathbf{T}$ . It describes not only the anisotropy of the bone microstructure through the fabric tensor  $\mathbf{T}$  but also the porosity of the bone tissue through the bone volume fraction  $\bar{\rho}$ . Damage evolution laws can be written equivalently in terms of the damage tensor  $\mathbf{d}$  or the effective damage tensor  $\mathbf{H}$ .

In this work, only isotropic damage is considered, the damage variable thus becomes  $d = d\mathbf{I}$  and the effective damage tensor  $\mathbf{H} = h\mathbf{I} = (1 - d)^{-1/2}\mathbf{I}$ .

The description of material property change with damage and the damage evolution law are described in the remodeling part of the model.

Regarding the phenomenological remodeling model, the stimulus is defined using the same equation III.20, reminded here:

$$\psi_t = \underbrace{N^{1/m} \rho_0^2 \sqrt{B}}_C \frac{\sqrt{\sigma : \varepsilon}}{\rho^{2-\beta/2}} \quad (\text{III.28})$$

The remodeling criteria take the form:

$$\text{For formation conditions : } g_f = C (1 - d)^{1/4} \sqrt{\frac{1}{2G} \mathbf{s} : \mathbf{s} + \frac{1}{K} p^2} - (1 + \Omega) \psi_t^* \rho^{2-\beta/2} \quad (\text{III.29})$$

$$= \underbrace{\frac{C (1 - d)^{1/4} \sqrt{\frac{1}{2G} \mathbf{s} : \mathbf{s} + \frac{1}{K} p^2}}{\rho^{2-\beta/2}}}_{\text{Stimulus } U_{\text{iso}}} - (1 + \Omega) \psi_t^* \quad (\text{III.30})$$

$$\text{For resorption conditions : } g_r = -U_{\text{iso}} + (1 - \Omega) \psi_t^* \quad (\text{III.31})$$

where  $\mathbf{s} = \text{dev}\sigma$ ,  $p = 1/3 \text{tr}\sigma$ ,  $K$  is the bulk modulus,  $G$  is the shear modulus,  $\Omega$  is the parameter setting the width of the lazy zone as a fraction of the homeostatic stimulus  $\psi_t^*$ .

The remodeling rate  $\dot{r}$  is then obtained from the remodeling criteria:

$$\dot{r} = \begin{cases} c_f g_f & \text{for } g_f \geq 0, \quad g_r < 0 \\ 0 & \text{for } g_f < 0, \quad g_r < 0 \\ -c_r g_r & \text{for } g_r \geq 0, \quad g_f < 0 \end{cases} \quad (\text{III.32})$$

The remodeling rate is schematically depicted in Figure III.2.

Using this remodeling rate function, the isotropic damage rate (evolution law for the damage variable) is then written:

$$\dot{d} = -\beta k S_v \dot{r} \frac{\rho_0}{\rho} (1 - d). \quad (\text{III.33})$$

Finally, the density rate of the Stanford model is reused here to complete the bone remodeling model:

$$\dot{\rho} = k S_v \rho_0 \dot{r} \quad (\text{III.34})$$



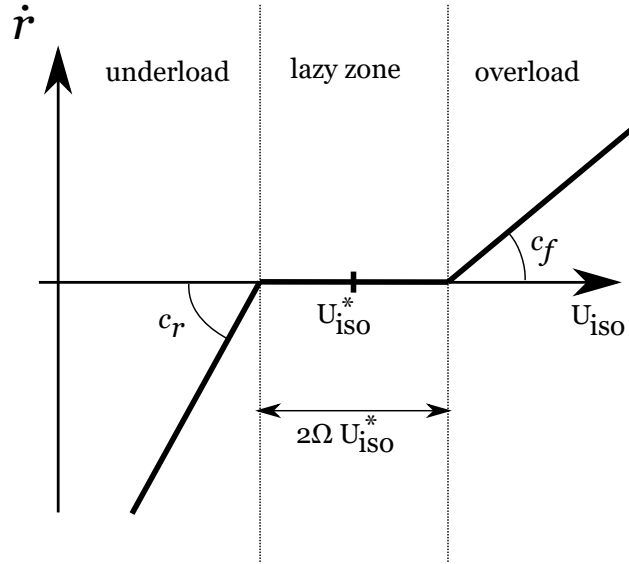


Figure III.2: Aspect of the remodeling rate in CDM based remodeling model.

Regarding the material property change with damage, it is computed by updating the material parameters in III.17 using the newly computed density.

This completes the list of all components for both a CDM model and a remodeling model, valid for isotropic remodeling in a finite-strains framework, this model supports an underlying material that is able to display hardening phenomena (the Stanford model was used only for linear elasticity in small strains).

Like for the Stanford model, time-integrating this model in a FE code gives the evolution of damage (and thus the density) at every point in space. This time integration is implemented in *Metafor*, the material law can be chosen under *ContinuousDamageEvplso-HHypoMaterial*  $\rightarrow$  *BoneRemodContinuousDamage*. The details of how the time integration is performed is out of the scope of this work, they can be found in [34] and [30] for the interested reader.

As previously mentioned, the isotropic remodeling model described here is a specialization to the isotropic case of a more general anisotropic remodeling model developed in [29] and [30]. In that reference, a thorough presentation of the anisotropic case is done and the model recalled here comes naturally when taking an isotropic damage variable. While the full understanding of the anisotropic model is suited for the interested reader, such a level of commitment is hopefully not necessary to understand the exploratory developments of the following sections.

### III.3 Sensitivity study of remodeling parameters

Before implementing the material law into a full simulation with a complex geometry, this section presents results obtained by using the material law on an elementary geometry. The developments in this section serve both as an illustration for the CDM-based remodeling model, allowing the reader to gain more insight into the behavior of remodeling and damage models, as well as a way to determine suitable parameter values for later simulations.

The tested geometry in this section is a cube composed of perfect hexaedral elements. the cube is compressed along one of its directions with a constant force applied normal to

a face of the cube with a magnitude of 5 [N]. The load is applied in a tenth of a day and then held constant for a year.

The simulated material is representative of bone material. The initial porosity of the damaged material is set at 0.6, corresponding to an initial damage in the material of  $d = 0.926$  and thus an initial density of  $\rho = 0.76$  [g/cc]. The initial Young's modulus at this density is  $E = 1014$  [MPa], corresponding to a very weak bone tissue.

The remodeling parameters of the base simulation are displayed in Table III.2, each of these parameters will be varied to explore the effect they have on the remodeling behavior of the material.

Homeostatic value of the stimulus :	$\phi_t^* = 10$ [MPa]
Remodeling velocity for resorption and formation :	$c_r = c_f = 0.02$ [ $\mu\text{m}/(\text{day MPa})$ ]
Surface area available for remodeling :	$k = 0.6$
Half-width of the lazy zone :	$\Omega = 0\%$ [-]
Number of daily load cycles :	$N = 1$

Table III.2: Base remodeling parameters used in this section

The indicators used to gauge the progress of remodeling in the material are the displacement of the plane on which the force applied, the magnitude of damage, the apparent density and the stimulus in the material. The displacement of the plane gives an indication of the current stiffness of the material, with an increased displacement indicating a weakening of the material (and thus an increase of the damage). The magnitude of the damage gives insight about how the damage law is evolving. The apparent density gives an indication about the remodeling process inside the bone, it is naturally linked to the amount of damage in the tissue. Finally, the value of the current stimulus in the material is displayed, this stimulus should in theory converge towards the homeostatic value of stimulus with time, its difference with the homeostatic value is indicative of the remodeling rate inside the material.

The results for a simulation using the base parameters can be found in Figure III.3. The plane displacement (Figure III.3a) decreases over time after the initial application of the load, indicating a strengthening of the material. That observation is corroborated by the curves of the damage (Figure III.3b) and the apparent density (Figure III.3c), both showing a strengthening behavior (i.e a decrease for the damage and an increase for the apparent density).

The stimulus in the tissue, motor of the damage variation (and in extension of the remodeling), is initially higher than the homeostatic tissue stimulus. The difference being positive, only the formation remodeling criteria is positive, and thus the remodeling rate is positive. Following the positive remodeling rate, the positive density rate then explains the increase in apparent density, ultimately yielding to the observed strengthening of the material. One may also expect that over time, a stabilization of the properties may be observed. Indeed, as the remodeling behavior here tends to increase the stiffness of the material, it in turns decreases the difference between the stimulus and the homeostatic value, thus decreasing the remodeling rate and slowing down the changes in material properties. This feedback loop continues until the stimulus reaches the edge of the lazy zone, at which point remodeling will stop.

In the following, each parameter in Table III.2 is varied through an appropriate range of values. Then, the impact that each parameter has on the remodeling behavior is described.

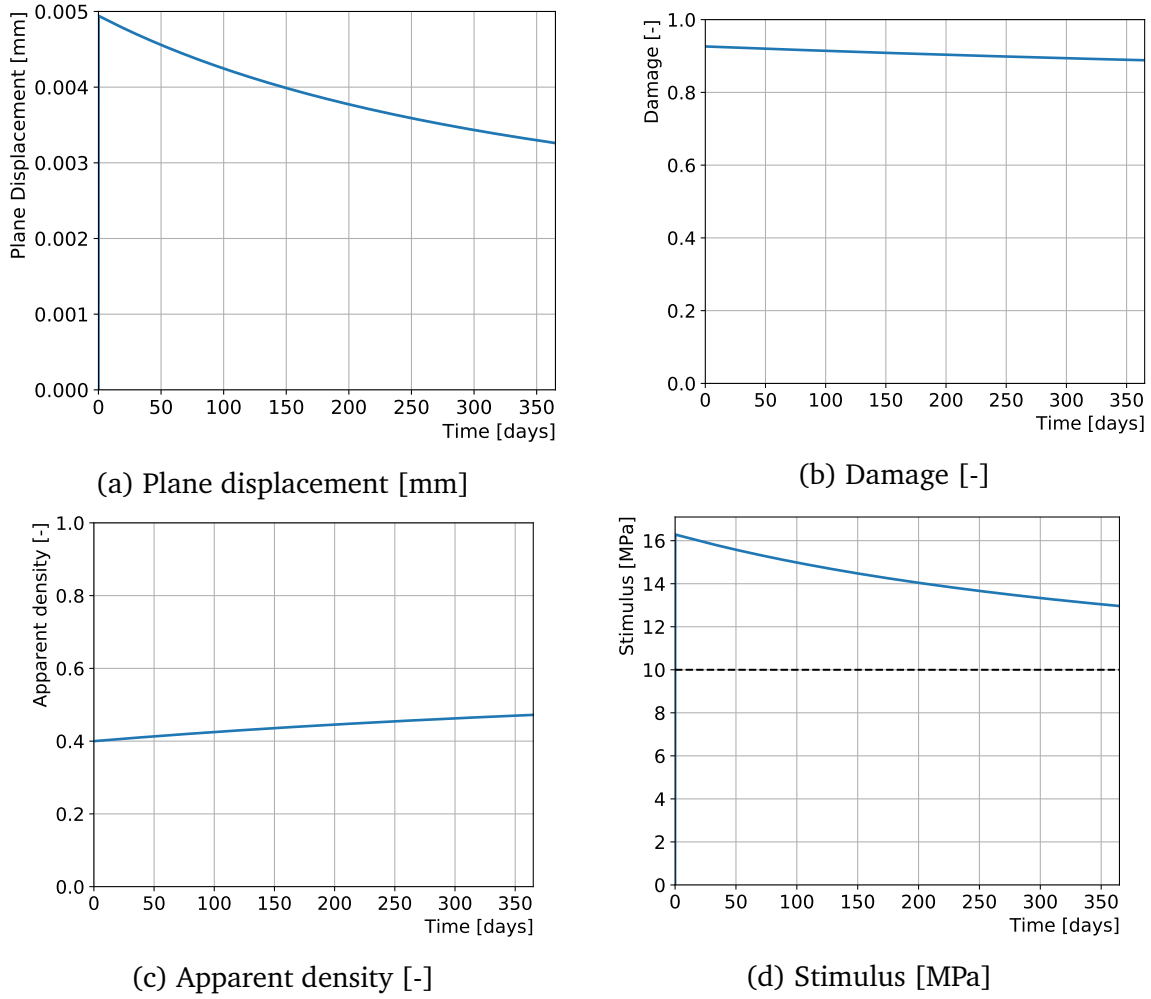


Figure III.3: Results of remodeling simulation using base parameters in Table III.2

### III.3.1 Homeostatic stimulus value $\psi_t^*$

The homeostatic value of the stimulus  $\psi_t^*$  is varied to take values in  $\{1, 5, 15, 30, 50\}$  [MPa], results are shown in Figure III.4. The higher values should raise the homeostatic stimulus above the initial stimulus in the material, thus inducing a reversal in behavior from the previous simulation. Using these higher values, the material is now in an "underloaded" state, the remodeling rate is now negative (the resorption criterion is enabled) and the remodeling now has the effect of decreasing the material stiffness. This decrease is evidenced in the increased plane displacement over time (Figure III.4a).

Like previously alluded to, the stimulus (Figure III.4d) tends to converge towards the homeostatic value, gradually decreasing the magnitude of the remodeling criteria, thus decreasing the magnitude of the remodeling rate, slowing down the damage evolution over time. This reduction in the remodeling force also explains the convergence of the damage and the apparent density for those higher values of the homeostatic stimulus.

### III.3.2 Remodeling velocities $c_r$ and $c_f$

The remodeling velocities that are the interface between the continuum modeling provided by CDM and the microscopic phenomena of bone apposition and bone resorption at the

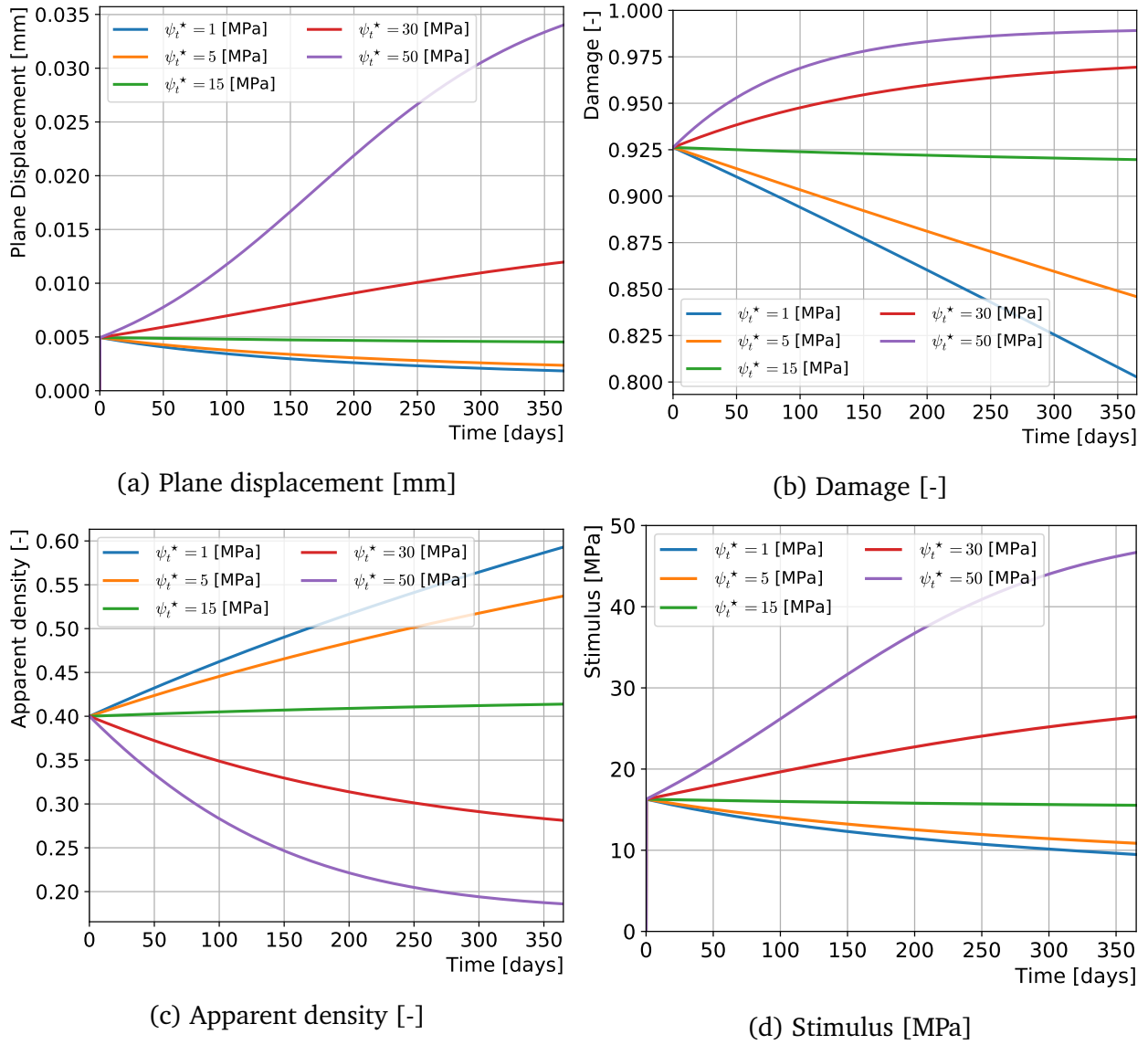


Figure III.4: Results of remodeling simulations. Parameters as in Table III.2 except for the homeostatic stimulus  $\psi_t^*$  taking values in  $\{1, 5, 15, 30, 50\}$  [MPa].

trabecular level. They are varied to take values in  $\{0.01, 0.02, 0.05, 0.10, 0.20\}$  [ $\mu\text{m}/(\text{day MPa})$ ]. The results are shown in Figure III.5. The effect that the remodeling velocity has on the remodeling behavior is, as indicated by their name, a change in the speed at which remodeling takes place. A higher value increases the magnitude of the remodeling rate in equation III.32, leading to a larger density rate, and thus a quicker remodeling phenomenon.

### III.3.3 Surface area available for remodeling $k$

The available surface area is varied to take values in  $\{20, 40, 60, 80, 100\}\%$ . This parameter represents the available surface at the trabecular level where bone cells are capable of depositing or removing bone. The last value of 100% is thus theoretical, as at any point, biologically speaking, not all the surface inside the bone experiences remodeling. Setting this value to 0 would inhibit the remodeling process (the material thus becoming a simple

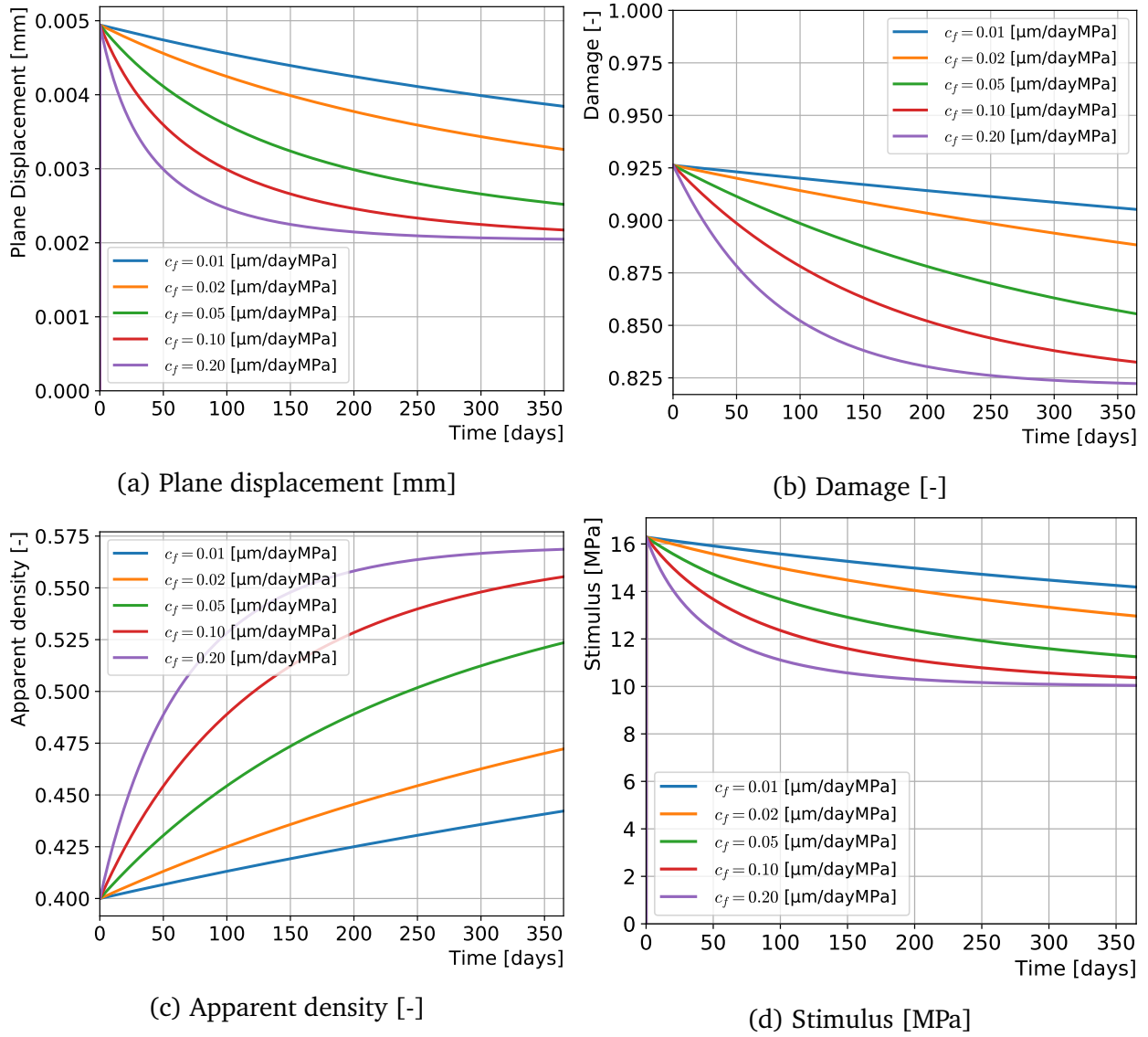


Figure III.5: Results of remodeling simulations. Parameters as in Table III.2 except for the remodeling velocities  $c_r$  and  $c_f$  taking values in  $\{0.01, 0.02, 0.05, 0.10, 0.20\}$  [ $\mu\text{m}/(\text{dayMPa})$ ].

linear isotropic hardening material). The effect of this variable on the remodeling behavior is similar to the one of the remodeling velocities, increasing or decreasing the magnitude of the density rate in III.24 for any remodeling rate. This effect is observable in Figures III.6a, III.6b and III.6c, where higher percentages of available area lead to faster evolution of the remodeling phenomenon.

### III.3.4 Half-width of the lazy zone $\Omega$

For this parameter, the homeostatic stimulus is set at  $\psi_t^* = 100$  [MPa] for clarity of interpretation of results.  $\Omega$  is varied in  $\{1, 5, 10, 20, 30\}\%$  of the homeostatic stimulus value. This parameter has a multiple effects on the remodeling phenomenon. By widening the width of the lazy zone, in addition to making the remodeling effect stop at an earlier stage (i.e. if formation occurs, the strengthening is less pronounced, the same in resorption conditions),

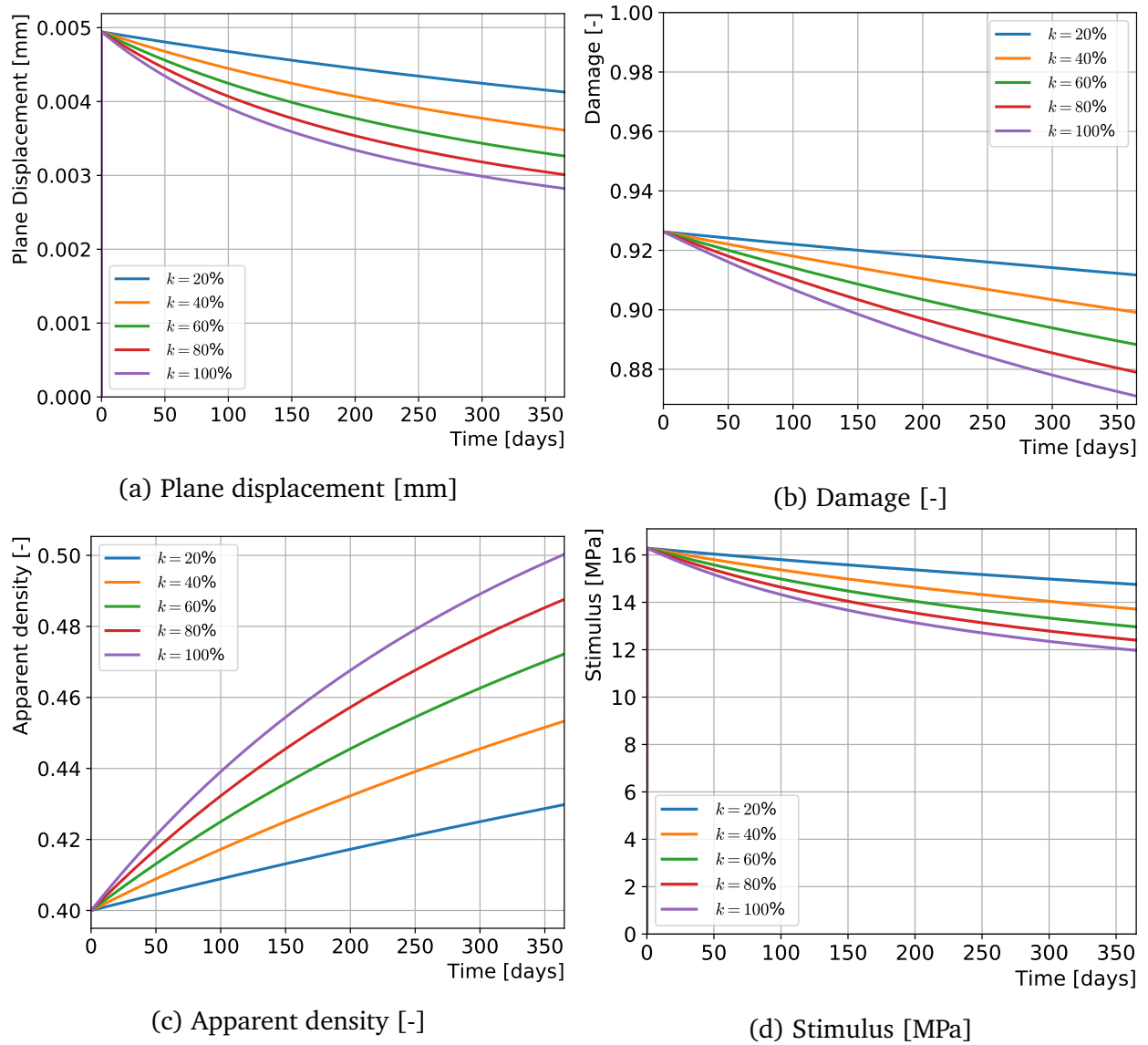


Figure III.6: Results of remodeling simulations. Parameters as in Table III.2 except for the available surface area for remodeling  $k$  taking values in  $\{20, 40, 60, 80, 100\}\%$ .

the stimulus also converges more slowly (see Figures III.7a, III.7b and III.7c). This effect is explained when taking a closer look at equations III.30, III.31 and III.32. The increase in the width of the lazy zone also decreases the magnitude of the remodeling criteria, thus impacting the magnitude of the remodeling rate.

### III.3.5 Number of daily load cycles $N$

When varying the number of daily cycles  $N$ , what should be understood is that for the remodeling laws in the background, it is as if the load (5 [N]) was applied  $N$  times per day. From what has been previously understood, increasing the number of daily cycles should yield a faster remodeling phenomenon, whether it is bone formation or resorption. It is indeed what is observed in Figures III.8a, III.8b and III.8c.



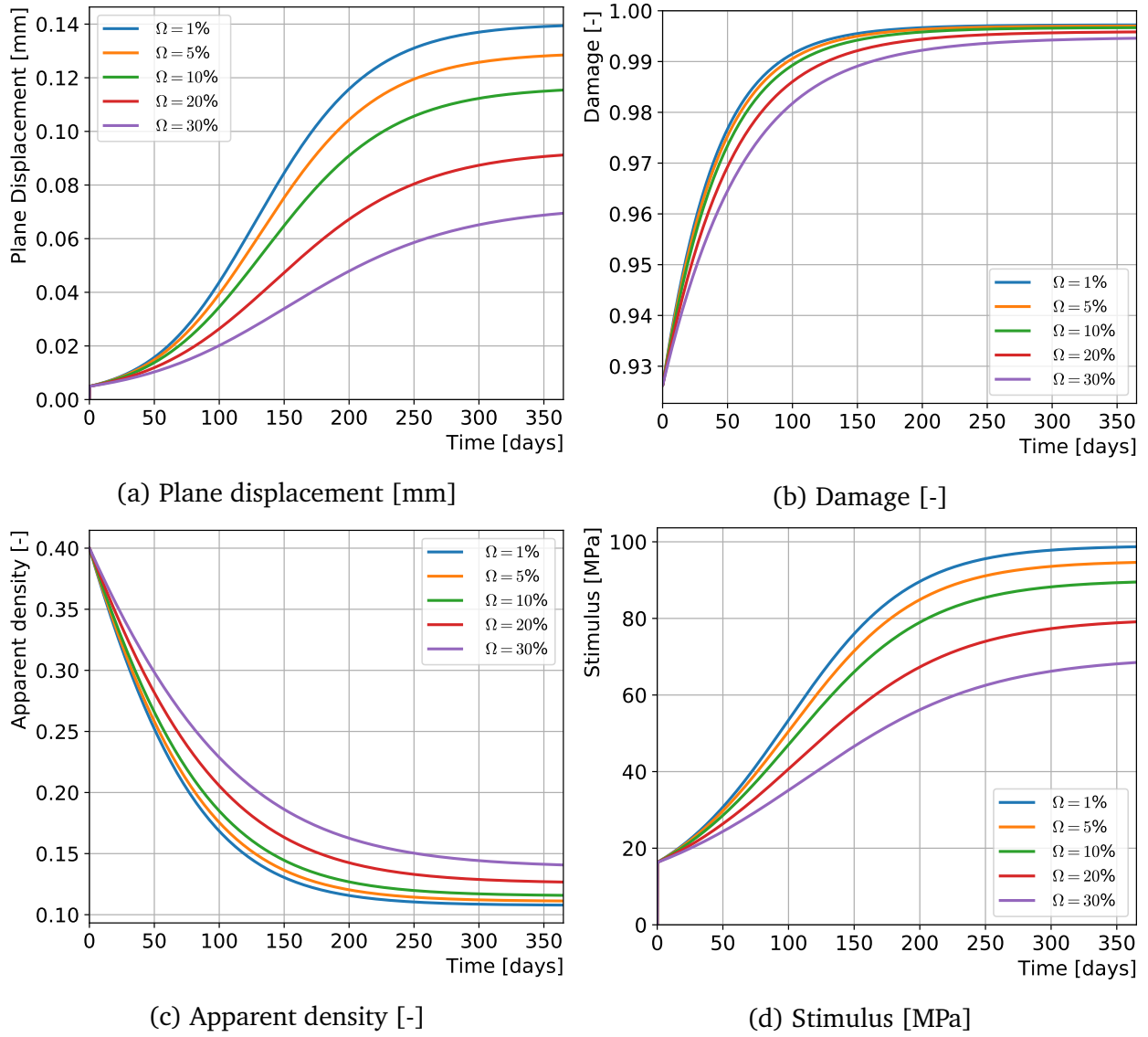


Figure III.7: Results of remodeling simulations. Parameters as in Table III.2 except for the homeostatic stimulus  $\psi_t^* = 100$  [MPa] and the half-width of the lazy zone  $\Omega$  taking values in  $\{1, 5, 10, 20, 30\}\%$ .

### III.3.6 Summary of parameter's effects on remodeling behavior

In summary, the main parameter conditioning the remodeling behavior is the homeostatic tissue stimulus  $\psi_t^*$ , whose value, relative to the stimulus generated by the loading in the material, is the condition that determines if new bone is formed or if bone is resorbed. All other parameters in this sensitivity study, i.e. the remodeling velocities  $c_f$  and  $c_r$ , the available surface area for remodeling  $k$  and the number of daily load cycles  $N$ , all have the effect of increasing the speed at which the currently active remodeling (formation or resorption) occurs. The width of the lazy zone controlled by  $\Omega$  has slightly more subtle effects, in the sense that both the remodeling speed as well as the final state of the remodeled material are impacted.

While the numerical study done here gives insight about the modeling of bone remodeling, it is important to note that the parameters involved in this study are for the most

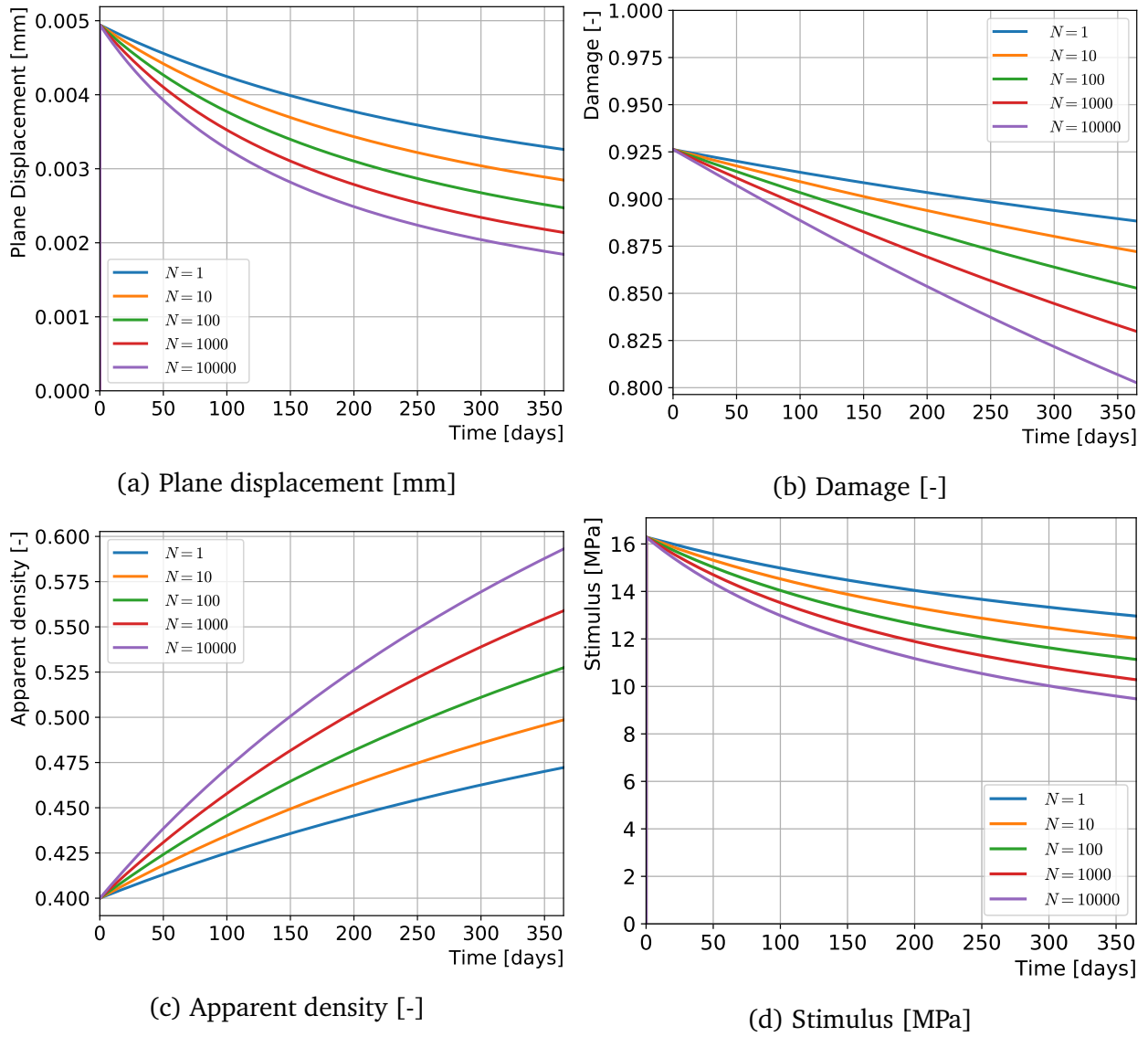


Figure III.8: Results of remodeling simulations. Parameters as in Table III.2 except for the number of load cycles  $N$  taking values in  $\{1, 10, 100, 1000, 10000\}\%$ .

part not directly observable in real experiments. Concepts such as a homeostatic stimulus value and lazy zone width are introduced to ease the mathematical modeling of biological phenomena but have no directly comparable quantities in terms of biological variables. Other parameters such as the remodeling velocities may have biological counterparts that are observable in experiments, but remain cryptic in terms of the exact value or range of values that they take. Indeed, these biological quantities may display large inter-individual disparities, or vary across orders of magnitude depending on the biological state of the tissue (tissue growth, remodeling, fracture healing). The same comment can be made for the available surface area for remodeling.

Determining the appropriate values for the remodeling parameters would thus require large amounts of experimental data. Recent developments in the tissue culture domain using highly controlled bioreactor environments may provide valuable insight at micro-mechanical phenomena, particularly those that relate to the growth rate of tissues e.g. bone apposition rate, cell activity or morphology of grown bone. Particularly, experiments

in the field of tissue engineering that attempt to grow bone on porous scaffolds are relevant to the determination of parameter values in this work. By looking at the speed and amount of bone tissue that grows on the scaffold, one may be able to extract information related to the parameters of this model such as the remodeling velocity and the available surface for remodeling. However, such a parameter fitting endeavor is outside of the scope of this work.

Reassuringly, even without experimental fitting of the parameters, the ranges of values used in this study (which are extracted from [30], themselves being based on parameter values provided in [28]) have been previously shown to produce final remodeled material properties that were qualitatively representative of real bone, provided that accurate boundary conditions are inputted to the simulation, this last requirement conditioning the capability of the remodeling model to produce realistic (at least qualitatively) material property distributions.

This section completes the first part of this chapter, exploring ways in which CDM provides a framework for the study of bone remodeling. Fundamental principles of Continuum Damage Mechanics have been presented, followed by the description of a simple remodeling model. Then, a full remodeling model based on CDM theory is recalled and a study of the effects that parameters in the model have on the outcome of remodeling is made.

In the next part, an application of this model to bone fracture healing is explored.

## III.4 Application to bone fracture healing

In the second part of this chapter, an exploratory application of the previously presented remodeling model to bone fracture healing is made. As stated in the description of the fracture healing process in the introduction of this work, it is the behavior when bone transitions from a soft callus to a bony callus in the case of secondary bone fracture healing that we want to investigate (see Figure I.1 for a graphical reminder). To this end, the geometrical model developed in chapter two, itself based on the works of C.Laurent in [9][10], will be used.

### III.4.1 Preliminary modeling assumptions

In the geometrical model presented in chapter two, the region of extremely low material properties (see Figure II.20) at the middle of the bone is now assigned with the remodeling material studied in the previous sections with material properties that are representative of very weak bone. In this section, the initial apparent density of the tissue will be set at  $\bar{\rho} = 0.2$ , corresponding to a density of  $\rho = 0.38$  [g/cc], a damage value of  $d = 0.98$  and an initial Young's modulus of 179 [MPa]. Essentially, this represents a situation where bone material is almost non-existent, with only an extremely weak collagenous phase present at the fracture site.

To adapt the remodeling model to represent the healing process, the choice is made here to manipulate the values of the homeostatic value of the stimulus  $\psi_t^*$  and the remodeling constants  $c_f$  and  $c_r$ . As has been displayed in the previous section, the homeostatic stimulus is the only parameter in the model that is capable of changing the type of remodeling that occurs under a given loading, as all the other parameters only have an effect on the speed at which that type of remodeling occurs. To prevent any confusion, it shall be noted that healing must be understood here as a gradual increase of the density in the fracture site, or, correspondingly, a decrease in the damage value, and not in the sense of biological processes taking place to reorganize collagenous structures into woven bone and then into lamellar bone. This note is further supported by the fact that the model used here is purely phenomenological, it makes absolutely no claims of representing any underlying micromechanical or biological phenomenon that might be happening in reality.

In order to model the fracture healing process, reducing the homeostatic stimulus value by multiple orders of magnitude would not be surprising, as bone healing processes have been reported to take place even when the bone is in practically immobilized states allowing for only very little interfragmentary movement. It may be assumed from the previously given description of the process of bone fracture healing that extremely little values of stimulus may still enable the healing of the fracture site. The range of values will be determined in a following series of simulations.

Regarding the remodeling velocities  $c_f$  and  $c_r$ , these values should also be expected to vary over orders of magnitude to represent the much faster phenomenon of bone fracture healing. Indeed, previous numerical experiments [28][27][30] have shown that the time needed to go from a perfectly uniform and isotropic bone to a bone with defined features such as a cortical wall and other landmark features in the femoral head was around 300 days ( $\sim 45$  weeks) using the base parameters in Table III.2 (see Figures 5.57 and 5.59 in [30]), whereas fracture consolidation can occur in any time between 4 to 12 weeks

depending on the age of the individual.

The number of daily load cycles  $N$  will be assumed to be 5000, based on typical numbers of steps of normal gait, assuming the animal has no post-operative downtime. Regarding the other parameters involved in the remodeling model, the available area for remodeling will be fixed in this section to  $k = 60\%$  and the lazy zone width is set to 25% of the homeostatic stimulus value, following the values of parameters set in [30][p.188].

The time frame considered in the simulations previously done in chapter 2 is also increased from  $t_{\text{end}} = 1$  [day] to  $t_{\text{end}} = 180$  [days], representing a healing response that is prolonged over a period of half a year. Note that the load application still takes place only during the first day, for which a force of 150 [N] will be applied to the end of the bone, a value that could typically be encountered by canine bone during normal gait [35]. After the first day, only remodeling/healing phenomena occur, i.e. the load is held constant (this load is virtually cycled in the remodeling law through the daily load cycle number  $N$ ).

### III.4.2 Determination of the homeostatic tissue stimulus value

To determine a value of the homeostatic stimulus that leads to healing of the bone, one needs to determine damage characteristics of healthy bone. From the meshes of intact (thus considered healthy) bones previously generated in chapter two, the average bone density is extracted, then from the value of the density, the value of the damage variable can in turn be found. Following that route, an average density of  $\rho = 1.34$  [g/cc] is extracted, leading to a damage value of  $d = 0.66$ . Knowing these values for healthy bone, a series of simulations is done to find a parameter value for the homeostatic stimulus that would lead to the damage value of healthy bone at the end of the simulations. A note to be made here about the interpretation of the damage variable is that a value of 0 does not correspond to a healthy bone state, it corresponds to a bone where all the material has the density of fully mineralized bone, which is obviously non-physiological. Any healthy bone would have a damage value that is closer to 1 than it is closer to 0 due to the voids and the heterogeneity of density found in the material.

A mesh density factor of  $m = 0.5$  is used here. As the loading is not large enough to induce plasticity, a lower value was deemed acceptable due to the trade-off in computation time, the simulations having a final time that is much larger than previously. Due to the fact that time-dependent effects are observed, a maximal time step of 1 [day] is set to guarantee that at least one data point per day is obtained (this maximal time step being, in practice, never reached with the tolerances set in the simulations).

In addition to the previously set parameters, the value for the remodeling velocities  $c_f$  and  $c_r$  is set here at 1 [ $\mu\text{m}/(\text{day MPa})$ ] to hasten the convergence towards the stable value for the damage. This preliminary value is chosen to quickly assess the end values of the damage variable. This value will be refined in the next subsection to match usual time frames observed in clinical fracture repair. Figures III.9b and III.9a show, respectively, damage and density values at the end of simulations.

A value for the homeostatic stimulus may now be determined. In this case, a value of  $\psi_t^* = 5$  [MPa] was selected through this small parameter variation test. It is to be noted that this value is determined for this type of loading only, changing the magnitude or type of loading could have drastic impacts on the stress and strain characteristics found at the fracture site and completely change the healing behavior that is observed here.

Another observation that is made here is that the value of the homeostatic stimulus

is indeed much smaller than for cases that explore bone remodeling. For reasons outlined before and due the much lower forces necessary to trigger bone fracture healing, this newly found lower value reassuringly affirms some of the modeling choices made in the previous subsection.

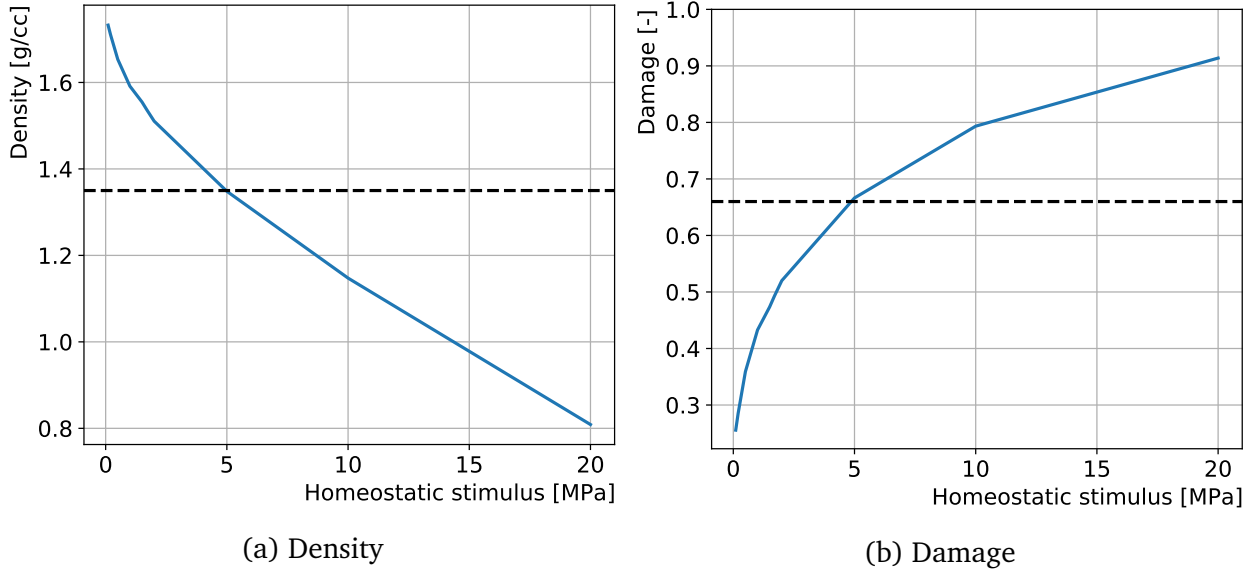


Figure III.9: Final density and damage for varying values of the homeostatic stimulus value  $\psi_t^*$ . The dashed line represents the average properties of healthy bone

### III.4.3 Determination of the remodeling velocity

The remodeling speed will be chosen to match an average reported 12.6 weeks recovery time for femoral fractures in [5]. The matching here is done by looking at density evolution curves to determine a value of the remodeling velocity such that the density evolution is mostly stopped at the  $\sim 90$  days mark. Naturally, as the reported times of fracture healing do not usually match with stages at the biological level, in the sense that bone may still be in the remodeling phase when physicians determine that the appearance of the fracture site is strong enough to sustain daily loads, this estimation cannot be done with high precision. This fuzziness is further accentuated by the fact that biological phases in bone fracture healing overlap, as one process never acts on its own to influence the mechanical outcome of fracture healing.

Using the previously determined parameter for the homeostatic stimulus, a series of simulations varying the value of the remodeling velocities is made. Figures III.10a and III.10b show, respectively, the density and damage evolution for multiple values of the remodeling velocity. As for the determination of the homeostatic stimulus value, a coarser mesh has also been chosen here to save computational resources. From the resulting curves, a value of  $c_f = c_r = 0.4$  [ $\mu\text{m}/(\text{day MPa})$ ] is chosen. The healing behavior using this value of the velocity shows an well advanced evolution of the density and the damage at the 90 days mark, while still allowing some healing to take place after 180 days. This choice was made under the observation that a fully healed bone is usually not present at the 90 days mark, with the last phase of fracture healing (remodeling) possibly lasting years before the bone fully heals.



In a similar way to the value of the homeostatic stimulus value, an observation to be made here is the significantly faster remodeling velocities that must be used to model bone fracture healing compared to slower phenomena such as remodeling (e.g. in [30][p188], where a value of  $c_f = c_r = 0.02 \text{ } [\mu\text{m}/(\text{day MPa})]$  is selected). One may also note that with the chosen value of the remodeling speed, the healing response is mainly decomposed in two phases. The first of which shows a significant increase in density (and reduction in damage) during the first 60 days before significantly slowing down. While the CDM-based model does not claim to represent the biological phenomena happening in the fracture site, the multi-phased behavior of fracture healing, with faster biological phenomena happening at the beginning of the healing response followed by a slower remodeling phase, is elegantly reproduced.

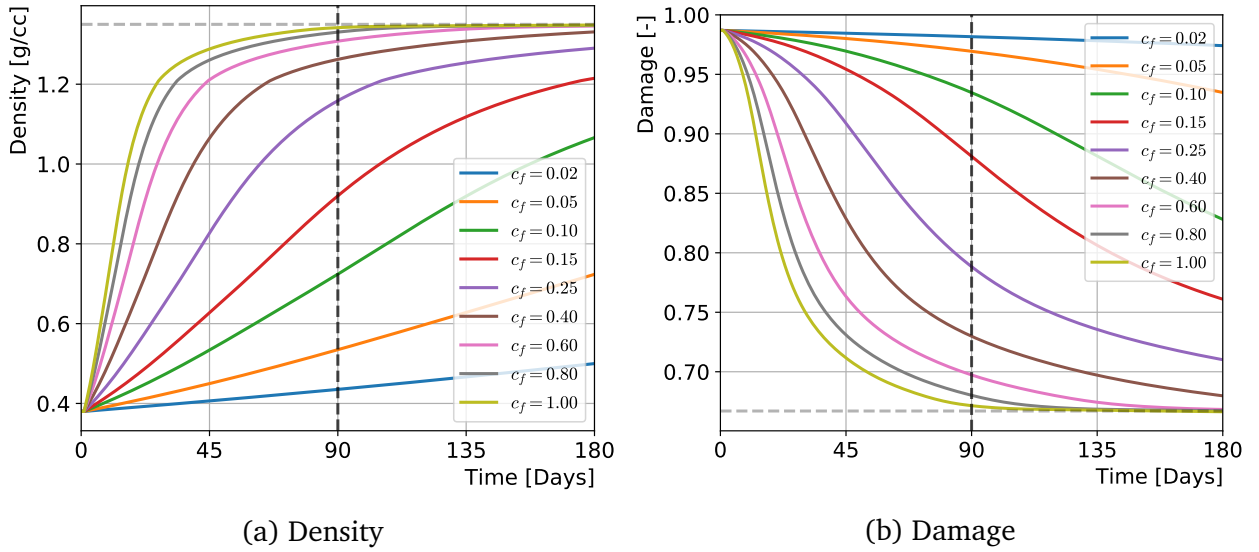


Figure III.10: Density and damage evolution for varying values of the remodeling velocities  $c_f$  and  $c_r$ . Units of the legend in  $[\mu\text{m}/(\text{day MPa})]$

#### III.4.4 Results using determined parameters

Using the previously determined parameter values for the homeostatic stimulus and the remodeling velocity, a full set of simulations with different mesh density factors is run to assess the importance of mesh refinement on the healing behavior of the fracture site. Apart from the coarse meshes (see Figures III.11a, III.11b and III.11c with low  $m$ ) presenting larger geometric inaccuracies seen using the lower values of mesh density factors, no abnormalities are observed between simulations using finer meshes.

The slight differences in the evolution curves likely stem from a combination of multiple factors. The first is the previously observed disparities in material properties between meshes due to the assignment process being able to capture finer details in the original CT volume (summarized in Table II.4), these disparities leading to the differences previously observed in the stress characteristics in the objects (depicted in Figure II.24a), themselves leading to different stimulus values in the fracture site. The second is the coarser meshes leading to geometrical inaccuracies in the representation of the plate and the screws, possibly leading to variations in the stress characteristics found at the fracture site, with a similar effect on the stimulus values. The third is the influence of the mesh-dependent

rigidity of contact discussed in section II.2.5 that also potentially has an impact dependent on the refinement of the mesh. The last effect that adds to the variability stems from the nodal interpolation effects from information at integration points being more prevalent on coarser meshes.

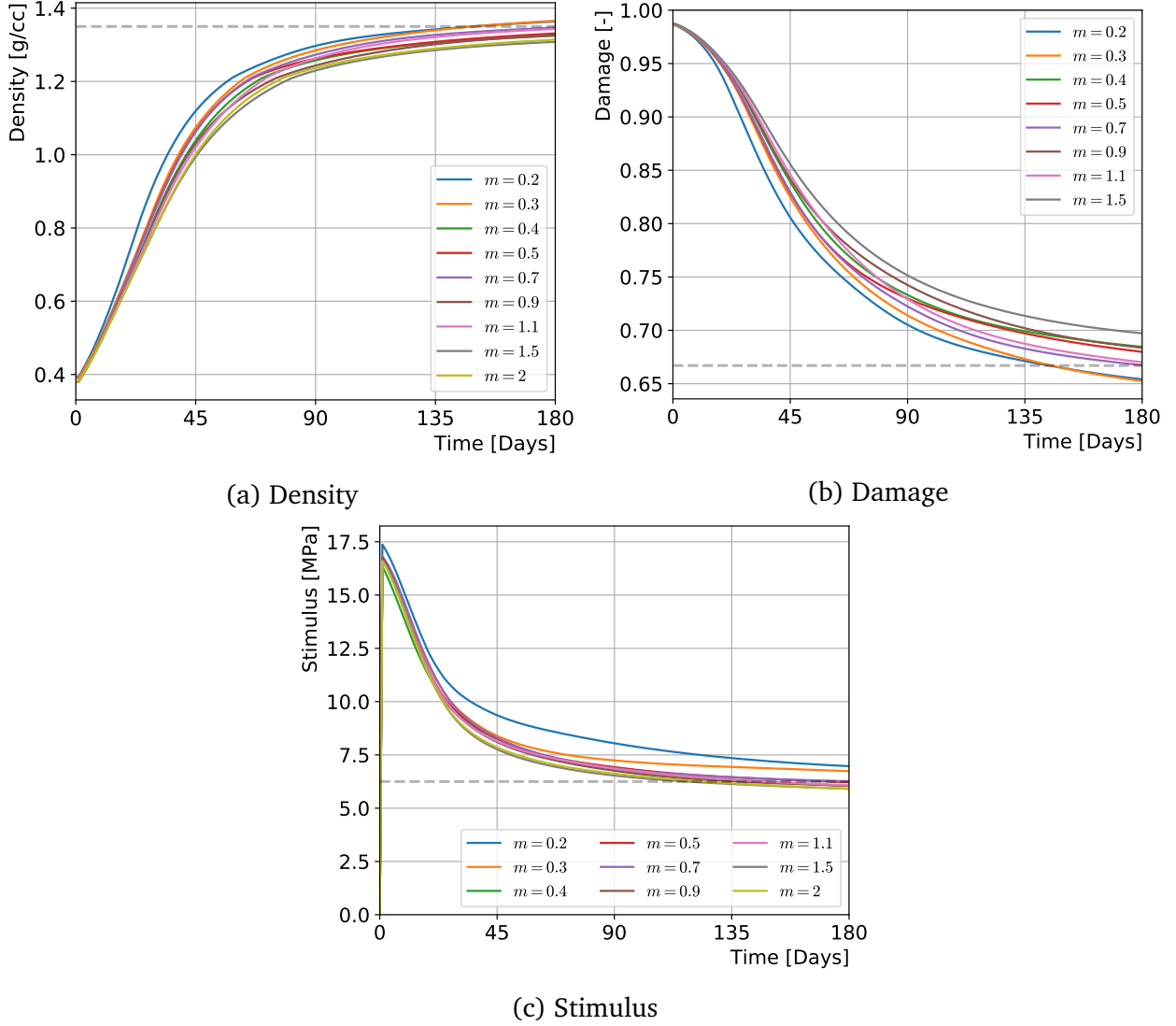


Figure III.11: Density, damage and stimulus evolution for varying values of the mesh density factor. Parameters as defined in the text, and  $\psi_t^* = 5$  [MPa],  $c_f = c_r = 0.4$  [ $\mu\text{m}/(\text{day MPa})$ ], as determined in subsections III.4.2 and III.4.3. For the density and damage evolution the dashed line represents healthy bone properties. For the stimulus, the dashed line represents the edge of the remodeling lazy zone.

Despite the multiple factors inducing variability in the results from this mesh refinement study, the general characteristics of material healing are still observed with a reasonable match between them. The faster healing period before  $\sim 60$  days followed by a gentler evolution after reaching the  $\sim 90$  days mark that were described in previous sections remains observable in all simulations.

The computation times required for these longer evolution observations can be found in Table III.3. With user CPU times ranging from  $\sim 10$  minutes in the coarser meshes to

$m$	0.2	0.3	0.4	0.5	0.7	0.9	1.1	1.5	2
Time (in s)	540	981	1714	2717	6164	11299	15451	33624	49634

Table III.3: Computation times in fracture healing simulations for different mesh density factors.

$\sim 14$  hours of computation time in the densest mesh considered.

Results are presented in the following to demonstrate the effects that bone healing has on the other geometrical objects in the simulation. The following evolution curves are extracted from the simulation using  $m = 0.9$ .

Regarding the evolution of the displacement at constant applied force at the epiphysis of the bone (Figure III.12a), a reduction over time is observed due to the gradual healing of the clot region, increasing the mechanical properties of the bone and thus inducing a displacement curve that seems to resist the applied loads. In traditional mechanical terms, this evolution resembles the behavior of creep, except that it evolves in the opposite way. Instead of observing an increase of the strain over time at constant loading, the strain diminishes until the bone has recovered mechanical properties close to those of intact bone. For equivalent Von Mises stresses in the plate, the bone and the screws (Figures III.12b and III.12c), a similar observation is made, due to the recovery in material properties in the fracture site, more efforts can be taken by the bone at that initially very weak central region, reducing the efforts that the plate and the screws have to sustain. This observation relates neatly to the reason why orthopedic instruments are needed in the first place. The main role of the plate is to sustain the efforts and maintain the shape of the fractured bone until the bone itself is strong enough to sustain them on its own, at which point the implant is (hopefully) not needed anymore. The mean EVMS at the fracture site is shown in Figure III.12d, displaying an increase over time as the fracture site is able to sustain more and more efforts with the increase in material properties due to healing.

### III.4.5 Influence of implant choice and positioning on the healing behavior

Now that the general behavior of bone fracture healing is able to be represented using a CDM-based remodeling model, this section explores the impact of practical geometrical factors that orthopedic surgeons can alter when dealing with bone fracture repair. In this analysis, two main factors are considered. The number and configuration of screws at fixed plate offset, and the plate offset at fixed configuration of screws.

In this section, the numerical parameters involved in the CDM-based remodeling model are fixed. Their values are a combination of the ones that were extracted from [30] and values determined to adapt the remodeling model to represent fracture healing. They are recalled here:

- Number of daily load cycles  $N = 5000$ ,
- Available surface area for remodeling  $k = 60\%$ ,
- Half-width of the remodeling lazy zone  $\Omega = 25\% \psi_t^*$ ,
- Homeostatic tissue stimulus  $\psi_t^* = 5$  [MPa],

- Remodeling velocities  $c_f = c_r = 0.40$  [ $\mu\text{m}/(\text{day MPa})$ ].

Regarding mesh parameters, the simulations in this section will use a mesh density factor of  $m = 0.9$ . Simulation using this mesh fineness take  $\sim 50$  minutes to complete on a 12-cores desktop-class machine.

**Impact of number and configuration of screws** The configurations tested here are, following the numbering of the holes of the plate from left to right in Figure II.22, with holes going from 1 to 8, are found in Table III.4. The configurations using 4 screws modulate on

Config	Nb. screws	Hole	1	2	3	4	5	6	7	8
1	4		■	■		■	■		■	■
2	4		■		■	■	■		■	■
3	4			■	■	■	■		■	
4	5		■	■		■	■		■	■
5	5		■	■		■	■	■	■	
6	5		■			■	■	■	■	■
7	5			■	■	■	■	■	■	■
8	6		■	■	■	■	■	■	■	■

Table III.4: Screw configurations used in impact study on healing behavior. Pink cells indicate positions occupied by the fracture site, blue cells indicate the presence of a screw.

the spacing and the distance to the fracture site and the configurations with more screws are there to investigate the behavior under a higher (projected) implant stiffness. Screw holes 4 and 5 are unavailable due to a conflict with the fracture site, they would be of no use both numerically and orthopedically as well.

Table III.5 summarizes end results observed in simulations using the different screw configurations. No significant difference on the end results and the maximum values reached in the simulation can be discerned based on the configuration. In the general healing behavior and stress evolution curves shown in Figures III.13a, III.13b, III.13c and III.13d, while differences exist, they are not linked to either an increase in the number of screws, nor a change in their configuration. This is evidenced specifically by the results of configuration 7 and 8 where one would expect a very similar behavior, but instead observes behaviors that are at the opposite ends of the (narrow) range of results. What is to be understood here is that trying to significantly or predictably alter the outcome of fracture healing (modeled here) through the number and configuration of screws seems to be inconclusive.

Moreover, the intuitive reasoning behind raising the stiffness by adding more screws to the orthopedic implant is not always confirmed by the results of the simulations. A more accurate reasoning would involve the configuration of the screws and the distance relative to the fracture site in the decision process. Figure III.14 shows the force displacement curves for each tested configuration. If one looks only at the number of screws, the stiffness cannot be accurately judged a priori. It is only when including the specific arrangement of the screws that the results make sense: the stiffest configuration have in common that the screw holes furthest away from the fracture site are occupied (e.g. hole number 8). The stiffness thus seems to be decided as much by the number of screws as by the distance they are situated from the fracture site, with larger distances inducing higher stiffness. This

Configuration	Max displacement	Max plate EVMS	Max bone EVMS	End damage	End density
1	21.2e-3	26.85	10.30	0.683	1.32
2	21.0e-3	29.47	7.69	0.718	1.27
3	23.6e-3	29.49	6.96	0.706	1.29
4	20.6e-3	28.87	8.38	0.714	1.28
5	22.6e-3	30.05	7.65	0.713	1.28
6	21.1e-3	27.35	9.48	0.684	1.32
7	21.3e-3	29.24	7.25	0.681	1.32
8	20.3e-3	29.94	7.02	0.708	1.29

Table III.5: End results of simulations with different screw configurations.

observation is likely explained by the fact that having a larger distance between screws leaves less bone material between the epiphysis and the first encountered screw to compress, thus leaving more efforts to be sustained by the stiffer steel implants, in turn raising the stiffness.

**Impact of plate offset** The plate offset is modulated from 0.0 to 3.0 [mm] in 0.5 [mm] steps. In theory, a shorter offset should increase the stiffness of the implant as reported in [36], leading to potentially more stress shielding effects and reducing the speed of the healing process. In the opposite way, increasing the offset would in theory have potentially beneficial effects on healing.

Figure III.15a shows the force displacement curves at the start of the simulation before any remodeling effects take place, where one can, as expected, observe increased stiffness for plate configurations with a smaller offset. Figure III.15b displays the maximum displacement observed after the initial application of the loads. Everything else being equal, increasing plate offset decreases the stiffness of the whole apparatus. In contrast with the screw configurations, the choice of the plate offset is an actionable parameter on which a surgeon may act upon to control the stiffness of the repaired bone, with healing optimization in mind.

Regarding the healing results found in Figures III.16a and III.16b, aside from the close to 0 plate offset values, the final healing response follows the aforementioned trend. Higher implant stiffness tends to decrease final density characteristics found at the fracture site. This result likely stems from the lower values of stimulus, themselves due to lower stresses, encountered at the fracture site due to the plate taking more of the efforts generated by the compression.

Due to the very crude application of boundary conditions in this work. The results of healing modeling using different implant configurations should not be compared to real-world healing experimental/clinical data, as the imposed loading does not reproduce the complex boundary conditions generated by normal gait, or even altered gait observed during the wound healing period.

The results do however provide relevant information about the behavior of the model when compared to each other. All else being equal, the healed properties are overall higher when larger plate offsets are used (see end of Figures III.16a and III.16b).

As mentioned previously, due to the exploratory nature of this work, the boundary conditions that are used here to represent the loading that a fracture site would encounter during fracture healing are simplistic and questionably representative of clinical practice. In the previous works investigating the remodeling phenomena, a high sensitivity to the boundary conditions is reported. For example, even with (more) realistic applications of boundary conditions on a femoral head (see Figure 5.59 in [30]), artifacts in the final remodeled morphology could be seen due to the localized nature of numerically applied forces (forces were applied on small areas of the mesh, generating a slight stress concentration locally). These artifacts were visible lines of over-remodeled material following the stress lines generated by these point loads. It would thus be expected that, in this work too, the realism of boundary conditions is the number one factor influencing the healing behavior of the fracture site. While the crude application of a simple compression may provide information about the model behavior, it makes no claim of being an accurate portrayal of the forces generated by post-operative gait.

A note for future users of this model: upon setting updated boundary conditions, the homeostatic stimulus value will likely need to be further refined to produce final bone properties that are representative of healthy bone. The same comment can be made if one chooses to use the full anisotropic version of the remodeling law proposed in [25], in which case care should be given to the selection of appropriate values of the remodeling parameters.

A caveat of this work is that all the material properties assigned to the bone stem from density-property relations determined for human bone. This choice is made for two main reasons, the first is due to the scarcity of density-property relations adapted to veterinary cases. Previous works [9][10] interpolated available material property data from canine samples and fitted a density-relation made for human bones to the canine data. In the interest of limiting the already large list of sources of approximations made in this work, the choice has been made to ignore the density-property fitting to canine data and use available laws calibrated for human bone. The second reason is that some of the density-property relations are explicitly coded into the remodeling law used in this work and access was not given to modify them. As having two different density-property relations would cause translation errors between the assignation step in mesh generation and the use of the remodeling law during simulations, the choice was made to apply the density-property relations found in [30], i.e. relations extracted from human bone properties. The result of this choice is an unusual mismatch of canine bone geometries, assorted with loading representative of canine gait, but material properties stemming from relations determined for human bones. Further work using this model should investigate ways to harmonize the density-property laws involved, either by including relations adapted to canine bone in the remodeling model, or by using geometries and loading from human bones and gait, respectively.

Nevertheless, even with that unusual mismatch, the resulting extended model is capable of representing fracture healing after a modification in the parameter values of the remodeling model. At least in the mechanical sense, material properties gradually recover to arrive to mean densities observed in intact bones.

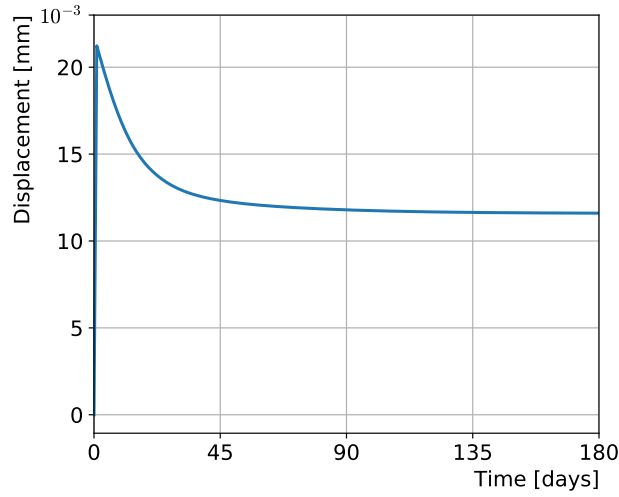
In summary, parameter values to adapt the remodeling model to represent bone fracture healing have been determined in this section. The resulting model was able to reproduce the multi-phased healing response commonly observed in fracture healing, with a faster recovery at the start before slowing down to allow remodeling phenomena to take place.



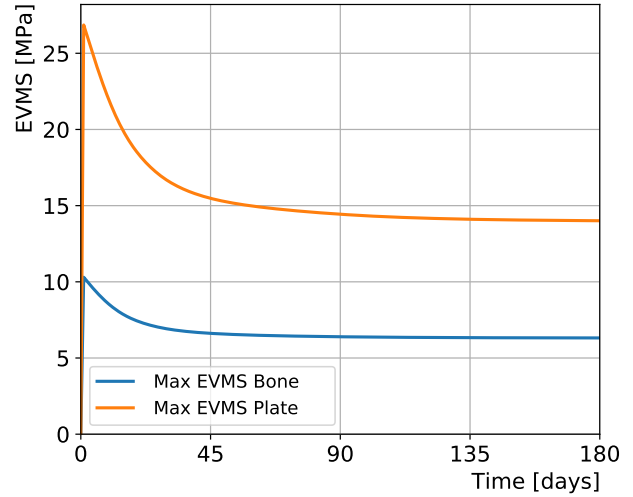
Then, a study of the impact of the screw number and configuration, and the plate offset on the healing behavior of the bone fracture site has been conducted.

### III.5 Chapter summary

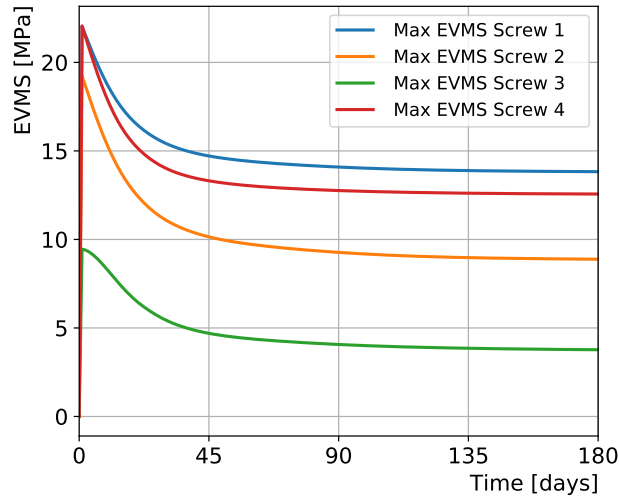
In this chapter, the use of a Continuum Damage Mechanics-based remodeling model [29] has been extended to represent bone fracture healing. The principles and requirements of a CDM-based model have been introduced using the example of isotropic damage. Then, one of the earliest mathematical models capable of representing bone remodeling (Stanford model) was described to initiate the reader to the representation of bone remodeling. Combining the fields of CDM and remodeling, a CDM-based isotropic remodeling model was recalled and a sensitivity analysis portraying the influence of the parameters involved in that model was conducted. After gaining familiarity with the effects of the parameters on the remodeling behavior, the previously described CDM-based remodeling model were extended to represent bone fracture healing. Relevant parameters in the model were singled out for their capability to induce different healing behaviors. Then, appropriate values of those parameters were determined to match the simulated healing behavior with commonly observed healing patterns in the literature. Information about the properties of healthy bone were used to determine the value of the other main remodeling parameter. Using the determined parameters, a multi-phased healing behavior is displayed by the adapted remodeling model. The effect of material healing on the other geometrical objects present in the simulation was shown. Finally, a series of implant configurations were tested to determine their effect on the healing behavior of the fracture site.



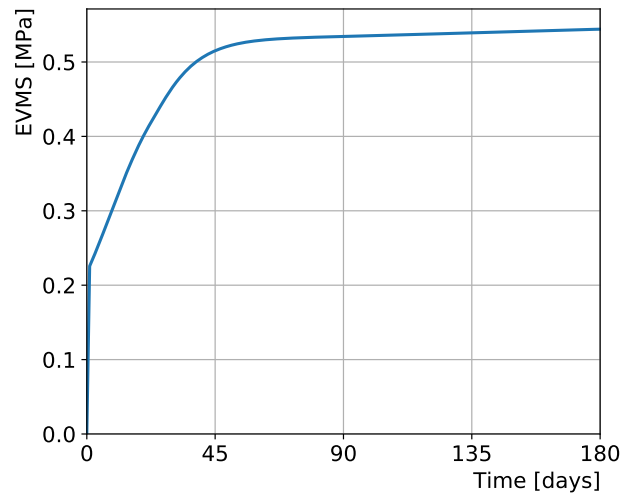
(a) Epiphysis displacement



(b) Maximum EVMS plate and bone

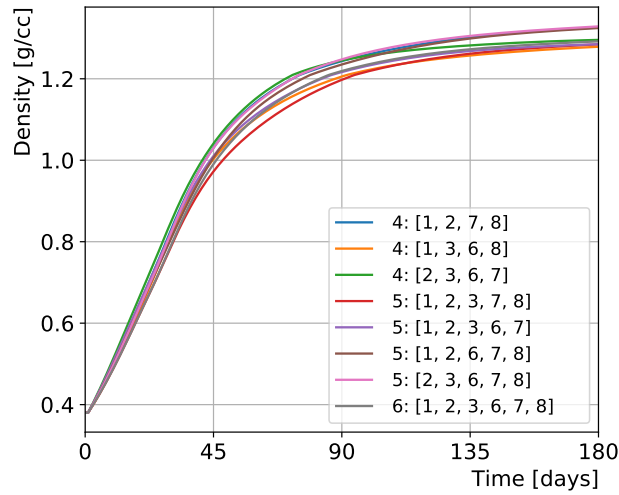


(c) Maximum EVMS screws

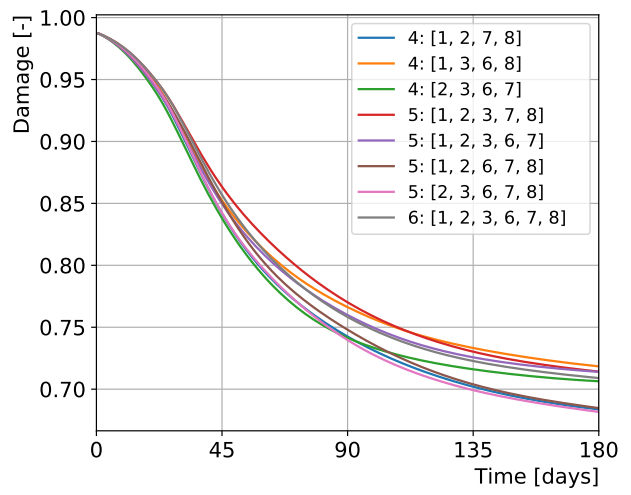


(d) Mean EVMS at fracture site

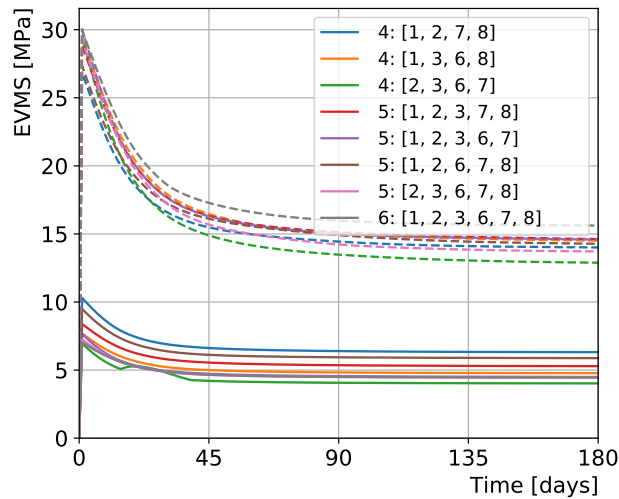
Figure III.12: Evolution of EVMS characteristics in the plate, the bone and the clot and epiphysis displacement evolution during modeled bone fracture healing. Parameters as defined in the modeling assumptions and  $\psi_t^* = 5$  [MPa],  $c_f = c_r = 0.4$  [ $\mu\text{m}/(\text{day MPa})$ ], mesh density factor  $m = 0.9$ .



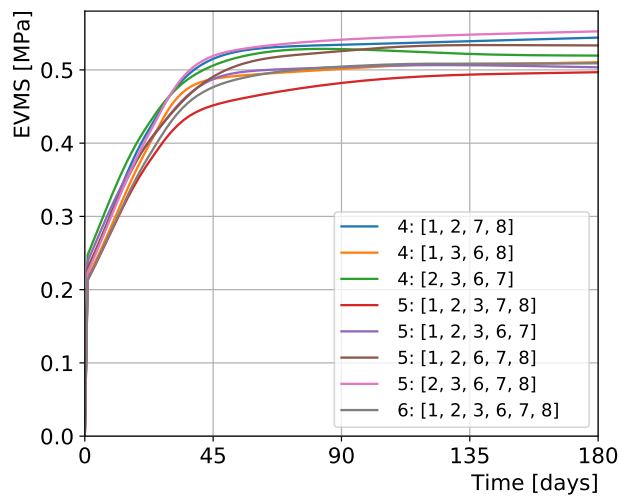
(a) Density



(b) Damage



(c) Plate (dashed) and Bone (plain) maximum EVMS



(d) Fracture site mean EVMS

Figure III.13: Evolution of mean damage and density in the clot using different screw configurations. Evolution of EVMS in the plate, the bone and at the fracture site. The plate offset is held constant at 2 [mm]. The legends display the number of screws and the configuration.  $m = 0.9$ .

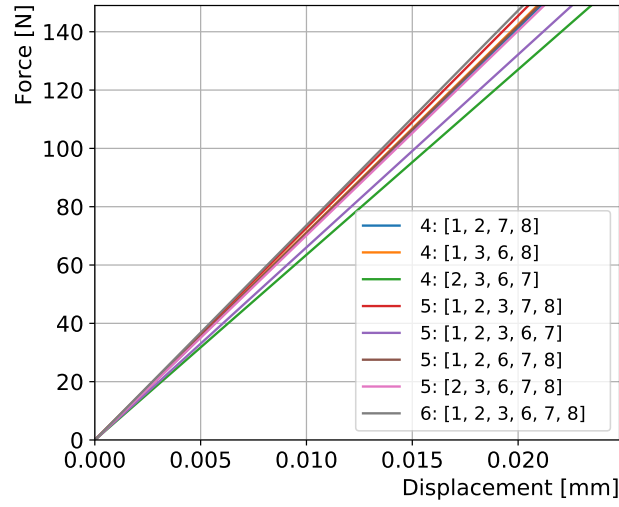


Figure III.14: Force displacement curves for different screw configurations. The plate offset is held constant at 2 [mm]. The legend displays the number of screws and the configuration.  $m = 0.9$ .

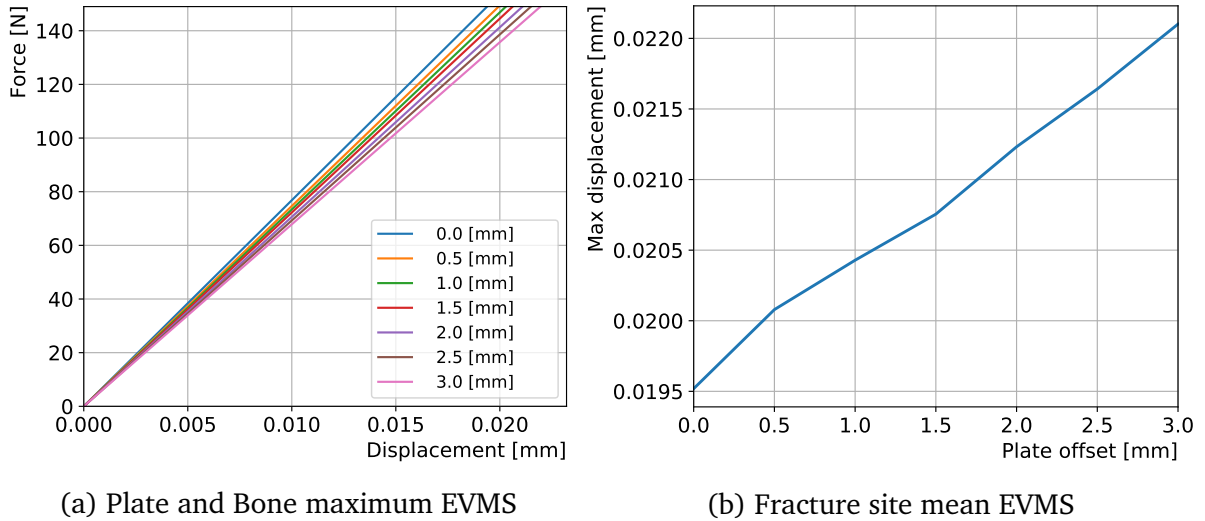
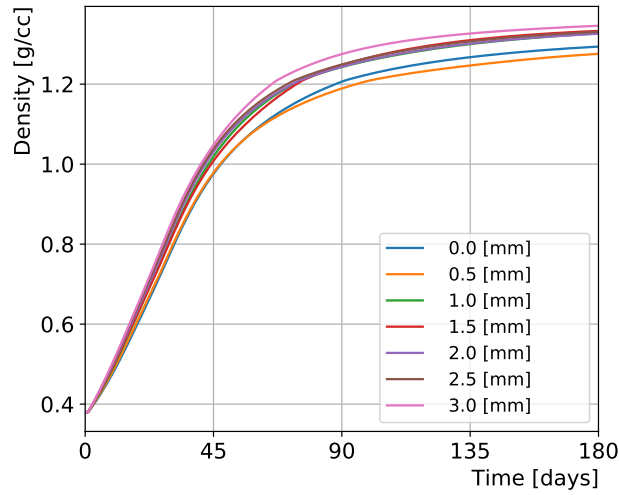
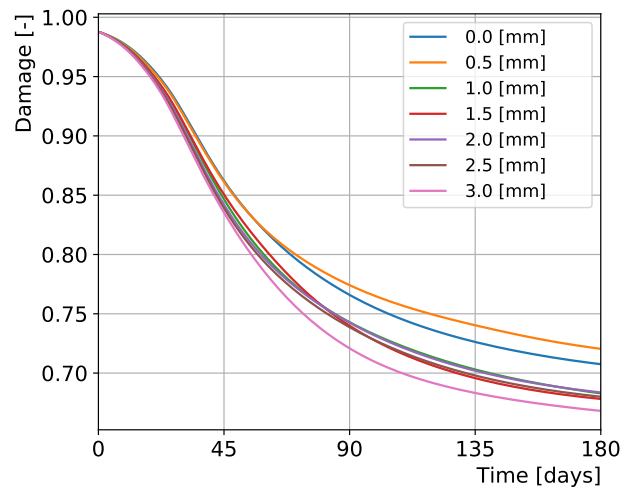


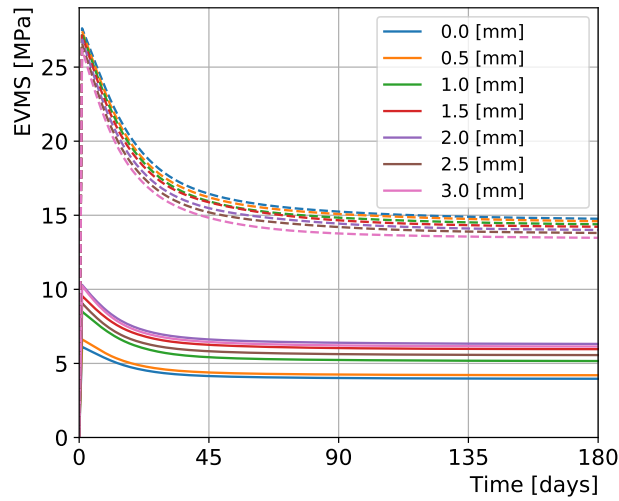
Figure III.15: Force displacement curves and maximum displacement for different plate offsets. The screws use configuration 1.  $m = 0.9$ .



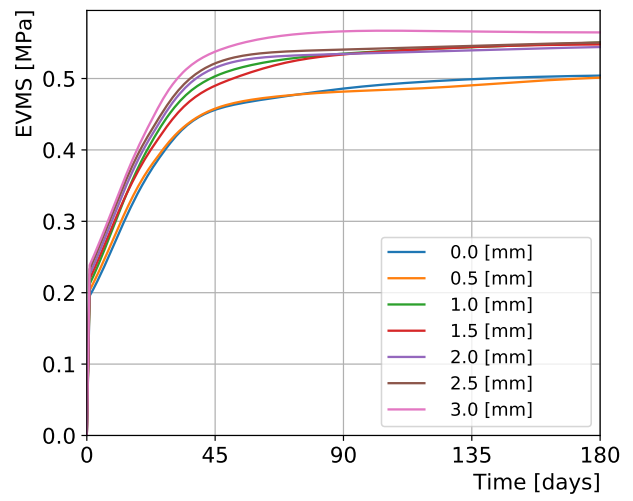
(a) Density



(b) Damage



(c) Plate and Bone maximum EVMS



(d) Fracture site mean EVMS

Figure III.16: Evolution of mean damage and density in the clot using different plate offsets. Evolution of EVMS seen in the plate, the bone and at the fracture site. Configuration 1 is used for the screw, i.e. screws in holes number 1, 2, 7 and 8.

## IV Conclusions and perspectives

Managing the outcome of bone fracture healing is one of the most challenging tasks that a veterinary orthopedic surgeon has to accomplish. On top of the difficulties involved with unusual patient morphologies, tight budgets and often short times to plan out surgeries in traumatic cases, surgeons have to deal with the delicate trade-off between assuring fracture stability and leaving enough interfragmentary motion to promote optimal bone healing on a case by case basis. While handbooks provide general guidelines for surgeons to follow in some cases, decisions are still often based on their experience and expertise gained on previously cases.

Patient-specific models have been seen as a promising and powerful tool to assist the surgeons in their decision-making and improve the outcome of surgical operations. In this context, a series of models has been developed in aims to reproduce experimental results with numerical equivalents. These models sparked the development of multiple tools required to translate information found in medical images to meshes suitable for use in Finite Element simulations, as well as complex material models designed to numerically represent the characteristic capability that bones have to adapt depending on the loading that is imposed on them.

In this work, a continuation of these efforts was made by extending the use cases of a bone remodeling model based on Continuum Damage Mechanics theory to represent bone fracture healing.

To this end, improvements to the image-to-mesh pipeline to go from a raw computerized tomography volume to a final volume mesh used in simulations were made. These modifications had two main objectives: the first was to increase the ease of use and modularity of the image to mesh pipeline and the second was to improve the quality of the generated meshes. For the first objective, the previous mono-block implementation is separated in clearly defined stages to increase the modularity and facilitate modifications to each stage. To increase the ease of use, a didactic description of all the steps involved in the image-to-mesh pipeline was proposed in this manuscript. For the second objective, the previously used decimation algorithm and geometry-specific cleaning operations were swapped in favor of openly available remeshing and mesh-fixing tools. In simulations themselves, the previously used contact management method was found to be over-constraining and an improved version was advanced in this work.

Then, to familiarize the reader to the use of damage models and to numerical representations of bone remodeling, the disciplines were first introduced separately using a simple example for damage mechanics and the historically first mathematically complete model for bone remodeling. The frameworks were then combined to form the Continuum Damage Mechanics-based remodeling model used in this work.

After a parameter sensitivity study to illustrate the effects of varying remodeling parameters involved in the model, an extension was proposed to adapt the remodeling model

to represent bone fracture healing. To achieve this, relevant parameters were selected and varied outside of their initial range to adequately capture the mechanical healing effects and time frames seen in the faster phenomenon that is fracture healing. Finally, as a demonstration of the adapted remodeling model (and the improved meshing pipeline), the impact on the healing behavior of the fracture of a few geometrical factors on which an orthopedic surgeon has control was investigated.

Key outcomes and limitations of this work are recalled here.

### Key outcomes

- The image-to-mesh pipeline is more modular and beginner-friendly, each step necessary for the creation of the ultimately used volume meshes is assorted with a didactic description in this text.
- The manual and mesh-specific cleaning operations are replaced by openly available mesh manipulation tools with a proven track-record. Magical coefficients that could make-or-break the image-to-mesh pipeline have been removed. These changes increase the robustness of the mesh generation pipeline.
- A significant improvement to the resulting mesh quality is achieved through a combination of more robust mesh cleaning tools and improved volume meshing algorithms.
- Following the modifications in the pipeline, previously observed spurious losses of volume throughout the meshing process are significantly reduced in low-to-medium density meshes, and virtually suppressed in higher density meshes.
- The over-constraining contact management method is relaxed, resulting in physically more adequate stress fields in the screws.
- A Continuum Damage Mechanics-based isotropic remodeling model is thoroughly introduced using a traditional damage mechanics model and a bone remodeling model.
- Said Continuum Damage Mechanics-based remodeling model is extended to represent the behavior of bone fracture healing.
- The extended model is able to capture the multi-staged healing behavior observed in bone fracture healing.
- Some orthopedic implant geometric parameters are shown to be able to influence the outcome of fracture healing represented using the extended remodeling model.

### Limitations

- Ideally, all the meshing process should be doable outside of *Metafor* using openly available or more robust tools. Currently, the initial surface mesh reconstruction from the CT volume still relies on the *geniso* tools integrated to *Metafor*.
- Surface meshes generated using the current pipeline are uniform, i.e. polygons are all of similar size. Creating meshes with polygon density depending on curvature is currently not feasible.



- Surface mesh cleaning operations currently cannot be done on a localized region of the mesh, leading to computational overhead in the mesh cleaning steps. Another consequence of this is mesh to mesh variability when different number of screws are used in the orthopedic implant. The cleaning algorithm slightly alters the whole mesh instead of only cleaning the areas affected by boolean cutting.
- The distinction of the fracture region is done based on material property (Young's modulus). Instead, a label should be assigned to elements in the fracture site during the mesh generation phase to prevent any other element in the bone mesh from receiving the properties of remodeling bone.
- The current contact management method, while improved over the previous over-constraining method, is still slightly mesh density dependent.
- Regarding material laws, the currently implemented laws are deduced from human samples, but the geometries involved are canine femurs. This limitation is tied to the density-property laws coded in the remodeling model.
- The use of the damage mechanics framework seems to be quite a niche in the study of bone remodeling, as a consequence, material laws suffer from translation approximations going from relations involving the density to relations involving the damage variable.

**Perspectives** Some perspectives and possible improvements that could be made to this work are described in the following.

- Globally, improving the realism of boundary conditions could have major impacts on the accuracy and validity of results output by remodeling algorithms. This is however likely far-fetched, as even for human bones, precise boundary conditions on specific bones during normal gait are not fully available. Another added difficulty is the application of these boundary conditions on the mesh of the bone, which likely would be sensitive to stress concentration effects.
- Regarding the image-to-mesh pipeline, implementing cleaning algorithms that are capable of selecting specific regions of the mesh would be an improvement. This could allow much more consistent meshes to be produced when varying implant configurations, leading to less error stemming from the meshing process.
- Similarly, cleaning and remeshing algorithms that take into account the curvature of the underlying surface could help save computation time in the long run.
- Even though an improved version of the contact management is proposed in this work, implementing proper sticking contact using penalties that are independent of mesh density would still be an improvement over the current method.
- Currently used material models for bone are isotropic, even though the literature has multiple accounts evidencing that bone is an orthotropic material. The limitation lies in the determination of local orthotropy axes starting from the raw CT volume. Techniques to determine these axes exist at the micro-CT level, but clinical CT scanners do not possess the resolution necessary to extract orthotropy axes from the resulting CT image.

- Instead of using a very detailed implicit geometry of a CAD generated plate, generating the geometries of the plate and screws directly using CAD software would allow to suppress the edge smoothing artifacts introduced by the cleaning algorithms used in this work. Integrating developments made in a previous Master's thesis [36] to the implant generation procedure would be the first step in this direction.
- With more time, a full mesh refinement study involving the plate, the screws and the bone separately could allow to determine appropriate mesh refinement individually for each geometrical object. The mesh refinement studies carried out in this work vary the mesh density of the three objects concurrently, making it difficult to track potential sources of errors.
- A study could be done to investigate the appropriate number of material bins to use during the import of property assigned volume meshes.
- Regarding the use of the Continuum Damage Mechanics model, one could investigate the possibility of setting the homeostatic stimulus value based on the material properties of healthy bone.
- The values of the other parameters in the remodeling model may potentially be determined from experimental data from the field of tissue engineering. A thorough literature survey may help in the linking of these remodeling parameters to observable biological variables.
- The remodeling model (and the healing model extended from it) use the isotropic version of the full model developed in [30]. Using more realistic boundary conditions, one could try to use the full anisotropic version and observe how material properties align with principal stress lines. This step will likely require that orthotropic material properties be set first, in order to prevent a mismatch between anisotropic remodeling at the fracture site and the isotropic material properties of surrounding tissue.

# Bibliography

- [1] Mohammad S. Ghiasi et al. "Bone fracture healing in mechanobiological modeling: A review of principles and methods". In: *Bone Reports* 6 (June 2017), pp. 87–100. DOI: 10.1016/j.bonr.2017.03.002. URL: <https://doi.org/10.1016/j.bonr.2017.03.002>.
- [2] Mohammad S. Ghiasi et al. "Computational modeling of human bone fracture healing affected by different conditions of initial healing stage". In: *BMC Musculoskeletal Disorders* 20.1 (Nov. 2019). DOI: 10.1186/s12891-019-2854-z. URL: <https://doi.org/10.1186/s12891-019-2854-z>.
- [3] S. Barry. "Non-steroidal anti-inflammatory drugs inhibit bone healing: A review". In: *Veterinary and Comparative Orthopaedics and Traumatology* 23.06 (2010), pp. 385–392. DOI: 10.3415/vcot-10-01-0017. URL: <https://doi.org/10.3415/vcot-10-01-0017>.
- [4] Richard Marsell and Thomas A. Einhorn. "The biology of fracture healing". In: *Injury* 42.6 (June 2011), pp. 551–555. DOI: 10.1016/j.injury.2011.03.031. URL: <https://doi.org/10.1016/j.injury.2011.03.031>.
- [5] Thomas A. Einhorn and Louis C. Gerstenfeld. "Fracture healing: mechanisms and interventions". In: *Nature Reviews Rheumatology* 11.1 (Sept. 2014), pp. 45–54. DOI: 10.1038/nrrheum.2014.164. URL: <https://doi.org/10.1038/nrrheum.2014.164>.
- [6] Leah C. Jackson and Philip D. Pacchiana. "Common complications of fracture repair". In: *Clinical Techniques in Small Animal Practice* 19.3 (Aug. 2004), pp. 168–179. DOI: 10.1053/j.ctsap.2004.09.008. URL: <https://doi.org/10.1053/j.ctsap.2004.09.008>.
- [7] David N Kugelman et al. "A Recurrent Stress Fracture of the Humerus following Fixation: The Effect of Implant Stress Shielding". In: *Journal of orthopaedic case reports* 9.2 (2019), pp. 3–6. ISSN: 2250-0685. DOI: 10.13107/jocr.2250-0685.1342.
- [8] Xinwen Zhao et al. "An experimental study on stress-shielding effects of locked compression plates in fixing intact dog femur". In: *Journal of Orthopaedic Surgery and Research* 16.1 (Jan. 2021). DOI: 10.1186/s13018-021-02238-3. URL: <https://doi.org/10.1186/s13018-021-02238-3>.
- [9] Cédric P Laurent et al. "Prediction of the mechanical response of canine humerus to three-point bending using subject-specific finite element modelling". In: *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine* 230.7 (Apr. 2016), pp. 639–649. DOI: 10.1177/0954411916644269. URL: <https://doi.org/10.1177/0954411916644269>.

- [10] Cédric P Laurent et al. “Effect of orthopedic implants on canine long bone compression stiffness: a combined experimental and computational approach”. In: *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine* 234.3 (Oct. 2019), pp. 255–264. DOI: [10.1177/0954411919882603](https://doi.org/10.1177/0954411919882603). URL: <https://doi.org/10.1177/0954411919882603>.
- [11] Vinciane d’Otreppe. “From medical imaging to finite element simulations: a contribution to mesh generation and locking-free formulations for tetrahedra”. In: *Université de Liège, Liège, Belgique* (Dec. 2012). URL: <http://hdl.handle.net/2268/136159>.
- [12] J. Ahrens, Berk Geveci, and Charles Law. “ParaView: An End-User Tool for Large Data Visualization”. In: *Visualization Handbook* (Jan. 2005).
- [13] Andriy Fedorov et al. “3D Slicer as an image computing platform for the Quantitative Imaging Network”. In: *Magnetic Resonance Imaging* 30.9 (2012). Quantitative Imaging in Cancer, pp. 1323–1341. ISSN: 0730-725X. DOI: <https://doi.org/10.1016/j.mri.2012.05.001>.
- [14] W. Schroeder et al. *The Visualization Toolkit: An Object-oriented Approach to 3D Graphics*. Kitware, 2006. ISBN: 9781930934191. URL: <https://books.google.be/books?id=rx4vPwAACAAJ>.
- [15] Vinciane d’Otreppe, Romain Boman, and Jean-Philippe Ponthot. “Generating smooth surface meshes from multi-region medical images”. In: *International Journal for Numerical Methods in Biomedical Engineering* 28.6-7 (2012), pp. 642–660. DOI: <https://doi.org/10.1002/cnm.1471>.
- [16] C. Bane Sullivan and Alexander Kaszynski. “PyVista: 3D plotting and mesh analysis through a streamlined interface for the Visualization Toolkit (VTK)”. In: *Journal of Open Source Software* 4.37 (May 2019), p. 1450. DOI: [10.21105/joss.01450](https://doi.org/10.21105/joss.01450). URL: <https://doi.org/10.21105/joss.01450>.
- [17] Sebastien Valette, Jean-Marc Chassery, and Remy Prost. “Generic Remeshing of 3D Triangular Meshes with Metric-Dependent Discrete Voronoi Diagrams”. In: *IEEE Trans. Vis. Comput. Graph.* 14 (Mar. 2008), pp. 369–381. DOI: [10.1109/TVCG.2007.70430](https://doi.org/10.1109/TVCG.2007.70430).
- [18] Sébastien Valette and Jean-Marc Chassery. “Approximated Centroidal Voronoi Diagrams for Uniform Polygonal Mesh Coarsening”. In: *Computer Graphics Forum* 23 (Sept. 2004). DOI: [10.1111/j.1467-8659.2004.00769.x](https://doi.org/10.1111/j.1467-8659.2004.00769.x).
- [19] Sébastien Valette, I. Kompatsiaris, and J.M. Chassery. “Adaptive Polygonal Mesh Simplification With Discrete Centroidal Voronoi Diagrams”. In: *Proceedings of 2nd International Conference on Machine Intelligence ICMI 2005*. Tozeur, Tunisia, Nov. 2005, pp. 655–662.
- [20] M. Attene. “A lightweight approach to repairing digitized polygon meshes”. In: *The Visual Computer* 26 (2010), pp. 1393–1406.
- [21] Christophe Geuzaine and Jean-François Remacle. “Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities”. In: *International Journal for Numerical Methods in Engineering* 79.11 (2009), pp. 1309–1331. DOI: <https://doi.org/10.1002/nme.2579>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2579>.

- [22] Hang Si. “TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator”. In: *ACM Trans. Math. Softw.* 41.2 (Feb. 2015). ISSN: 0098-3500. DOI: [10.1145/2629697](https://doi.org/10.1145/2629697).
- [23] Dieter Christian Wirtz et al. “Critical evaluation of known bone material properties to realize anisotropic FE-simulation of the proximal femur”. In: *Journal of Biomechanics* 33.10 (Oct. 2000), pp. 1325–1330. DOI: [10.1016/s0021-9290\(00\)00069-5](https://doi.org/10.1016/s0021-9290(00)00069-5). URL: [https://doi.org/10.1016/s0021-9290\(00\)00069-5](https://doi.org/10.1016/s0021-9290(00)00069-5).
- [24] Elise F. Morgan, Ginu U. Unnikrisnan, and Amira I. Hussein. “Bone Mechanical Properties in Healthy and Diseased States”. In: *Annual Review of Biomedical Engineering* 20.1 (2018). PMID: 29865872, pp. 119–143. DOI: [10.1146/annurev-bioeng-062117-121139](https://doi.org/10.1146/annurev-bioeng-062117-121139). URL: <https://doi.org/10.1146/annurev-bioeng-062117-121139>.
- [25] Marlène Mengoni, Vinciane d’Otreppe, and Jean-Philippe Ponthot. “A bone remodelling model for long term orthodontic tooth movement”. In: *Journal of Biomechanics* 45 (July 2012), S180. DOI: [10.1016/s0021-9290\(12\)70181-1](https://doi.org/10.1016/s0021-9290(12)70181-1). URL: [https://doi.org/10.1016/s0021-9290\(12\)70181-1](https://doi.org/10.1016/s0021-9290(12)70181-1).
- [26] M. Mengoni and J.P. Ponthot. “Isotropic continuum damage/repair model for alveolar bone remodeling”. In: *Journal of Computational and Applied Mathematics* 234.7 (Aug. 2010), pp. 2036–2045. DOI: [10.1016/j.cam.2009.08.061](https://doi.org/10.1016/j.cam.2009.08.061). URL: <https://doi.org/10.1016/j.cam.2009.08.061>.
- [27] M. Doblaré and J.M. Garcia. “Anisotropic bone remodelling model based on a continuum damage-repair theory”. In: *Journal of Biomechanics* 35.1 (Jan. 2002), pp. 1–17. DOI: [10.1016/s0021-9290\(01\)00178-6](https://doi.org/10.1016/s0021-9290(01)00178-6). URL: [https://doi.org/10.1016/s0021-9290\(01\)00178-6](https://doi.org/10.1016/s0021-9290(01)00178-6).
- [28] M. Doblaré and J.M. Garcia. “Application of an anisotropic bone-remodelling model based on a damage-repair theory to the analysis of the proximal femur before and after total hip replacement”. In: *Journal of Biomechanics* 34.9 (Sept. 2001), pp. 1157–1170. DOI: [10.1016/s0021-9290\(01\)00069-0](https://doi.org/10.1016/s0021-9290(01)00069-0). URL: [https://doi.org/10.1016/s0021-9290\(01\)00069-0](https://doi.org/10.1016/s0021-9290(01)00069-0).
- [29] M. Mengoni and J.P. Ponthot. “An enhanced version of a bone-remodelling model based on the continuum damage mechanics theory”. In: *Computer Methods in Biomechanics and Biomedical Engineering* 18.12 (Apr. 2014), pp. 1367–1376. DOI: [10.1080/10255842.2014.903933](https://doi.org/10.1080/10255842.2014.903933). URL: <https://doi.org/10.1080/10255842.2014.903933>.
- [30] Marlène Mengoni. “On the development of an integrated bone remodeling law for orthodontic tooth movements models using the Finite Element Method.” In: *Université de Liège, Liège, Belgique* (Aug. 2012). URL: <http://hdl.handle.net/2268/126082>.
- [31] J. L. Chaboche. “Continuum Damage Mechanics: Part I—General Concepts”. In: *Journal of Applied Mechanics* 55.1 (Mar. 1988), pp. 59–64. DOI: [10.1115/1.3173661](https://doi.org/10.1115/1.3173661). URL: <https://doi.org/10.1115/1.3173661>.
- [32] G. S. Beaupré, T. E. Orr, and D. R. Carter. “An approach for time-dependent bone modeling and remodeling-theoretical development”. In: *Journal of Orthopaedic Research* 8.5 (Sept. 1990), pp. 651–661. DOI: [10.1002/jor.1100080506](https://doi.org/10.1002/jor.1100080506). URL: <https://doi.org/10.1002/jor.1100080506>.

- [33] G. S. Beaupré, T. E. Orr, and D. R. Carter. “An approach for time-dependent bone modeling and remodeling-application: A preliminary remodeling simulation”. In: *Journal of Orthopaedic Research* 8.5 (Sept. 1990), pp. 662–670. DOI: 10.1002/jor.1100080507. URL: <https://doi.org/10.1002/jor.1100080507>.
- [34] M. Mengoni et al. “A non-linear homogeneous model for bone-like materials under compressive load”. In: *International Journal for Numerical Methods in Biomedical Engineering* 28.2 (Nov. 2011), pp. 273–287. DOI: 10.1002/cnm.1473. URL: <https://doi.org/10.1002/cnm.1473>.
- [35] *Canine Gait Analysis*. URL: <https://veteriankey.com/gait-analysis/>. Accessed 15 July, 2021.
- [36] Alice Bouchoms. “Numerical modelling of the mechanical response of bones repaired with orthopedic implants.” In: *Université de Liège, Liège, Belgique* (2015).