Division of Solid Mechanics

ISRN LUTFD2/TFHF--02/5095--SE (1-67)

ADAPTIVE BONE REMODELLING

Master's Dissertation by

Magnus Harrysson

Supervisors
Ingrid Svensson, Div. of Solid Mechanics
Ian McCarthy, Biomechanics Laboratory,
Dep. Of Orthopaedics,
Lund University Hospital

For information, address:
Division of Solid Mechanics, Lund University, Box 118, SE-221 00 Lund, Sweden.
Homepage: http://www.solid.lth.se

Acknowledgement

The work was carried out at the Division of Solid Mechanics at Lund University, Sweden, under the supervision of Dr. Ingrid Svensson, Division of Solid Mechanics and Dr. Ian McCarthy, Biomechanics Laboratory, Department of Orthopaedics, Lund University Hospital.

I wish especially to thank Ingrid Svensson and Ian McCarty for their guidance and advice, which has helped me a lot during this time. I would also like to thank Paul Håkansson, Matti Ristinmaa and Mathias Wallin for helping me with some problems of numerical nature which turned up during the work.

Abstract

This masters thesis is done in collaboration between the Division of Solid Mechanics at Lund University and Orthopaedic Department at Lund University Hospital. The thesis is the start of a multi disciplinary research project where the goal is to develop a model for adaptive bone remodelling and.

The main task of the thesis is to investigate existing bone remodelling theories found in the literature. Extra effort has been put down to understand the medical issues involved in the bone remodelling process. The remodelling theories was investigated by use of the finite element method which provides a sufficient solution to the field problems involved.

The first model investigated was a model developed by Huiskes in 1987. This model has a very loose connection to the actual medical issues guiding the remodelling process, instead it is based on an assumption of constant strain energy density distribution in the structure. A change of numerical solution technique from an Euler forward method, proposed by Huiskes, to a Newton algorithm was done in this investigation to ensure force equilibrium. Still the results obtained was similar to the one given by Huiskes.

The second model investigated is a prestudy of a model with a more rigid connection to the medical and physiological effects involved in bone remodelling process. The theory is assumed to explain the contradiction that bone remodelling occurs at much lower tissue level strains then is needed to cause bone signaling in deformed cell cultures. The remodelling theory involves the movement of bone fluid in bone and the fluid flow is assumed to induce strains large enough to trigger the cells. A compleat model has not been developed, due to the limited amount of time given, so the magnitudes of the cell strains induced by the fluid flow has not been evaluated. The effect that has been investigated is how the different loads influence the magnitude and pattern of the fluid flow. It was found that there is a linear relationship between the magnitude of the fluid flow and the frequency of the load applied.

Suggestions and guidelines for future work within the research project is also given, since this thesis has only coped with a fraction of all aspects involved in adaptive bone remodelling.

Contents

Chapter 1

Introduction

1.1 Background to the assignment

Bone is made up of living material which posses the ability to change properties when exposed to mechanical loading. Even thought this phenomena has been known for centuries the actual mechanism controlling the bone remodelling proses is poorly understood. Due to this fact a multidisciplinary research project involving both orthopedics and solid mechanics has started at the University of Lund. The aim for this project is to gain better understanding for the bone remodelling process itself and evidently find a treatment for osteoporosis which is a growing problem worldwide.

1.2 Objective and restrictions

The main objective of this master thesis is to investigate and evaluate existing bone adaptive remodelling models found in the literature. The objective can be broken down in the following

- study of medical literature to understand the medical and physiological issues involved in the bone remodelling process.
- study of biomechanical literature to find suitable bone remodelling models.
- implementation and evaluation some of the found bone remodelling models.

Due to the size and complexity of the problem it is impossible to investigate all the aspect of bone remodelling during the short time period given. In this thesis only mechanical effects will be considered i.e. the influence of chemical or genetic processes is not accounted.

1.3 About this thesis

The thesis can be divided into three sections. The first section consists of chapter 2 to chapter 4 where the medical and physiological aspects of bone remodelling will be described. Also in this section the most interesting bone remodelling aspects will be pointed out. In chapter 5 to chapter 8 the issues of the numerical simulation will be described and in the last section, chapter 9, a discussion of the results from simulations is given. Suggestions of future work will also be given since this thesis only coped with a fraction of all aspects involved in adaptive bone remodelling.

Chapter 2

Anatomical description of bone structure

In this chapter the structure of bone will be described. This will be done in a hierarchical manner to improve understanding and to introduce some commonly used terms. For more information within this area see Bagge [1], Taber [14] or Sonesson [13]

2.1 Long bones

Only long bones will be examined in the thesis, so a short presentation of this type of bones is needed.

Long bones are composed of a hollow shaft called *diaphysis* with an extension, *epiphysis*, at the ends, see fig. 2.1. The inner and outer surfaces of the diaphysis are called *endosteum* and *periosteum*, respectively. The cavity in the diaphysis is called *medullary cavity* and contains the bone marrow. The ends of the long bones are covered with *cartilage* which is a very specialized type of material with extreme low frictional coefficient ($\mu \approx 0.002$, Bagge [1]). The low friction ensures that the joint movements run easily.

2.2 Macroscopic level

On the macroscopic level, which corresponds to what can be seen with the naked eye, there are two types of bone, *cortical* and *cancellous* bone. The latter one is also referred to as *trabecular* bone. Cortical bone is the compact type of bone that primarily can be found in the diaphysis of the long bones. Cancellous bone is a more poroses type of bone made up of a lattice of rods and plates which are referred to as *trabeculas*, see fig 2.2. This type of bone, also known as *spongy bone*, makes up the epiphysis of the long bones.

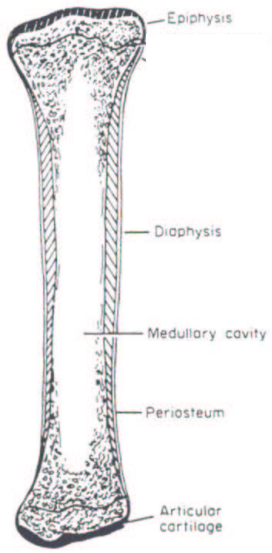


Figure 2.1: Cross-section of a long bone. (taken from Taber [14])

2.3 Microscopic level

Microscopic there are three types of cortical bone, *wowen bone*, *lamellar bone* and *Haversian bone*. Wowen bone is randomly arranged and can be seen in very young bone and in large fracture sites. Lamellar bone is, as the name implies, a more arranged type of bone. It is made up by a system of concentric lamina separated by vascular network. Some of the lamellar bone eventually converts to Haversian bone, which consists of cylindrical elements called *osteons* or Haversian systems. The Haversian systems have a radii of $100 - 150\mu m$ and are roughly aligned with the axial direction of the bone.

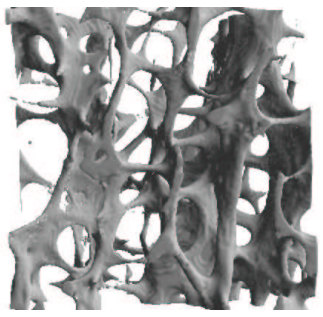


Figure 2.2: Cancellous bone structure showing a network of rods and plates.(taken from the web-site http://www.nd.edu/ gniebut/research.html)

In the center of each Haversian system a *Haversian canal* is located. These canals have a radii of $25\mu m$ and contains blood vessels, nerves and bone fluid. The connection of some Haversian systems is done by *Volkmann canals*. The Volkmann canals are of the same dimension as the Haversian canals ($10\mu m$) and these canals do also contain blood vessels, nerves and bone fluid. These two type of canals makes up a porosity level of bone which is referred to as the *vascular porosity*.

On the wall of each Haversian canal, the entrances to the *canaliculi* can be found. The canaliculi are passageways (radii in the order of $0.1\mu m$) that run from the Haversian canal to small cavities, *lacunae*, or between lacunae. This porosity level is known as the *lacuna-canalicular porosity*. The spreading of the canaliculi in restricted by the *cementline*. The cementline is relative impermeable for the bone fluid and makes up the circumference of each osteon. A sketch of a systems of osteons can be seen in 2.3 (Volkmann canals are not visible in this sketch).

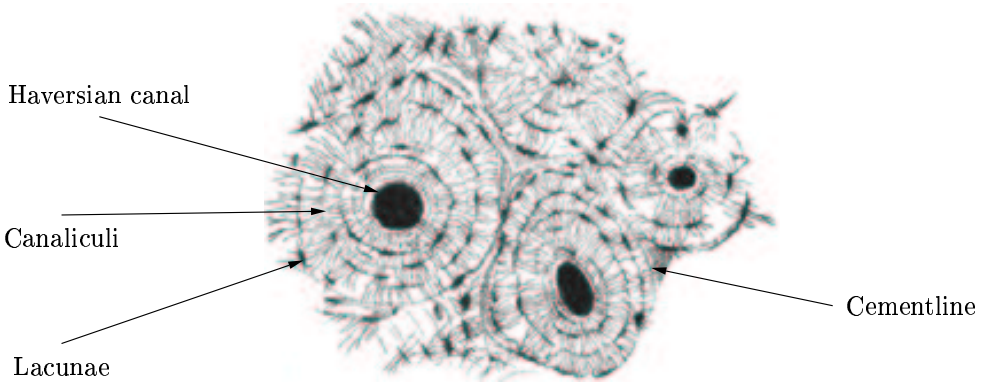


Figure 2.3: Cross-section of cortical bone showing a system of osteons (taken form the web-site http://www.bartelby.com)

2.4 Bone cells and osteoporosis

Bone consist primarily of thee types of cells:

- osteoblasts: a cell that create (deposit) bone.
- osteoclasts: a cell that destroy (resorb) bone.
- osteocytes: a cell that takes care of the maintenance of the bone

Growth of bone do not involve cell division, instead first the osteoclasts resorb bone and then the osteoblasts fill in the gap with new bone. The new bone is initially soft but hardens gradually due to mineralization. By involving both osteoclasts and osteoblasts it is assumed that new bone can be lied down without creating residuals stresses, [14].

The activity of the two processes involved in bone growth changes with age. For the first 25-30 years, the deposition of bone is greater the the resorption leading to steadily increase of bone mass. From here on, the bone resorption process will be more active leading to a slow decomposition of the bone. The resorption process is extremely active for women during there menopause, this due to an almost complete stop of the production of the female hormone estrogen. The process of growing older described here, is a completely normal process and it will take place in all human life. Although, if the resorption of bone becomes to severe the strength of the skeleton will be so reduced. Complications during normal life will then occur often in terms of fractures in the hip spine or wrist. This phenomena is known as *osteoporosis*. The treatment of extreme osteoporosis consists today of bone resorption decreasing medication such as estrogen (for women) and bisphosphonates (for both men and women). For less extreme cases increased physical activity is the most effective treatment.

Chapter 3

Mechanical properties of bone

In this chapter, the material properties of bone will be described.

3.1 Cortical and cancellous bone

The material properties of both cortical and cancellous bone have been measured by using different techniques such as machine testing or ultrasound. The specimen used for measurement of mechanical properties are often taken from the femur, the tibia, the humerus or the radius. This because both cortical and cancellous bone are present in larger blocks here then in other parts in the skeleton. Some example of mechanical properties for cortical bone can be seen in Table 3.1

The same techniques have been used to measure the mechanical properties for cancellous bone, some examples can be seen in Table 3.2.

As can be seen in Tables 3.1 and 3.2, there is a grate diversity for the material properties obtained by the various investigators. Besides the variation obtained from the use of different measuring techniques, the material properties show dependency of moisture level and anatomical position.

3.2 Material properties used in this thesis

In this thesis the bone material is assumed to be isotropic linear elastic. Furthermore, it will be assumed that both cortical and cancellous bone have the same material properties. Evens if these assumptions do not perfectly agree with the real behavior of human bone, they will be assumed to do as a sufficient approximations.

	Measuring technique	Material symmetry	Properties
Reilly and Burstein (1975) long bones, freshly frozen	machine testing	transversely isotropic	$E_1 = E_2 = 11.5\ GPa$ $E_3 = 17.0\ GPa$ $G_{13} = G_{23} = 3.28\ GPa$ $\nu_{31} = \nu_{32} = 0.46$ $\nu_{21} = 0.58$
Ashman et al. (1984) femur, freshly frozen	ultrasound	orthotropic	$E_1 = 12.0\ GPa$ $E_2 = 13.4\ GPa$ $E_3 = 20.0\ GPa$ $G_{12} = 4.53\ GPa$ $G_{13} = 5.61\ GPa$ $G_{23} = 6.23\ GPa$ $\nu_{12} = 0.376$ $\nu_{13} = 0.222$ $\nu_{23} = 0.235$
Choi et al. (1991) tibia, freshly frozen	microstructural level machine testing	isotropic	$E = 5.44\ GPa$

Table 3.1: Some measured properties for cortical bone. The coordinate system is oriented with the first axis in the radial direction, the second in circumferential direction and the third in the axial direction of a long bone.

	Measuring technique	Material symmetry	Properties
Choi et al. (1990) tibia, freshly frozen	microstructural level machine testing	isotropic	$E = 4.59\ GPa$
Linde et al. (1991) proximal tibia, frozen	machine testing	isotropic	$E = 0.77\ GPa$

Table 3.2: Some measured properties for cancellous bone.

Chapter 4

Bone adaptation and stimulus

In this chapter the behavior of bone adaptation will be discussed. Further the actual mechanism that triggers the bone remodelling, i.e. the stimulus, will be investigated.

4.1 Bone adaptation

Bone is considered as a self-organizing material which adapts its structure to its function. This can for example be seen in long bones, e.g. the femur, whose main task is to transfer the body weight from one end of the bone to the other. The relative flexible ends of the long bones act like shock absorbers when the bone is exposed to impact forces. The trabeculas in each end of long bones are oriented in a way that optimizes the load carrying capacity, the phenomena is known as Wolff's law (Wolff [17]). The hollow shaft effectively support bending stresses and at the same time it is a structure that minimize weight.

This self-organizing ability also implies that when the loading conditions are changed the bone will adapt to this new state. It is a well known fact that increased activity, such as military training, leads to bone deposition. On the other hand inactivity, such as bed resting, leads to bone resorption, sometimes as much as 200 mg a day (Sonesson [13]).

Many experimental studies has been done in this area mainly using animal testing. When doing these types of studies it is of grate importance that the loading environment is controlled and the properties of the bones are measured with satisfying accuracy. Moreover the experiment has to be done in such a way that statistic accuracy is ensured.

One experimental discovery of grate importance for this thesis was made by Lanyon et al.[8] and more recently by Rubin [11]. Both these experiments show that the bone remodelling process does not only depend of the magnitude of the loading but also of the frequency of the applied load. This can be of high interest when designing a rehabilitation program for osteoporosis.

4.2 Stimulus

Even thought the phenomena of bone remodelling has been known for centuries the actual signal triggering the bone remodelling process is poorly understood. Several mechanisms have been proposed for the transduction of mechanical loads to the bone remodelling response, Treharne [15], such as:

- direct stresses or strains on bone cells
- mechanical micro damage
- piezoelectric or streaming potentials
- alteration in mineral solvability due to stresses

Experimental data can be found to support all of these mechanisms so this problem is by no means solved.

4.3 Stimuli investigated in this thesis

Two types of stimuli will be investigated in this thesis. The first one is based on the assumption that the bone structure adapts to the loading situation i.e. resorb bone in areas where it is not needed and deposit bone in more affected areas. This assumption of stimulus will be investigated by use of a plane strain finite element simulation.

The second type of stimulus investigated in this thesis is based on the assumption that direct strains on the bone cell can be acting as stimuli. Thought, an extension of this hypothesis is needed to explain the experimental result found by Rubin and Lanyon [12] and Fritton et al. [5]. Their experiments show that the strains applied to the whole bone (i.e. tissue level strains) is many times lower that the strains needed to cause bone signaling in deformed cell cultures. The extension made involves the bone fluid in the Haversian canal and in the canaliculi-lacuna system. It has been hypothised that the flow of the bone fluid in the canaliculi-lacuna system will induce strains large enough to trigger cell reaction in the cells on the bone surface. This process will be investigated using a finite element simulation.

Chapter 5

Basic continuum mechanics and numerical treatment

In this chapter the basics of continuum mechanics, finite element formulation, constitutive relation and numerical solution technique will be described. For more information within these areas, see Ottosen and Ristinmaa [9] and Ottosen and Ristinmaa [10]

5.1 Definitions of strains and stresses

When dealing with deformable bodies it is vital to establish a quantity that only describes the deformation of the body, i.e. the rigid body motion is excluded.

Consider a body before deformation i.e. a reference configuration. In this body a material point with the position x_i can be found. The body is now deformed into a current configuration and the same material point can then be found at the position x_i^* given by:

$$x_i^* = x_i + u_i \tag{5.1}$$

where $u_i = u_i(x_i, t)$ is called the displacement vector and t is the time.

Now, consider two material points, P and Q, located infinitely close to each other in the undeformed body. After deformation the same two material points has moved to the position P^* and Q^* respectively, see Fig. 5.1. From (5.1) it can be concluded that the vector from P^* to Q^* is given by (for fix t):

$$dx_i^* = dx_i + \frac{\partial u_i}{\partial x_j} dx_j \tag{5.2}$$

where $\frac{\partial u_i}{\partial x_j}$ is called the deformation gradient. (5.2) can then be written

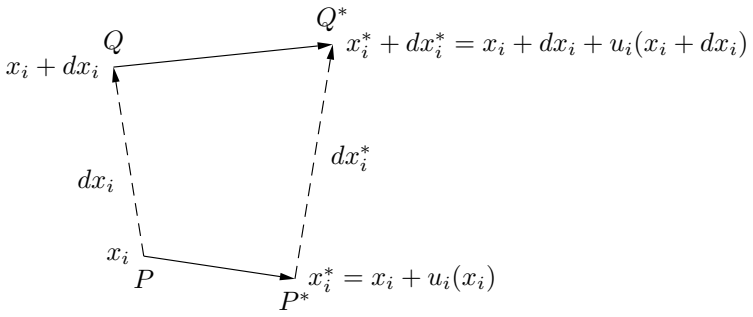


Figure 5.1: Displacements of the material points P and Q

$$dx_i^* = \left(\delta_{ij} + \frac{\partial u_i}{\partial x_j}\right) dx_j \tag{5.3}$$

where δ_{ij} is Cronecer's delta. Now, let the distance between P and Q be denoted by ds and the distance between P^* and Q^* by ds^*. From fig. 5.1 the following relations are concluded

$$ds^2 = dx_j dx_j; \quad (ds^*)^2 = dx_k^* dx_k^* \tag{5.4}$$

It is evident that the deformation of the body is related to change of distance. The change of distance between the two material points discussed earlier may, together with (5.3), be described as:

$$(ds^*)^2 - ds^2 = \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{\partial u_k}{\partial x_i}\frac{\partial u_k}{\partial x_j}\right) dx_i dx_j \tag{5.5}$$

It is possible to write (5.5) as

$$(ds^*)^2 - ds^2 = 2dx_i E_{ij} dx_j \tag{5.6}$$

where

$$E_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \frac{\partial u_k}{\partial x_i}\frac{\partial u_k}{\partial x_j}\right) \tag{5.7}$$

is called the Green strain tensor. Here only small deformations will be considered i.e.

$$|\frac{\partial u_i}{\partial x_j}| \ll 1 \tag{5.8}$$

Under these circumstances, (5.7) can be approximated to

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{5.9}$$

where ε_{ij} is called the small strain tenor. Note that ε_{ij} is a symmetric tensor.

As stress measure the "true" stress or the Cauchy stress will be used. Cauchy's relation i given by:

$$t_i = \sigma_{ij} n_j \tag{5.10}$$

where t_i is the traction force acting on a surface with the normal n_j pointing outward form the body, see fig 5.2, and σ_{ij} is a symmetric tensor called the Cauchy stress tensor.

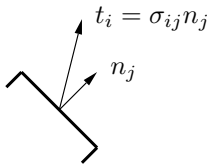


Figure 5.2: Traction force acting on a surface

5.2 Balance of linear momentum and principal of virtual work

The balance of linear momentum for a body with the volume V, the surface area S and the outward pointing normal n_j (see fig 5.3) can for a static situation be written as:

$$\int_S t_i dS + \int_V \rho b_i dV = 0 \tag{5.11}$$

where t_i is the traction force discussed earlier, ρ is the density of the medium and b_i represent the body forces per unit mass. By use of the divergens theorem together with (5.11) and (5.10), the following equation is obtained

$$\int_V \left(\frac{\partial \sigma_{ij}}{\partial x_j} + \rho b_i\right) dV = 0 \tag{5.12}$$

Or, since this relation holds for an arbitrary volume:

$$\frac{\partial \sigma_{ij}}{\partial x_j} + \rho b_i = 0 \tag{5.13}$$

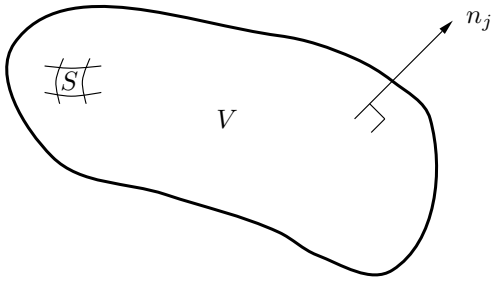


Figure 5.3: Body with the volume V, surface S and the normal n_j

This equation is often called the strong form of equilibrium.

Now, if (5.13) is multiplied by a some arbitrary weight function, v_i and integrated over the volume the following equation is obtained:

$$\int_V v_i \left(\frac{\partial \sigma_{ij}}{\partial x_j} + \rho v_i b_i \right) dV = 0 \tag{5.14}$$

or

$$\int_V \left(\frac{\partial (v_i \sigma_{ij})}{\partial x_j} - \frac{\partial v_i}{\partial x_j} \sigma_{ij} + \rho v_i b_i \right) dV = 0 \tag{5.15}$$

From the divergence theorem and (5.10) together with (5.15) the following equation is obtained:

$$\int_V \frac{\partial v_i}{\partial x_j} \sigma_{ij} dV = \int_S v_i t_i dS + \int_V \rho v_i b_i dV \tag{5.16}$$

Since σ_{ij} is a symmetric tensor, only the symmetric part of $\frac{\partial v_i}{\partial x_j}$ will contribute to the left hand term in (5.16). The symmetric part of $\frac{\partial v_i}{\partial x_j}$ is written as:

$$sym \left(\frac{\partial v_i}{\partial x_j} \right) = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \tag{5.17}$$

Then, by comparing this equation with (5.9) the following quantity can be defined:

$$\varepsilon_{ij}^v = sym \left(\frac{\partial v_i}{\partial x_j} \right) \tag{5.18}$$

Introducing (5.18) into (5.16) the following relation is obtained:

$$\int_V \varepsilon_{ij}^v \sigma_{ij} dV = \int_S v_i t_i dS + \int_V \rho v_i b_i dV \tag{5.19}$$

This equation is known as the weak form of equilibrium or the principal of virtual work, and will later be used to derive the finite element formulation.

5.3 Finite element formulation

To obtain a useful finite element formulation, matrix notation turns out to be convenient because this format is more suitable for numerical implementation.

If matrix formulation is adopted the equation of virtual work (5.19) can be written as

$$\int_V (\boldsymbol{\epsilon}^{\boldsymbol{v}})^T \boldsymbol{\sigma} dV = \int_S \boldsymbol{v}^T \boldsymbol{t} dS + \int_V \rho \boldsymbol{v}^T \boldsymbol{b} dV \tag{5.20}$$

where

$$\boldsymbol{\epsilon}^v = \begin{bmatrix} \varepsilon_{11}^v \\ \varepsilon_{22}^v \\ \varepsilon_{33}^v \\ 2\varepsilon_{12}^v \\ 2\varepsilon_{13}^v \\ 2\varepsilon_{13}^v \end{bmatrix}; \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix}; \quad \boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}; \quad \boldsymbol{t} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}; \quad \boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{5.21}$$

The, perhaps, funny looking form of $\boldsymbol{\epsilon}^v$ will be explained in the next section.

The next step in the finite element formulation is to introduce an approximation for the displacement vector $\boldsymbol{u}$ for the body. The approximation is given by:

$$\boldsymbol{u} = \boldsymbol{N}\boldsymbol{a} \tag{5.22}$$

where $\boldsymbol{N}$ is the *global shape functions* and $\boldsymbol{a}$ is a vector containing the *nodal* displacements of the body. Since the shape functions only depend on the position and the nodal displacement vector only depends on the time, it can be concluded that

$$\boldsymbol{\epsilon} = \boldsymbol{B}\boldsymbol{a} \tag{5.23}$$

where the matrix $\boldsymbol{B}$ is derived from the matrix $\boldsymbol{N}$ according to (5.9).

Now, to obtain the final expression for the finite element formulation, the arbitrary vector $\boldsymbol{v}$ must be investigated. It is possible to approximate this vector in the same way as the displacement vector i.e.

$$\boldsymbol{v} = \boldsymbol{N}\boldsymbol{c} \tag{5.24}$$

where $\boldsymbol{N}$ is the same shape functions describe earlier and $\boldsymbol{c}$ is an arbitrary vector. This is known as Galerkin's method. Since $\boldsymbol{c}$ is arbitrary it is possible to define a quantity similar to $\boldsymbol{\epsilon}$ according to:

$$\boldsymbol{\epsilon}^v = \boldsymbol{B}\boldsymbol{c} \tag{5.25}$$

If (5.24), (5.25), (5.22) and (5.23) are introduced into (5.20) the following equation arise:

$$\boldsymbol{c}^T \left[\int_V (\boldsymbol{B})^T \boldsymbol{\sigma} dV - \int_S (\boldsymbol{N})^T \boldsymbol{t} dS - \int_V \rho (\boldsymbol{N})^T \boldsymbol{b} dV \right] = 0 \tag{5.26}$$

Since this equation must hold for an arbitrary $\boldsymbol{c}$ vector it is concluded that

$$\int_V (\boldsymbol{B})^T \boldsymbol{\sigma} dV = \boldsymbol{f} \tag{5.27}$$

where $\boldsymbol{f}$ is the external forces according to:

$$\boldsymbol{f} = \int_S (\boldsymbol{N})^T \boldsymbol{t} dS + \int_V \rho (\boldsymbol{N})^T \boldsymbol{b} dV \tag{5.28}$$

Since the FE-formulation is derived using only the balance of linear momentum, (5.27) holds for every constitutive relation.

5.4 Constitutive relation

It is now time to investigate the constitutive relation i.e. the relation between the stresses and the strains. The constitutive relation that will be used in this thesis is called Hook's law and it can be written as

$$\sigma_{ij} = D_{ijkl}\varepsilon_{kl} \tag{5.29}$$

where D_{ijkl} is a fourth orders tensor. Since both σ_{ij} and ε_{kl} are symmetric quantities it is evident that

$$D_{ijkl} = D_{jikl}; \quad D_{ijkl} = D_{ijlk} \tag{5.30}$$

It was discussed earlier that matrix formulation is convenient for numerical implementation, it is therefore vital to establish the constitutive relation in this fashion. By expanding equation (5.29) by its dummy index and using (5.30) the following relation is obtained

$$[\sigma_{ij}] = \begin{bmatrix} D_{ij11} & D_{ij22} & D_{ij33} & D_{ij12} & D_{ij13} & D_{ij23} \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{13} \end{bmatrix} \tag{5.31}$$

By choosing proper values for the indexes i and j the constitutive relation in matrix form is given by

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\epsilon} \tag{5.32}$$

where

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix}; \quad \boldsymbol{D} = \begin{bmatrix} D_{1111} & D_{1122} & D_{1133} & D_{1112} & D_{1113} & D_{1123} \\ D_{2211} & D_{2222} & D_{2233} & D_{2212} & D_{2213} & D_{2223} \\ D_{3311} & D_{3322} & D_{3333} & D_{3312} & D_{3313} & D_{3323} \\ D_{1211} & D_{1222} & D_{1233} & D_{1212} & D_{1213} & D_{1223} \\ D_{1311} & D_{1322} & D_{1333} & D_{1312} & D_{1313} & D_{1323} \\ D_{2311} & D_{2322} & D_{2333} & D_{2312} & D_{2313} & D_{2323} \end{bmatrix} \quad \boldsymbol{\epsilon} = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{bmatrix}$$

5.5 General format of Newton's method

The equation system that arise from the finite element formulation is usually nonlinear and, which will be shown later, can involve many parameters. When solving these type of problems the usual approach is some kind of iterative method such as Newton's method.

For a one-dimensional problem the idea of Newton's method can be seen in fig 5.4. The issue is to find the solution to the nonlinear equation $f(x) = 0$. This is done by guessing an initial value of x, i.e. x^0, and then identify the point $f(x^0)$. For this point the tangent is determined and extrapolated to obtain the next estimation if the solution x^1. This procedure is then repeated until satisfying accuracy of the solution is obtained.

Consider now the following equation, that for the wanted state fulfills

$$\begin{cases} \boldsymbol{\psi}_1(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) &= \boldsymbol{0} \\ \quad \vdots & \quad \vdots \\ \boldsymbol{\psi}_n(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) &= \boldsymbol{0} \end{cases} \tag{5.33}$$

where $(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n)$ are all the involved parameters. Assume, in accordance with the one-dimensional case, that the state $(\boldsymbol{a}_1 + d\boldsymbol{a}_1, \dots, \boldsymbol{a}_n + d\boldsymbol{a}_n)$ satisfies (5.33). Here $(d\boldsymbol{a}_1, \dots, d\boldsymbol{a}_n)$ denotes small increments, or *corrections*, for each parameter. By using a Taylor expansion and ignoring higher order terms the following equation is obtained

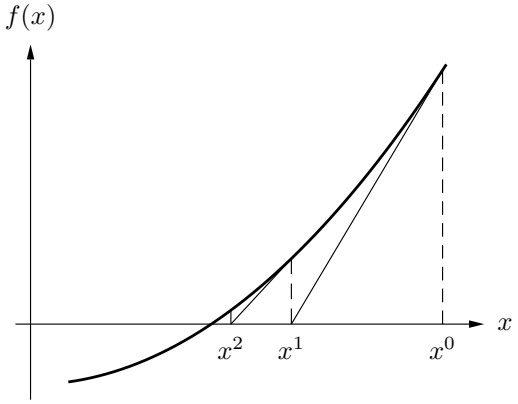


Figure 5.4: Newton's method for a one-dimensional problem

$$\begin{cases} \boldsymbol{\psi}_1(\boldsymbol{a}_1 + d\boldsymbol{a}_1 \dots \boldsymbol{a}_n + d\boldsymbol{a}_n) & = \boldsymbol{\psi}_1(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) + d\boldsymbol{\psi}_1(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) \\ \vdots & \vdots \\ \boldsymbol{\psi}_n(\boldsymbol{a}_1 + d\boldsymbol{a}_1 \dots \boldsymbol{a}_n + d\boldsymbol{a}_n) & = \boldsymbol{\psi}_n(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) + d\boldsymbol{\psi}_n(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) \end{cases} \tag{5.34}$$

or, since the state $(\boldsymbol{a}_1 + d\boldsymbol{a}_i, \dots, \boldsymbol{a}_n + d\boldsymbol{a}_n)$ satisfies (5.33)

$$\begin{cases} \boldsymbol{0} & = \boldsymbol{\psi}_1(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) + d\boldsymbol{\psi}_1(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) \\ \vdots & \vdots \\ \boldsymbol{0} & = \boldsymbol{\psi}_n(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) + d\boldsymbol{\psi}_n(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) \end{cases} \tag{5.35}$$

Here the symbol d denotes differentiation. Furthermore may the term $d\boldsymbol{\psi}_i(\boldsymbol{a}_1 \dots \boldsymbol{a}_n)$ be evaluated as

$$d\boldsymbol{\psi}_i(\boldsymbol{a}_1 \dots \boldsymbol{a}_n) = \sum_{j=1}^{n} \frac{\partial \boldsymbol{\psi}_i(\boldsymbol{a}_1 \dots \boldsymbol{a}_n)}{\partial \boldsymbol{a}_j} d\boldsymbol{a}_j \tag{5.36}$$

By using (5.36) into (5.35) the following relation is obtained

$$-\boldsymbol{\Psi} = \boldsymbol{K}_t d\boldsymbol{A} \tag{5.37}$$

where

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\psi}_1(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n) \\ \vdots \\ \boldsymbol{\psi}_n(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n) \end{bmatrix}; \quad \boldsymbol{K}_t = \begin{bmatrix} \frac{\partial \boldsymbol{\psi}_1(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n)}{\partial \boldsymbol{a}_1} & \dots & \frac{\partial \boldsymbol{\psi}_1(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n)}{\partial \boldsymbol{a}_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{\psi}_n(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n)}{\partial \boldsymbol{a}_1} & \dots & \frac{\partial \boldsymbol{\psi}_n(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n)}{\partial \boldsymbol{a}_n} \end{bmatrix}; \quad d\boldsymbol{A} = \begin{bmatrix} d\boldsymbol{a}_1 \\ \vdots \\ d\boldsymbol{a}_n \end{bmatrix}$$

$\boldsymbol{K}_t$ is usually called the *tangential stiffness matrix*. From (5.37) it is possible to calculate the corrections for each parameter i.e. $d\boldsymbol{A}$, sins all other quantities are known. It is furthermore possible to update each parameter according to

$$\boldsymbol{A}_{i+1} = \boldsymbol{A}_i + d\boldsymbol{A} \tag{5.38}$$

where the subscript i denotes the increment index and the vector $\boldsymbol{A}$ contains the values for each parameter.

Chapter 6

Bone adaptation simulation by use of an updating algorithm for Young's modulus

This bone remodeling simulation was originally done by Huiskes [7]. The simulation is done by using a plane strain finite element model in combination with an updating algorithm for Young's modulus. Here the finite element problem is solved by a Newton algorithm instead of an Euler forward method which was done originally. To do so, the general format of Newton's method discussed is section 5.4 is further investigated. In this simulation an expression to update the Young's modulus within each iteration has to be derived. Further, when using a Newton algorithm, the tangential stiffness in each iteration must be found to ensure good convergence. The model is implemented in Matlab together with the Calfem toolbox (the code is listed in appendix).

6.1 Newton's method

In this simulation, Newton's method is based on the equilibrium (5.27) so it can be concluded:

$$\boldsymbol{\psi}(\boldsymbol{a}) = \int_V (\boldsymbol{B})^T \boldsymbol{\sigma} dV - \boldsymbol{f} \tag{6.1}$$

where $\boldsymbol{\psi}(\boldsymbol{a})$ represents the out of balance forces. For equilibrium it it evident that

$$\boldsymbol{\psi}(\boldsymbol{a}) = \boldsymbol{0} \tag{6.2}$$

In this case the external forces $\boldsymbol{f}$ are known and kept constant. Further more is the stresses only depending on the nodal displacements $\boldsymbol{a}$. Since the matrix $\boldsymbol{B}$ does not depend on the displacement the Taylor expansion, according to (5.35), is given by:

$$\int_V (\boldsymbol{B})^T d\boldsymbol{\sigma} dV = \boldsymbol{f} - \int_V (\boldsymbol{B})^T \boldsymbol{\sigma} dV \tag{6.3}$$

To obtain the final expression the constitutive relation given by (5.32). Differentiation of this equation yields

$$d\boldsymbol{\sigma} = \boldsymbol{D}_t d\boldsymbol{\epsilon} \tag{6.4}$$

where $\boldsymbol{D}_t$ is the *tangential stiffness matrix*. This quantity will be discussed later in this chapter. The term $d\boldsymbol{\epsilon}$ is given by differentiation of (5.23) according to

$$d\boldsymbol{\epsilon} = \boldsymbol{B} d\boldsymbol{a} \tag{6.5}$$

If (6.4) and (6.5) is introduced to (6.3) the final expression for Newton's method is obtained as

$$\boldsymbol{K}_t d\boldsymbol{a} = \boldsymbol{f} - \int_V (\boldsymbol{B})^T \boldsymbol{\sigma} dV \tag{6.6}$$

where

$$\boldsymbol{K}_t = \int_V (\boldsymbol{B})^T \boldsymbol{D}_t \boldsymbol{B} dV \tag{6.7}$$

is the tangential stiffness matrix. The state for the next iteration is the given in accordance with (5.38)

6.2 Updating of Young's modulus

The updating algorithm is based on the theory of adaptive elasticity and uses the strain energy density (SED) as feed-back control variable. The SED is defined as:

$$U = \frac{1}{2}\varepsilon_{ij}\sigma_{ij} = \frac{1}{2}\boldsymbol{\epsilon}^T\boldsymbol{\sigma} \tag{6.8}$$

The difference between the actual SED and a site-specific homeostatic equilibrium SED, U_n, is used as stimuli. The rate equation for the Young's Modulus is the defined as:

$$\frac{dE}{dt} = \begin{cases} C_e(U - (1+s)U_n), & U > (1+s)U_n \\ 0, & (1-s) \leq U \leq (1+s)U_n \\ C_e(U - (1-s)U_n), & U < (1-s)U_n \end{cases} \tag{6.9}$$

Here s is half the width of the "lazy" zone and C_e is the slope of the curve. A graphical interpretation of (6.9) is presented in figure 6.1

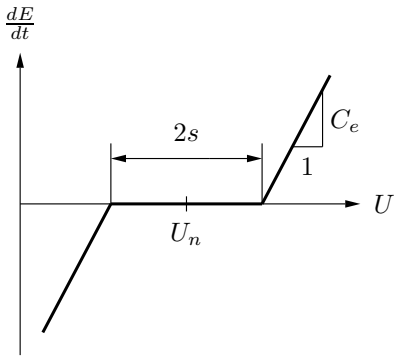


Figure 6.1: Rate equation for Young's modulus

If the width of the lazy zone is assumed to be zero, this in accordance with Huiskes [7], (6.9) can be written as

$$\frac{dE}{dt} = C_e(U - U_n) \tag{6.10}$$

(6.10) is then integrated i.e.

$$\int_{E^n}^{E^n+dE} dE = \int_t^{t+\Delta t} C_e(U - U_n)dt \tag{6.11}$$

The index n means that the quantity is taken from the last state of equilibrium. (6.11) is now approximated by a Euler backward method since it ensures unconditional stability

$$\Delta E = C_e \Delta t(U^{n+1} - U_n) \tag{6.12}$$

where the index $n+1$ refers to the next iteration step.

The next step is to involve the constitutive relation. In this analyze bone is assumed to be isotropic linear elastic i.e.:

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\epsilon} \tag{6.13}$$

where $\boldsymbol{D}$ is defined by

$$\boldsymbol{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \frac{1}{2}(1-2\nu) \end{bmatrix} \tag{6.14}$$

For the next iteration step, (6.13) gives:

$$\boldsymbol{\sigma}^{n+1} = \boldsymbol{D}^{n+1}\boldsymbol{\epsilon}^{n+1} \quad : \boldsymbol{D}^{n+1} = \frac{E^n + \Delta E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \frac{1}{2}(1-2\nu) \end{bmatrix} \tag{6.15}$$

Now, the issue is to derive an scalar expression for the updating term for Young's modulus ΔE This can be done by using some simple algebraic manipulation (here the constant matrix in the expression for $\boldsymbol{D}^{n+1}$ in (6.15) is replaced with $\boldsymbol{G}$ so the equation can be written in a more compact manner):

$$C_e \Delta t \left[\frac{1}{2} (\boldsymbol{\epsilon}^{n+1})^T \boldsymbol{\sigma}^{n+1} - U_n \right] = C_e \Delta t \left[\frac{E^n + \Delta E}{2(1+\nu)(1-2\nu)} (\boldsymbol{\epsilon}^{n+1})^T \boldsymbol{G} \boldsymbol{\epsilon}^{n+1} - U_n \right] \tag{6.16}$$

By using (6.16) together with (6.8) and (6.12) the following relation is derived

$$\Delta E = \frac{c_2}{1 - c_1} \tag{6.17}$$

where:

$$c_1 = \frac{C_e \Delta t}{2(1+\nu)(1-2\nu)} (\boldsymbol{\epsilon}^{n+1})^T \boldsymbol{G} \boldsymbol{\epsilon}^{n+1} \tag{6.18}$$

and

$$c_2 = C_e \Delta t \left(\frac{E^n}{2(1+\nu)(1-2\nu)} (\boldsymbol{\epsilon}^{n+1})^T \boldsymbol{G} \boldsymbol{\epsilon}^{n+1} - U_n \right) \tag{6.19}$$

The expression for the updating term of Young's modulus in (6.17) can now be used in a Newton algorithm.

6.3 Derivation of the algorithmic tangential stiffness

In this section the algorithmic tangential stiffness matrix will be derived. Here tensor notation will be used due to its compact manner.

To start with (6.15) is written on tensor form:

$$\sigma_{ij}^{n+1} = D_{ijkl}^{n+1} \varepsilon_{kl}^{n+1} \tag{6.20}$$

where

$$D_{ijkl}^{n+1} = \frac{E^{n+1}}{1+\nu}\left(\frac{1}{2}(\delta_{ik}\delta_{jl}+\delta_{il}\delta_{jk}) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl}\right) \tag{6.21}$$

Further the (6.20) is differentiated according to:

$$d\sigma_{ij}^{n+1} = dD_{ijkl}^{n+1}\varepsilon_{kl}^{n+1} + D_{ijkl}^{n+1}d\varepsilon_{kl}^{n+1} \tag{6.22}$$

To obtain a useful expression the term dD_{ijkl}^{n+1} needs further investigation. This term can be rewritten as:

$$dD_{ijkl}^{n+1} = d(E^{n+1}A_{ijkl}) = d(\Delta E)A_{ijkl} \tag{6.23}$$

where

$$A_{ijkl} = \frac{1}{1+\nu}\left(\frac{1}{2}(\delta_{ik}\delta_{jl}+\delta_{il}\delta_{jk}) + \frac{\nu}{1-2\nu}\delta_{ij}\delta_{kl}\right)$$

When (6.23) is introduced into (6.12) the following equation is obtained:

$$dD_{ijkl}^{n+1} = C_e\Delta t\left(\frac{\partial U^{n+1}}{\partial\sigma_{mn}^{n+1}}d\sigma_{mn}^{n+1} + \frac{\partial U^{n+1}}{\partial\varepsilon_{mn}^{n+1}}d\varepsilon_{mn}^{n+1}\right)A_{ijkl} \tag{6.24}$$

(6.24) is inserted into (6.22) and the result is

$$d\sigma_{op}^{n+1}\left(I_{ijop} - C_e\Delta tA_{ijkl}\varepsilon_{kl}^{n+1}\frac{\partial U^{n+1}}{\partial\sigma_{op}^{n+1}}\right) = \left(C_e\Delta tA_{ijkl}\varepsilon_{kl}^{n+1}\frac{U^{n+1}}{\partial\varepsilon_{qr}^{n+1}} + D_{ijqr}^{n+1}\right)d\varepsilon_{qr}^{n+1} \tag{6.25}$$

Now the only thing remaining is to invert the forth orders tensor on the left hand side of (6.25). Combining this with (6.8) the following equation is obtained:

$$d\sigma_{ij}^{n+1} = (D_{ijqr}^{n+1})^{ATS}d\varepsilon_{qr} \tag{6.26}$$

where the *algorithmic tangential stiffness tensor*, $\left(D_{ijkl}^{n+1}\right)^{ATS}$, is introduced as:

$$(D_{ijqr}^{n+1})^{ATS} = \frac{1}{2}C_e\Delta tA_{ijkl}\varepsilon_{kl}^{n+1}\sigma_{qr}^{n+1} + D_{ijqr}^{n+1} + \frac{\frac{1}{2}C_e\Delta tA_{ijmn}\varepsilon_{mn}^{n+1}\varepsilon_{op}^{n+1}(\frac{1}{2}C_e\Delta tA_{opkl}\varepsilon_{kl}^{n+1}\sigma_{qr}^{n+1} + D_{opqr}^{n+1})}{1 - \frac{1}{2}\varepsilon_{uv}^{n+1}C_e\Delta tA_{uvst}\varepsilon_{st}^{n+1}}$$

The expression for the arithmetic tangential stiffness is then expanded, similar to (5.31), to obtain a matrix formulation suitable for finite element implementation.

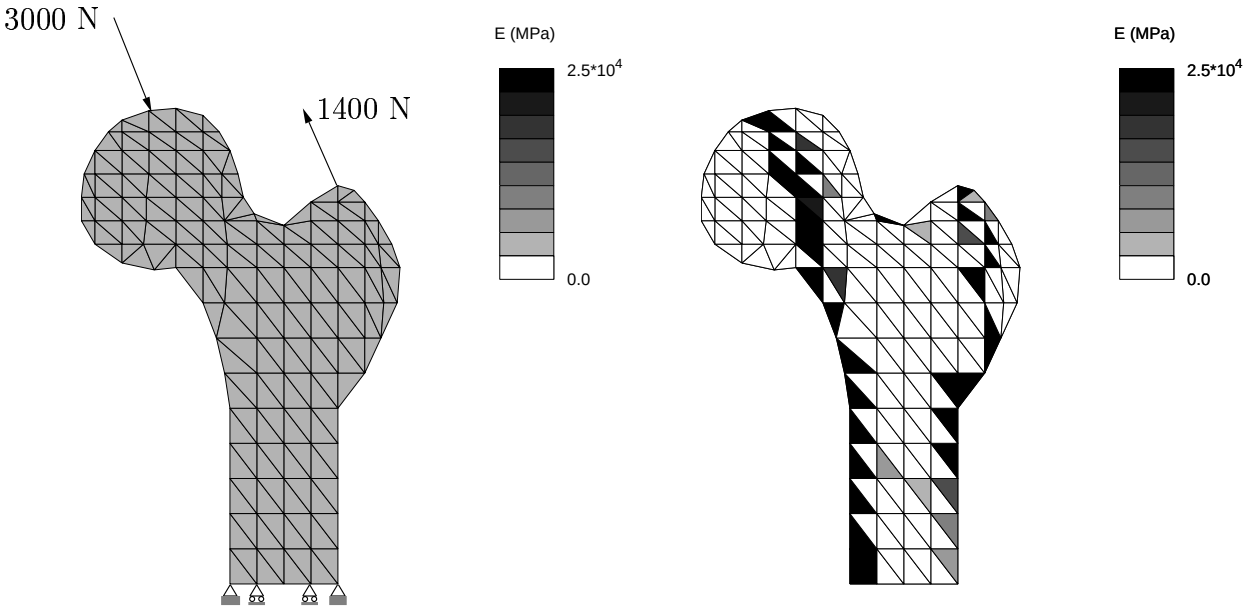
6.4 Bone-density prediction

The updating algorithm described above can be used to predict bone-density in a proximal femur. The initial configuration is a proximal femur with uniform Young's Modulus($E = 0.5 \cdot 10^4$MPa) which is load on the femoral head and by the abductor muscles, see Fig.6.2. The constant homeostatic SED, U_n, is calculated in this configuration according to:

$$U_n = \frac{1}{nelm} \sum_{i=1}^{nelm} U_i \tag{6.27}$$

where $nelm$ is the number of element used in the finite element model.

The updating algorithm is now used and Young's modulus is updated within every element in each time increment. Constraints Young's modulus must be specified so no unphysiological values(higher than $2.5 \cdot 10^4$MPa or lower than zero MPa) is obtained. The final stat of the proximal femur can be seen in Fig. 6.2



(a) Initial state of the proximal femur showing load situation and boundary conditions

(b) Final state of proximal femur

Figure 6.2: Initial and final state of proximal femur

When the variation of Young's Modulus is known, the density distribution can be calculated by the relation(see Huiskes [7])

$$E = a\rho^3 \tag{6.28}$$

where a is a constat.

Discussion of the result is done in chapter 9.

Chapter 7

Finite element formulation and numerical solution technique for a porous medium

In chapter 3 it was stated that the movement of bone fluid is involved in the bone remodelling process. Even thought a commercial finite element code (ABAQUS) will be used, a brief introduction to the theory for a porous media presented. First of all some basic quantities must be defined, then the governing equations and the finite element formulation will be investigated. Finally the numerical solution technique for this type of problem will be discussed. For more information see Hibbit, Karlsson and Sorensen [6]

7.1 General definitions

A porous media is considered as a multiphase material where the elementary volume, dV, is made up of a volume of solid material dV_s and volume of voids, dV_v. The volume of voids is filled with a volume of liquid , dV_w, under the condition $0 < dV_w \leq dV_v$. The liquid if free to move through the material when driven. It is now possible to define the porosity, n, of the medium according to

$$n = \frac{dV_v}{dV} \tag{7.1}$$

or

$$n = 1 - \frac{dV_s}{dV_s^0}\frac{dV^0}{dV}\frac{dV_s^0}{dV^0} \tag{7.2}$$

where the superscript 0 indicates the values in the reference configuration. From (7.1) and (7.2) it is concluded that

$$\frac{1-n}{1-n^0} = \frac{J_s}{J} \tag{7.3}$$

where

$$J = \frac{dV}{dV^0}$$

and

$$J_s = \frac{dV_s}{dV_s^0}$$

The saturation, s is defined as

$$s = \frac{dV_w}{dV_v} \tag{7.4}$$

Finally, the volume ratio of the liquid at a point is defined as

$$n_w = \frac{dV_w}{dV} = sn \tag{7.5}$$

7.2 Force equilibrium

In a solid which contains a fluid it is assumed that both the fluid and the solid can carry load. The total stress σ_{ij} acting in a point can the be written as:

$$\sigma_{ij} = \overline{\sigma}_{ij} - \chi u_w \delta_{ij} \tag{7.6}$$

where $\overline{\sigma}_{ij}$ is the stress in the solid phase or the effective stress , u_w is the pore fluid pressure and χ is a variable depending on the saturation. In ABAQUS the variable χ is assumed to equal the saturation s of the medium.

If (5.18) together with (7.6) is introduced to (5.16) the following equation is obtained:

$$\int_V \varepsilon_{ij}^v \overline{\sigma}_{ij} dV - \int_V \varepsilon_{ij}^v \chi u_w \delta_{ij} dV = \int_S v_i t_i dS + \int_V \rho v_i b_i dV \tag{7.7}$$

or in matrix notation

$$\int_V (\boldsymbol{\epsilon}^{\boldsymbol{v}})^T \overline{\boldsymbol{\sigma}} dV - \int_V (\boldsymbol{\epsilon}^{\boldsymbol{v}})^T \boldsymbol{I} \chi u_w dV = \int_S \boldsymbol{v}^T \boldsymbol{t} dS + \int_V \rho \boldsymbol{v}^T \boldsymbol{b} dV \tag{7.8}$$

where $\boldsymbol{I}$ a matrix defined as:

$$\boldsymbol{I} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

By using (5.24) and (5.25) into equation (7.8) gives

$$\int_V (\boldsymbol{B})^T \overline{\boldsymbol{\sigma}} dV - \int_V (\boldsymbol{B})^T \boldsymbol{I} \chi u_w dV = \int_S (\boldsymbol{N})^T \boldsymbol{t} dS + \int_V \rho (\boldsymbol{N})^T \boldsymbol{b} dV \tag{7.9}$$

If the solid is assumed to be linear elastic the constitutive relation can be described as:

$$\overline{\sigma}_{ij} = \overline{D}_{ijkl} \overline{\varepsilon}_{kl} \tag{7.10}$$

where $\overline{D}_{ijkl}$ is, for this case, the same tensor as in equation 6.21. The term $\overline{\varepsilon}_{kl}$ represent the effective strain and is defined by

$$\overline{\varepsilon}_{kl} = \varepsilon_{kl} + \frac{1}{3} \frac{s u_w}{K_s} \delta_{kl} \tag{7.11}$$

where K_s is the bulk modulus for the solid. (7.11) may also be written in matrix formulation

$$\overline{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon} + \frac{1}{3} \frac{s u_w}{K_s} \boldsymbol{I} \tag{7.12}$$

The constitutive relation in matrix formulation may then be written as

$$\overline{\boldsymbol{\sigma}} = \overline{\boldsymbol{D}} \left(\boldsymbol{\epsilon} + \frac{1}{3} \frac{s u_w}{K_s} \boldsymbol{I} \right) \tag{7.13}$$

If the fluid compressible, the following relation is assumed

$$\frac{\rho_w}{\rho_w^0} \approx 1 + \frac{u_w}{K_w} \tag{7.14}$$

where ρ_w is the density and K_w is the bulk modulus for the fluid.

Furthermore, if the solid is compressible a similar relation is assumed

$$\frac{\rho_s}{\rho_s^0} = \frac{1}{J_s} \approx 1 + \frac{1}{K_s}\left(su_w + \frac{\overline{p}}{1-n}\right) \tag{7.15}$$

where the mean effective pressure, $\overline{p}$, is defines as

$$\overline{p} = -\frac{1}{3}\overline{\sigma}_{kk} \tag{7.16}$$

or, in matrix notation

$$\overline{p} = -\frac{1}{3}\boldsymbol{I}^T\overline{\boldsymbol{\sigma}} \tag{7.17}$$

7.3 Pore fluid flow

This equation is based on mass continuity for the pore fluid. The total mass for the fluid in a control volume is

$$\int_V \rho_w dV_w = \int_V \rho_w n_w dV \tag{7.18}$$

where ρ_w is the density for the fluid. The time rate of change of the mass for the fluid may be written as

$$\frac{d}{dt}\left(\int_V \rho_w n_w dV\right) = \int_V \frac{d}{dt}\left(\rho_w n_w\right) dV \tag{7.19}$$

The time rate of change of mass for the fluid must equal the mass that enters the volume trough its surface. This statement gives rise to the mass continuity equation for the fluid i.e.

$$\int_V \frac{d}{dt}\left(\rho_w n_w\right) dV = -\int_S \rho_w n_w n_i (v_w)_i dS \tag{7.20}$$

where n_i is the normal to the surface pointing out form the body and $(v_w)_i$ is the average velocity of the fluid relative the solid. By using the divergence theorem (7.20) can by written as

$$\int_V \frac{d}{dt}\left(\rho_w n_w\right) + \frac{\partial}{\partial x_i}\left(\rho_w n_w (v_w)_i\right) dV = 0 \tag{7.21}$$

Or, since the volume is arbitrary

$$\frac{d}{dt}\left(\rho_w n_w\right) + \frac{\partial}{\partial x_i}\left(\rho_w n_w (v_w)_i\right) = 0 \tag{7.22}$$

This relation is often called the strong formulation. To obtain the equivalent weak formulation, (7.22) is multiplied by an arbitrary weight function, w, and integrated over the volume i.e.

$$\int_V w\frac{d}{dt}\left(\rho_w n_w\right) dV + \int_V w\frac{\partial}{\partial x_i}\left(\rho_w n_w (v_w)_i\right) dV = 0 \tag{7.23}$$

or, by using the divergence theorem

$$\int_V \left[w\frac{d}{dt}\left(\rho_w n_w\right) - \frac{\partial w}{\partial x_i}\rho n_w (v_w)_i\right] dV + \int_S w\rho_w n_w n_i (v_w)_i dS = 0 \tag{7.24}$$

To deal with the time derivative term (7.24) is approximately time integrated by an Euler backward formula giving

$$\int_V \left[w\left((\rho_w n_w)_{t+\Delta t} - (\rho_w n_w)_t\right) - \Delta t\frac{\partial w}{\partial x_i}\rho_w n_w (v_w)_i\right] dV + \Delta t\int_S w\rho_w n_w n_i (v_w)_i dS = 0 \tag{7.25}$$

or, by adopting matrix formulation

$$\int_V \left[w\left((\rho_w n_w)_{t+\Delta t} - (\rho_w n_w)_t\right) - \Delta t\rho_w n_w \left(\frac{\partial w}{\partial \boldsymbol{x}}\right)^T \boldsymbol{v}_w\right] dV + \Delta t\int_S w\rho_w n_w (\boldsymbol{n})^T \boldsymbol{v_w} dS = 0 \tag{7.26}$$

where

$$\frac{\partial w}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial w}{\partial x_1} \\ \frac{\partial w}{\partial x_2} \\ \frac{\partial w}{\partial x_3} \end{bmatrix}; \quad \boldsymbol{n} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix}; \quad \boldsymbol{v}_w = \begin{bmatrix} (v_w)_1 \\ (v_w)_2 \\ (v_w)_3 \end{bmatrix}$$

It is now possible to introduce an approximation for the weight function w according to

$$w = \overline{\boldsymbol{N}}\overline{\boldsymbol{c}} \tag{7.27}$$

where the vector $\overline{\boldsymbol{N}}$ represents some shape functions (not to be confused with the shape function $\boldsymbol{N}$ discussed in chapter 5) and $\overline{\boldsymbol{c}}$ is an arbitrary vector. Since $\overline{\boldsymbol{c}}$ do not depend on the position it is evident that

$$\frac{\partial w}{\partial \boldsymbol{x}} = \overline{\boldsymbol{B}}\overline{\boldsymbol{c}} \tag{7.28}$$

where

$$\overline{\boldsymbol{B}} = \frac{\partial \overline{\boldsymbol{N}}}{\partial \boldsymbol{x}} \tag{7.29}$$

Introduction of (7.27) and (7.29) into (7.26) then yields

$$(\overline{\boldsymbol{c}})^T \int_V \left[\left(\overline{\boldsymbol{N}}\right)^T \left((\rho_w n_w)_{t+\Delta t} - (\rho_w n_w)_t\right) - \Delta t \rho_w n_w \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{v}_w \right] dV + (\overline{\boldsymbol{c}})^T \Delta t \int_S \left(\overline{\boldsymbol{N}}\right)^T \rho_w n_w (\boldsymbol{n})^T \boldsymbol{v_w} dS = 0 \tag{7.30}$$

Since (7.30) must hold for every vector $\overline{\boldsymbol{c}}$ it is obvious that

$$\int_V \left[\left(\overline{\boldsymbol{N}}\right)^T \left((\rho_w n_w)_{t+\Delta t} - (\rho_w n_w)_t\right) - \Delta t \rho_w n_w \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{v}_w \right] dV + \Delta t \int_S \left(\overline{\boldsymbol{N}}\right)^T \rho_w n_w (\boldsymbol{n})^T \boldsymbol{v_w} dS = \boldsymbol{0} \tag{7.31}$$

It will later turn out to be convenient to normalize (7.31) by the density of the fluid in the reference configuration, ρ_w^0, giving

$$\int_V \left[\left(\overline{\boldsymbol{N}}\right)^T \left(\left(\frac{\rho_w}{\rho_w^0} ns\right)_{t+\Delta t} - \left(\frac{\rho_w}{\rho_w^0} ns\right)_t \right) - \Delta t \frac{\rho_w}{\rho_w^0} ns \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{v}_w \right] dV + \Delta t \int_S \left(\overline{\boldsymbol{N}}\right)^T \frac{\rho_w}{\rho_w^0} ns (\boldsymbol{n})^T \boldsymbol{v_w} dS = \boldsymbol{0} \tag{7.32}$$

Here the term n_w has been replaced by ns in accordance with (7.5)

The constitutive relation for the pore fluid flow is governed by Darcy's law, which states that the fluid flow is proportional to the gradient of the piezometric head i.e.

$$sn(v_w)_i = -\widehat{k}_{ij}\frac{\partial \phi}{\partial x_j} \tag{7.33}$$

where $\widehat{k}_{ij}$ is a tensor describing the permeability of the medium and ϕ is the piezometric head, defined as

$$\phi = x_3 + \frac{u_w}{g\rho_w} \tag{7.34}$$

where x_3 represents the elevation from some reference datum and g is the magnitude of the gravitational acceleration acting in opposite direction to x_3. Since the direction and magnitude of g is constant, (7.33) can be written as

$$sn(v_w)_i = -\widehat{k}_{ij}\frac{1}{g\rho_w}\left(\frac{\partial u_w}{\partial x_j} - \rho_w g_j\right) \tag{7.35}$$

where

$$g_j = -g\frac{\partial z}{\partial x_j}$$

or, by adopting matrix formulation

$$sn\boldsymbol{v}_w = -\widehat{\boldsymbol{k}}\frac{1}{g\rho}\left(\frac{\partial u_w}{\partial \boldsymbol{x}} - \rho_w \boldsymbol{g}\right) \tag{7.36}$$

7.4 Numeric solution technique

In this section the numerical solution technique for a coupled fluid flow/displacement system will be discussed. The solution technique that will be used is Newton's method described in section 5.5 and the system of equations that is to be solved is defined by

$$\begin{cases} \boldsymbol{\psi}_1(\boldsymbol{a}, u_w) &= \boldsymbol{0} \\ \boldsymbol{\psi}_2(\boldsymbol{a}, u_w) &= \boldsymbol{0} \end{cases} \tag{7.37}$$

where

$$\boldsymbol{\psi}_1(\boldsymbol{a}, u_w) = \int_V (\boldsymbol{B})^T \overline{\boldsymbol{\sigma}} dV - \int_V (\boldsymbol{B})^T \boldsymbol{I}\chi u_w dV - \int_S (\boldsymbol{N})^T \boldsymbol{t} dS - \int_V \rho\,(\boldsymbol{N})^T \boldsymbol{b} dV \tag{7.38}$$

and

$$\boldsymbol{\psi}_2(\boldsymbol{a}, u_w) = \int_V \left[\left(\overline{\boldsymbol{N}}\right)^T \left((\frac{\rho_w}{\rho_w^0} ns)_{t+\Delta t} - (\frac{\rho_w}{\rho_w^0} ns)_t \right) - \Delta t \frac{\rho_w}{\rho_w^0} ns \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{v}_w \right] dV + \Delta t \int_S \left(\overline{\boldsymbol{N}}\right)^T \frac{\rho_w}{\rho_w^0} ns (\boldsymbol{n})^T \boldsymbol{v_w} dS = \boldsymbol{0} \quad (7.39)$$

In accordance with (5.35) the following relation is given

$$\begin{cases} \boldsymbol{\psi}_1(\boldsymbol{a}, u_w) + d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w) & = \boldsymbol{0} \\ \boldsymbol{\psi}_2(\boldsymbol{a}, u_w) + d\boldsymbol{\psi}_2(\boldsymbol{a}, u_w) & = \boldsymbol{0} \end{cases} \quad (7.40)$$

where the terms $d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w)$ and $d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w)$ needs further investigation in order to obtain the final expression for Newton's method.

Since the external forces do not depend on the strains or the pore pressure, the term $d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w)$ can be written as

$$d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w) = \int_V (\boldsymbol{B})^T d(\overline{\boldsymbol{\sigma}}) dV + \int_V (\boldsymbol{B})^T \boldsymbol{I} d(\chi u_w) dV \quad (7.41)$$

Introducing (7.36) into (7.41) then gives

$$d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w) = \int_V (\boldsymbol{B})^T \overline{\boldsymbol{D}} d\boldsymbol{\epsilon} dV + \int_V \frac{1}{3K_s} (\boldsymbol{B})^T \overline{\boldsymbol{D}} \boldsymbol{I} \left(\frac{ds}{du_w} u_w + s \right) du_w dV + \int_V (\boldsymbol{B})^T \boldsymbol{I} \left(\frac{d\chi}{du_w} + \chi \right) du_w dV \quad (7.42)$$

It is now possible to approximate the pore pressure u_w in the same fashion as described in (7.27) i.e.

$$u_w = \overline{\boldsymbol{N}} \boldsymbol{p} \quad (7.43)$$

where $\boldsymbol{p}$ is a vector containing the nodal values for the pore pressure. Since the shape functions do not depend on the pore pressure it is evident that

$$du_w = \overline{\boldsymbol{N}} d\boldsymbol{p} \quad (7.44)$$

Introducing (6.5), (7.43) and (7.44) into (7.42) gives

$$d\boldsymbol{\psi}_1(\boldsymbol{a}, u_w) = \int_V (\boldsymbol{B})^T \overline{\boldsymbol{D}}\boldsymbol{B} dV d\boldsymbol{a} + \left[\int_V \frac{1}{3K_s} (\boldsymbol{B})^T \overline{\boldsymbol{D}}\boldsymbol{I} \left(\frac{ds}{du_w}\overline{\boldsymbol{N}}\boldsymbol{p} + s\right) \overline{\boldsymbol{N}} dV + \int_V (\boldsymbol{B})^T \boldsymbol{I} \left(\frac{d\chi}{du_w}\overline{\boldsymbol{N}}\boldsymbol{p} + \chi\right) \overline{\boldsymbol{N}} dV\right] d\boldsymbol{p} \quad (7.45)$$

Evaluation of the term $d\boldsymbol{\psi}_2(\boldsymbol{a}, u_w)$ is a bit more complicated. To derive a useful expression, it must be assumed that the current configuration differs a bit from the reference configuration e.i. the differentiation of (7.39) at time $t + \Delta t$ is written as

$$d\boldsymbol{\psi}_2(\boldsymbol{a}, u_w) = \int_V \frac{1}{J}\left[(\overline{\boldsymbol{N}})^T d\left(J\frac{\rho_w}{\rho_w^0} sn\right) + \Delta t d\left(J\frac{\rho_w}{\rho_w^0} ns \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{v}_w\right)\right] dV \quad (7.46)$$

here the contribution from the surface integral has been neglected. The constitute relation given by (7.36) is then introduced to (7.46), this yields

$$d\boldsymbol{\psi}_2(\boldsymbol{a}, u_w) = \int_V \frac{1}{J}\left[(\overline{\boldsymbol{N}})^T d\left(J\frac{\rho_w}{\rho_w^0} sn\right) + \frac{\Delta t}{\rho_w^0 g} d\left(Jk_s \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{k}\left(\frac{\partial u_w}{\boldsymbol{x}} - \rho_w \boldsymbol{g}\right)\right)\right] dV \quad (7.47)$$

To evaluate the first differentiated term in (7.47), investigation of the quantity $J\frac{\rho_w}{\rho_w^0} sn$ is needed. To start with (7.3) is introduced to (7.15) giving

$$n = \frac{1}{1 + \frac{su_w}{K_s}}\left(\frac{\overline{p}}{K_s} + \frac{su_w}{K_s} + 1 - \frac{1}{J}(1 - n_0)\right) \quad (7.48)$$

Maclaurin expansion of the first term in (7.48) and neglecting all but first order terms will give

$$n \approx 1 + \frac{\overline{p}}{K_s} + \frac{1}{J}(1 + n^0)\left(\frac{su_w}{K_s} - 1\right) \quad (7.49)$$

Then, by introducing (7.14) to (7.49) and again neglecting all but first order terms, the following relation is given

$$\frac{\rho_w}{\rho_w^0} n \approx 1 + \frac{\overline{p}}{K_s} - \frac{1}{J}(1 - n^0) + u_w\left(\frac{1}{K_w} + \frac{1}{J}(1 - n^0)\left(\frac{s}{K_s} - \frac{1}{K_w}\right)\right) \quad (7.50)$$

Introducing (7.17) (7.36) and (7.12) into (7.50) yields

$$J\frac{\rho_w}{\rho_w^0} sn \approx Js\left[1 - \frac{1}{3K_s}\boldsymbol{I}^T\overline{\boldsymbol{D}}\left(\boldsymbol{\epsilon} + \frac{u_w}{3K_s}\boldsymbol{I}\right) + \frac{u_w}{K_w}\right] - s - + sn + u_w(1 - n^0)\left(\frac{s^2}{K_s} - \frac{s}{K_w}\right) \quad (7.51)$$

Differentiating (7.51) and neglecting small quantities compared with unity finally gives

$$d\left(J\frac{\rho_w}{\rho_w^0}sn\right) \approx \left(-sJ\frac{1}{3K_s}\boldsymbol{I}^T\overline{\boldsymbol{D}} + sJ\boldsymbol{I}^T\right)d\boldsymbol{\epsilon} + \left[\frac{ds}{du_w}(J-1+n^0) + (1-n^0)\left(\frac{s^2}{K_s} - \frac{s}{K_w}\right) + \frac{sJ}{K_w} - \frac{sJ}{9K_s^2}\boldsymbol{I}^T\overline{\boldsymbol{D}}\boldsymbol{I}\right]du_w \quad (7.52)$$

The second differentiated term an the right hand side in (7.47) is evaluated as

$$d\left[Jk_s\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\left(\frac{\partial u_w}{\partial \boldsymbol{x}} - \rho_w\boldsymbol{w}\right)\right] = Jk_s\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\left(\frac{\partial u_w}{\partial \boldsymbol{x}} - \rho_w\boldsymbol{w}\right)\boldsymbol{I}^T d\boldsymbol{\epsilon} + J\frac{dk_s}{ds}\frac{ds}{du_w}\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\left(\frac{\partial u_w}{\partial \boldsymbol{x}} - \rho_w\boldsymbol{g}\right)du_w + Jk_s\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\frac{\partial du_w}{\partial \boldsymbol{x}} \quad (7.53)$$

The expression for $d\boldsymbol{\psi}_2(\boldsymbol{a}, u_w)$ is obtained by introducing (7.52) and (7.53) into (7.47) and neglecting small terms, this gives

$$\int_V \left[-\frac{s}{3K_s}\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\overline{\boldsymbol{D}} + s\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\right]d\boldsymbol{\epsilon} + \left[\frac{ds}{du_w}\left(1 - \frac{1-n^0}{J}\right) + \frac{1-n^0}{J}\frac{s^2}{K_s} + \left(\frac{1-n^0}{J}\frac{s}{K_w} + \frac{s}{K_w}\right)\right]\overline{\boldsymbol{N}}du_w + \left[-\frac{s}{9K_s^2}\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\overline{\boldsymbol{D}}\boldsymbol{I} + \frac{\Delta t}{\rho_w^0 g}\frac{dk_s}{ds}\frac{ds}{du_w}\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\left(\frac{\partial u_w}{\partial \boldsymbol{x}} - \rho_w\boldsymbol{g}\right)\right]du_w + k_s\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\frac{\partial du_w}{\partial \boldsymbol{x}}dV \quad (7.54)$$

It is now possible to introduce the approximation for the pore pressures (7.43) into (7.54). In accordance with (7.28) it is evident that

$$\frac{\partial u_w}{\partial \boldsymbol{x}} = \overline{\boldsymbol{B}}\boldsymbol{p} \quad (7.55)$$

Finally (6.5), (7.43) and (7.55) together with (7.54) gives

$$\int_V -\frac{s}{3K_s}\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\overline{\boldsymbol{D}}\overline{\boldsymbol{B}} + s\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\boldsymbol{B}dVd\boldsymbol{a} + \int_V \frac{ds}{du_w}\left(1 + \frac{1-n^0}{J}\right)\left(\overline{\boldsymbol{N}}\right)^T\overline{\boldsymbol{N}} + \frac{1-n^0}{J}\frac{s^2}{K_s}\left(\overline{\boldsymbol{N}}\right)^T\overline{\boldsymbol{N}} + \left(\frac{1-n^0}{J}\frac{s}{K_w} + \frac{s}{K_w}\right)\left(\overline{\boldsymbol{N}}\right)^T\overline{\boldsymbol{N}} - \frac{s}{9K_s^2}\left(\overline{\boldsymbol{N}}\right)^T(\boldsymbol{I})^T\overline{\boldsymbol{D}}\boldsymbol{I}\overline{\boldsymbol{N}} + \frac{\Delta t}{\rho_w^0 g}\frac{dk_s}{ds}\frac{ds}{du_w}\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\left(\overline{\boldsymbol{B}}\boldsymbol{p} - \rho_w\boldsymbol{g}\right)\overline{\boldsymbol{N}} + k_s\left(\overline{\boldsymbol{B}}\right)^T\boldsymbol{k}\overline{\boldsymbol{B}}dVd\boldsymbol{p} \quad (7.56)$$

the relations (7.45) and (7.56) is then rewritten on the form given by (5.37) as

$$-\begin{bmatrix} \boldsymbol{\psi}_1(\boldsymbol{a}, u_w) \\ \boldsymbol{\psi}_2(\boldsymbol{a}, u_w) \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_{aa} & \boldsymbol{K}_{ap} \\ \boldsymbol{K}_{pa} & \boldsymbol{K}_{pp} \end{bmatrix} \begin{bmatrix} d\boldsymbol{a} \\ d\boldsymbol{p} \end{bmatrix} \tag{7.57}$$

where

$$\begin{cases} \boldsymbol{K}_{aa} = \displaystyle\int_V (\boldsymbol{B})^T \overline{\boldsymbol{D}}\boldsymbol{B} dV \\ \boldsymbol{K}_{ap} = \displaystyle\int_V \frac{1}{3K_s} (\boldsymbol{B})^T \overline{\boldsymbol{D}}\boldsymbol{I} \left(\frac{ds}{du_w} \overline{\boldsymbol{N}}\boldsymbol{p} + s \right) \overline{\boldsymbol{N}} dV + \int_V (\boldsymbol{B})^T \boldsymbol{I} \left(\frac{d\chi}{du_w} \overline{\boldsymbol{N}}\boldsymbol{p} + \chi \right) dV \\ \boldsymbol{K}_{pa} = -\displaystyle\int_V \frac{s}{3K_s} \left(\overline{\boldsymbol{N}}\right)^T (\boldsymbol{I})^T \overline{\boldsymbol{D}}\boldsymbol{B} dV + \int_V s \left(\overline{\boldsymbol{N}}\right)^T (\boldsymbol{I})^T \boldsymbol{B} dV \\ \boldsymbol{K}_{pp} = \displaystyle\int_V \frac{ds}{du_w} \left(1 - \frac{1-n^0}{J} \right) \left(\overline{\boldsymbol{N}}\right)^T \overline{\boldsymbol{N}} dV + \int_V \frac{1-n^0}{J} \frac{s^2}{K_s} \left(\overline{\boldsymbol{N}}\right)^T \overline{\boldsymbol{N}} dV + \\ \qquad \displaystyle\int_V \left(\frac{1-n^0}{J} \frac{s}{K_w} + \frac{s}{K_w} \right) \left(\overline{\boldsymbol{N}}\right)^T \overline{\boldsymbol{N}} dV - \int_V \frac{s}{9K_w^2} \left(\overline{\boldsymbol{N}}\right)^T (\boldsymbol{I})^T \overline{\boldsymbol{D}}\boldsymbol{I}\overline{\boldsymbol{N}} dV + \\ \qquad \displaystyle\int_V \frac{\Delta t}{\rho_w^0 g} \frac{dk_s}{ds} \frac{ds}{du_w} \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{k} \left(\overline{\boldsymbol{B}}\boldsymbol{p} - \rho_w \boldsymbol{g} \right) \overline{\boldsymbol{N}} dV + \int_V k_s \left(\overline{\boldsymbol{B}}\right)^T \boldsymbol{k}\overline{\boldsymbol{B}} dV \end{cases}$$

From (7.57) it is possible to calculate the corrections for each variable and then the new state is given according to (5.38)

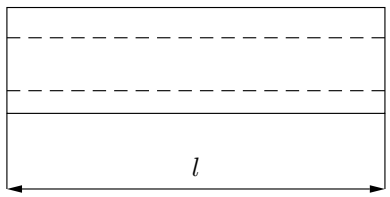
Chapter 8

Simulation of bone fluid flow in cortical bone

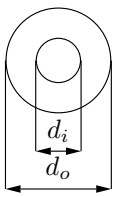
In this chapter simulations of bone fluid movement in cortical bone will be done. The issue is to investigate the magnitude of bone fluid flow in the diaphysis of a human femur exposed to different loading situations. The fluid flow is assumed to take place on the vascular porosity level since this porosity corresponds to the whole bone level. The simulations will be preformed by using a commercial finite element code (ABAQUS).

8.1 Geometry and mechanical properties

The geometry used in the simulations represents the proximal half of a human femural diaphysis. This has been approximated by a cylindrical tube, see Fig. 8.1.



(a) Geometry of diaphysis of femur, side view



(b) Geometry of diaphysis of femur, front view

Figure 8.1: Geometry of femural diaphysis

From this geometry a FE-model was created. The element type that has been used is fully integrated, 8-nodal linear brick element that are able to represent a linear pore pressure distribution. The FE-model can be seen in Fig. 8.2.

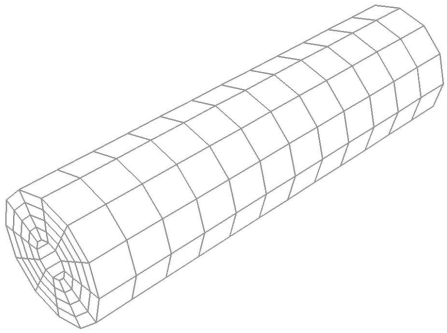


Figure 8.2: FE-model of femural diaphysis

To be able to represent the loading conditions that will be discussed in the next section, rigid elements where attached to the proximal end of the model.

The dimensions and mechanical properties that has been used in the simulation are listed in table 8.1 (see Cowin [3])

Property	Value
l (mm), length	120
d_o (mm), outer diameter	34
d_i (mm), inner diameter	10
n, porosity	0.04
E_s (GPa), Young's modulus	15
K_s (GPa), bulk modulus of solid	15
ν , Poisson's ratio	0.33
k (m/s), isotropic permeability	$50 \cdot 10^{-9}$
K_w (GPa), bulk modulus of fluid	2.2
ρ_w (kg/m^3), density of fluid	1000

Table 8.1: Geometrical and mechanical properties of the model

8.2 Loading situation and boundary conditions

The loading situation and the mechanical boundary conditions can be seen if Fig. 8.3 The fixed end is due to symmetry, this end represents the mid section of the femur. The applied force and moment represents the loads from the femural head and the abductor muscles. The nature of the applied load will be described in detail in the next sections.

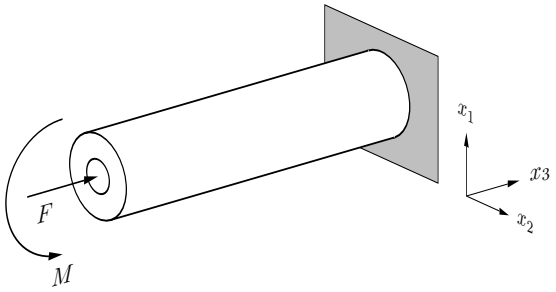


Figure 8.3: Boundary conditions and load case

The boundary conditions for the pore pressure is taken from a similar simulation done by Cowin [3]. Here the pore pressure on the endosteal surface is prescribed to be 3.3 kPa and it is assumed to remain constant during the loading cycle. Furthermore, the periosteal surface and the ends are assumed to be unpermeable for the bone fluid.

8.3 Gait walk analysis

Regular walking is probably the most common loading situation exposed to the long bones in the legs. Due to this fact it is of high interest to investigate how this type of loading will affect the of bone fluid flow.

The results from the simulations are examined in a section halfway thought the model. This section is choosen to minimize the effects of the boundary conditions for the ends. Furthermore will all quantities that are plotted as a function of time be evaluated in just one point at this section. The section and evaluation point can be seen if Fig. 8.4

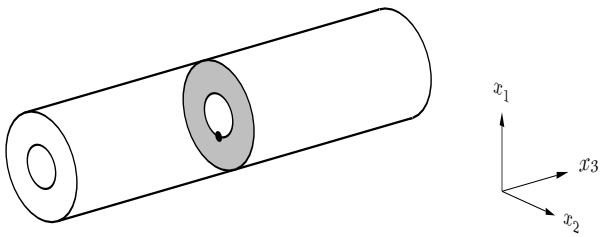
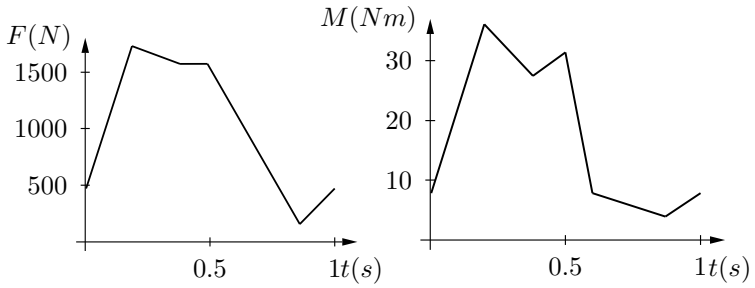


Figure 8.4: Section and evaluation point for the model

The loading situation used in this section is taken from Bergmann [2] who has measured the load acting on the proximal femur during some routine activities. The result for normal walking can be seen in Fig. 8.5 (the terms "compressing force" and the "bending moment" corresponds to the notations in Fig. 8.3).



(a) Compressing force (b) Bending Moment

Figure 8.5: Loading situation during a gait cycle

These loadings are used as input data to the FE-simulation.

The first thing that is to be examined is the magnitude and the time dependence of the bone fluid flow. The result can be seen in Fig. 8.6

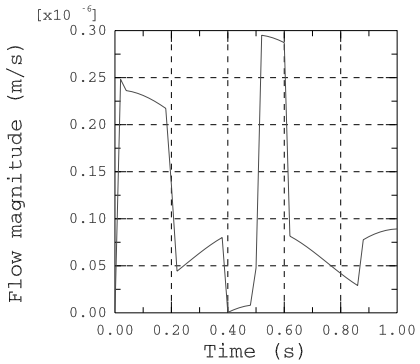
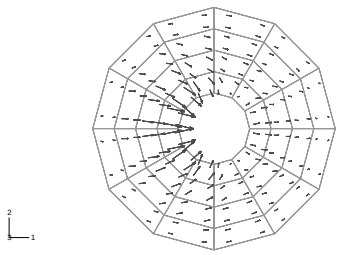
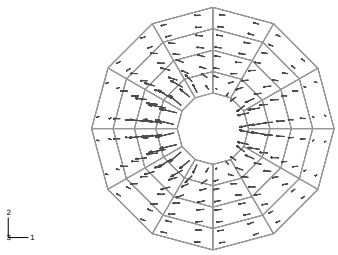


Figure 8.6: Magnitude of fluid flow for gait walk analysis

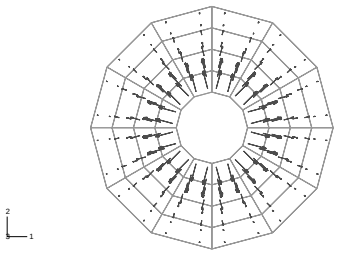
Secondly, the bone fluid flow directions throughout the mid section has been examined. The flow directions under different stages in the loading cycle can be seen in Fig. 8.7



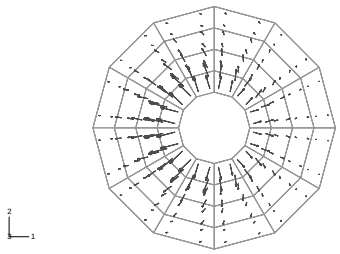
(a) Fluid flow, time=0.05 s.



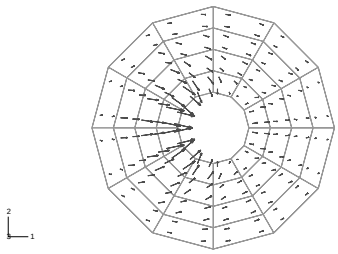
(b) Fluid flow, time=0.25 s.



(c) Fluid flow, time=0.50 s.



(d) Fluid flow, time=0.75 s.



(e) Fluid flow, time=1.00 s

Figure 8.7: Fluid flow directions for different stages during a gait load cycle.

8.4 Vibration analysis

As stated earlier, one of the most interesting issues for this thesis is to investigate how different loading situations effect the fluid flow in cortical bone. In this section the behavior of the fluid flow will be investigated when the model is exposed to a pulsating loading situation.

The quantities will be evaluated in the same manner as in the previous section (see Fig. 8.4).

The loading situation is assumed to take the form of

$$\begin{cases} F = F_0 + F_0 \sin(2\pi f t) \\ M = M_0 + M_0 \sin(2\pi f t) \end{cases} \tag{8.1}$$

where f is the frequency for the load. Simulations are then done using different values for the constants F_0, M_0 and f. The ratio between F_0 and M_0 is kept constant and is taken form Bagge [1]. The frequency of the load is variated in the range from 1 Hz to 10 Hz. Since the model does not include inertia forces, higher values of the frequency will result in too inaccurate results.

The results from the different simulations can be seen Fig.8.8, where the values plotted is the peak value of the fluid flow during the loading cycle. The magnitude of the fluid flow is given by a sine function with the same frequency as the load.

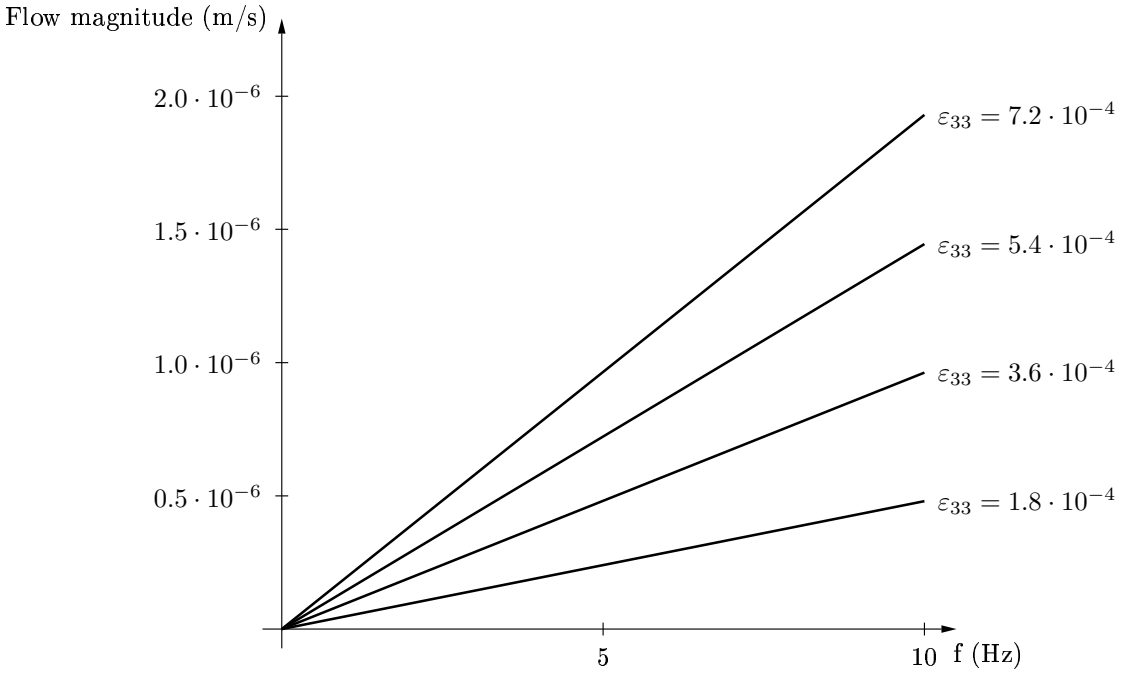
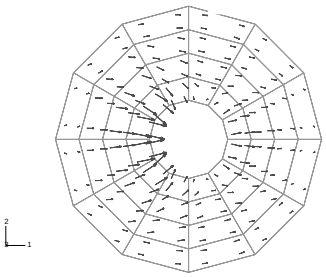


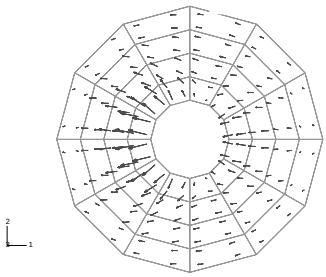
Figure 8.8: Magnitude of fluid flow for different frequencies

As can be seen in Fig. 8.8 there is a linear relation between the magnitude of the fluid flow and the frequency. Furthermore are the magnitude of strains for the different simulations within physiological correct values [4]

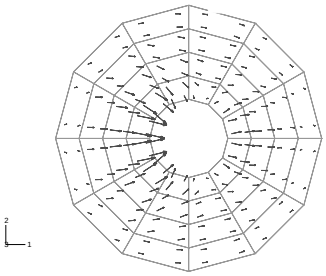
The direction of the fluid flow for different stages of the pulsating loading cycle is given in Fig. 8.9



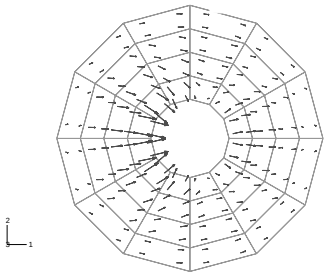
(a) Fluid flow, time=0.05 s.



(b) Fluid flow, time=0.30 s.



(c) Fluid flow, time=0.80 s.



(d) Fluid flow, time=1.00 s

Figure 8.9: Fluid flow directions for different stages during a pulsating load cycle ($\varepsilon_{33} = 1.8 \cdot 10^{-4}$, 1 Hz).

The pattern of the fluid flow directions obtained from the different frequencies and load magnitude was identical.

Chapter 9

Discussion and future work

In this chapter a discussion of the accuracy and the reliability of the different simulations will be carried out. Future work for the project will also be discussed since this thesis only has dealt with a fraction of all the issues involved in the area of adaptive bone remodelling.

9.1 Discussion of the results

Initially it can be discussed whether the stimuli chosen to be investigated in this thesis are representative for real bone remodelling process. This is hard to answer since the actual medical reason for bone remodelling by no meas has been established. It is very likely that different choices of stimulus would result in similar results.

The result from the simulation done in chapter 6, bone adaptation simulation by use of an updating algorithm for Young's modulus, shows grate similarities to the results obtained by Huiskes, even thought a different solution technique has been used. It can furthermore be assumed that the solution technique used in this thesis will result in a numerical more accurate result, since the Newton's method provides control of the residual (out of balance forces). Whether the result corresponds to real life is a bit more uncertain. It is clear that the model can not represent the development of a human femur since the simulation starts for a physiologically incorrect state i.e equal density distribution, this process is more likely to be guided by genetic factors. The final state shows some similarities to a adult human femur so it can be assumed that some similar density distribution process is present in the real bone remodelling process. It is worth mentioning that the simulation will not result in a global optimal structure since the aim of the updating algorithm is to give a uniform strain energy density distribution. An optimal structure is given by a minimization of the strain energy.

The magnitude of the results from the simulation of bone fluid flow are in the same order as the result obtained by Cowin [3]. The linear relation between fluid flow and frequency

where also found by Weinbaum [16]. How these results match the the real behavior of the bone fluid is hard to say since no medical testing with similar loading situation was found in the literature. There are two major drawbacks with the model used in the simulations. The first and most important one is that no inertia forces was included. This effects the result for higher frequencies since then the inertia forces would be one of the major influences on the behavior of the bone fluid. The reason for not including inertia forces was due to a software problem (ABAQUS does not include this in their code), so only low frequency loading could by investigated. The second drawback of the model is that it only can predict the bone fluid flow in the vascular porosity of the bone. It was stated in chapter 4 that the fluid flow in the canaliculi-lacunae porosity could play an important roll in bone remodelling so the magnitude of the bone fluid flow on this level of porosity would be of interest. However, it can be assumed that the behavior of the fluid flow in the canaliculi-lacunae porosity can be represented by the fluid flow on vascular porosity level.

9.2 Suggestions for future work

The major effect that has to be incorporated in a future model is probably, as mentioned in the previous section, inertia effects. This has to be done by implementation of the model in for example Matlab or Fortran since this allows for better control of the issues involved. A development of a two level porosity model, similar to the one described by Cowin [3], is also vital for an accurate simulation of the bone fluid behavior. Furthermore may also the blood pressure driven bone fluid flow be taken into account. The ultimate goal for the model is probably to link together the two models investigated in this thesis providing an updating algorithm for material properties due to a physiological and medical correct stimuli.

Bibliography

[1] M. Bagge. *Remodelling of bone structure.* PhD thesis, Danich center for applied mathematics and mechanics, March 1999.

[2] G. Bergmann, G. Deuretzbacher, M. Heller, F. Graichen, and A. Rohlmann. Hip contact force and gait pattern from routine activities. *Journal of Biomechanics*, (34):859–871, 2001.

[3] S.C. Cowin, S.P. Fritton, L. Wang, and S. Weinbaum. Interaction of mechanical loading, blood flow, and interstitial fluid flow in osteonal bone. *ASME*, 2001.

[4] S.C. Cowin, L. You, M.B. Schaffler, and S. Weinbaum. A model for strain amplification in the actin cytoskeleton of osteocytes due to fluid drag on pericellular matrix. *Journal of Boimechanics*, (34):1375–1386, 2001.

[5] S.P Fritton, J.M Kenneth, and C.T Rubin. Quantifying the strain history of bone: spatial uniformity and self-similarity of low magnitude strains. *Journal of Biomechanics*, (33):317–325, 2000.

[6] Hibbit, Karlsson, and Sorensen. *ABAQUS Theory manual.* 1998.

[7] R. Huiskes, H. Weinans, H.J. Grootenboer, M. Dalstra, B. Fudala, and T.J Slooff. Adaptive bone-remodling theory applied to prosthetic-design analysis. *Journal of Biomechanics*, (20):1135–1150, 1987.

[8] L.E. Layon and C. T. Rubin. Static vs dynamic loads as an influence on bone remodelling. *Journal of Biomechanics*, 17(12):897–905, 1984.

[9] N.S. Ottosen and M. Ristinmma. *The Mechanics of Constitutive Modelling*, volume 1. Division of Solid Mechanics, Lund, 1999.

[10] N.S. Ottosen and M. Ristinmma. *The Mechanics of Constitutive Modelling*, volume 2. Division of Solid Mechanics, Lund, 1999.

[11] C. Rubin, G. Xu, and S. Judex. The anabolic activity of bone tissue, supressed by disuse, is normalized by brief exposure to extremely low-magnitude mechanical stimuli. *The FASEB Journal*, 15:2225–2229, October 2001.

[12] C.T. Rubin and L.E Layon. Regulation of bone formation by applied dynamic loads. *Journal of Bone and Joint Surgery*, (66A):397–410, 1984.

[13] B. Sonesson and G. Sonesson. *Manniskans anatomi och fysiologi.* Liber, Stockholm, 1993.

[14] L.A. Taber. Biomechanics of groeth, remobelling and morphogenesis. *Applied Mechanics Reviews*, 48(8):487–545, August 1995.

[15] R.W. Treharne. Review of wolff's law and its proposed means of operation. *Ortop Rev*, (10):35–37, 1981.

[16] S. Weinbaum, S.C. Cowin, and Y. Zeng. A model for the exitation of osteocytes by mechanical loading-induced bone fluid shear stresses. *Journal of Biomechanics*, 27(3):339–360, 1993.

[17] J. Wolff. *The Law of Bone Remodelling.* Springer, Berlin, 1986.

Appendix A

Matlab code

Here the Matlab code used in for the simulations done in chapter 6 is listed. The geometry file has been excluded since it is to sparse to print.

```
%------------Huiskes bone remodelling (1987)----------%
%                                                     %
%                   Main program                      %
%               By: Magnus Harrysson                  %
%                                                     %
%-----------------------------------------------------%
clear all warning debug
[Edof,Dof,bc,ndof,nelm,Ex,Ey,Edof_plot,ndof_plot]=bone_geometry2;


U=ones(nelm,1); Un=0; a=zeros(ndof,1); E=0.5e4*ones(nelm,1);
E_new=0.5e4*ones(nelm,1); ny=0.3; ep=[2 30]; t=30; kass=0;

dt=0.0005; nstep=7000; Tplot(1)=0; Eplot(:,1)=E(:,1);

eps1=0.0000005; eps2=1e-5;

fi_1=-pi/9; fi_2=-pi/9; p_1=-3000; p_2=1400;

f=zeros(ndof,1); df=zeros(ndof,1); f1=zeros(ndof,1);


f1(251)=p_1*sin(fi_1); f1(252)=p_1*cos(fi_1);
f1(221)=p_2*sin(fi_2); f1(222)=p_2*cos(fi_2);

f_int=zeros(ndof,1); f_int_norm=25*ones(ndof,1); K=zeros(ndof);

%------------Begin load step-------------------

for step=1:nstep
```

```
iter=0;

%-------Begin iteration loop---------------

while norm(f1-f_int_norm,2) > eps1 | iter==0

    iter=iter+1;


%------------Calculate tangential stiffness matrix------

    K=zeros(ndof);
    for i=1:nelm
        if iter==1 & step==1
            D(:,:,i)=hooke(2,E(i),ny);
            Ke=plantec(Ex(i,:),Ey(i,:),t,D(:,:,i));
            assemc(Edof(i,:),K,Ke);

        else
            Ke=plantec(Ex(i,:),Ey(i,:),t,Dt(:,:,i));
            assemc(Edof(i,:),K,Ke);
        end

    end

    %---------Solve displacements-------------


    da=solveq(sparse(K),f1-f_int,bc);

    a=a+da;



%----------Set value for reference SED--------
    if iter==1 & step==1

        for i=1:nelm

            Ed=extract(Edof,a);
            [es,et]=plants(Ex(i,:),Ey(i,:),ep,D(:,:,i),Ed(i,:));
            U_ne(i,1)=0.5*(es(1,:)*et(1,:)');
        end
        Un=sum(U_ne,1)/nelm;
    end

%------Calculate new E-modulus, internal loadvector and tangential stiffness---------


    [E_new,f_int,U,Dt,dE]=E_update3(E,ny,nelm,ndof,Ex,Ey,a,ep,dt,Edof,Un,step);
```

```
        f_int_norm=f_int;

        for i=1:length(bc)
            f_int_norm(bc(i,1))=0;
        end

        if iter==30

            f_int_norm=f1;
                kass=kass+1;        %if convergence not is obtained
                fkass=f(5);
                break
            if kass==1
                Ekass=E_new;
            end
        end


        iter
        disp('norm')
        norm(f1-f_int_norm,2)       %print to screen
        step
        kass
        history(step,iter)=norm(f1-f_int_norm,2);

    end
    %----End of iteration-loop---------

    E=E_new;

    Eplot(:,step+1)=E;
    Tplot(step+1)=Tplot(step)+dt;
    if iter < 4
        dt=dt*1.1;
    end

    if iter > 5
        dt=dt*0.9;
    end


end
%---------Plot functions----------------

first=[1:1:nelm]'; Edof_plot2=[first Edof_plot];

for i=1:ndof_plot
    Ee_plot_sum=0;

    [r,c]=find(Edof_plot==i);
```

```
    for j=1:length(r)
        Eee=E(r(j),1);
        Ee_plot_sum=Ee_plot_sum+Eee;

    end
    Ee_plot=Ee_plot_sum/length(r);
    E_nod(i,1)=Ee_plot;
end


%-------------------------------------------------------------------%
%                                                                   %
%                       E_update function                           %
%                       By: Magnus Harrysson                        %
%                                                                   %
%-------------------------------------------------------------------%

function[E_new,f_int,U,Dt,dE]=E_update3(E,ny,nelm,ndof,Ex,Ey,a,ep,dt,Edof,Un,step)


L=zeros(6); H=zeros(6); C=10e6; D_red=[1-ny ny  ny 0
       ny 1-ny  ny 0
       ny ny 1-ny  0
       0 0 0 0.5*(1-2*ny)];

Ed=extract(Edof,a);

f_int=zeros(ndof,1);

for j=1:nelm

    %-----------solve increment of E-modulus----------
    [es,et]=plants(Ex(j,:),Ey(j,:),ep,D_red,Ed(j,:));

    T_et=et*D_red*et';
    C1=T_et*C*dt/(2*(1+ny)*(1-2*ny));
    C2=T_et*C*dt*E(j,1)/(2*(1+ny)*(1-2*ny))-C*dt*Un;

    dE=C2/(1-C1);

    %-----------Calculate new E-modulus-----------------

    if E(j,1)+dE > 50 & E(j,1)+dE < 2.5e4

        E_new(j,1)=E(j,1)+dE;
    else

        E_new(j,1)=E(j,1);
    end

    %----------Calculate new stress and internal force-----
```

```
sigma=E_new(j,1)/((1+ny)*(1-2*ny))*D_red*et';

%sigma_33=E_new(j,1)*ny/((1+ny)*(1-2*ny))*(et(1)+et(2));

sigmaf=[sigma(1)
    sigma(2)
    sigma(4)];

fe=(plantf(Ex(j,:),Ey(j,:),ep,sigma'));
f_int=insert(Edof(j,:),f_int,fe);

U(j,1)=0.5*et*sigma;

%----------Calculate new tangential stiffness-----------

if E(j,1)+dE > 50 & E(j,1)+dE < 2.5e4

    A=hooke(4,1,ny);
    D_1=hooke(4,E_new(j,1),ny);
    Q=1-(C*dt*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))*0.5*et(1)+
      C*dt*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))*0.5*et(2)+C*dt*(A(3,1)*et(1)
      +A(3,2)*et(2)+A(3,4)*et(4))*0.5*et(4));

    B=[C*dt*0.5*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))*sigma(1)
       C*dt*0.5*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))*sigma(1)
       C*dt*0.5*(A(3,1)*et(1)+A(3,2)*et(2)+A(3,4)*et(4))*sigma(1)
       C*dt*0.5*(A(4,1)*et(1)+A(4,2)*et(2)+A(4,4)*et(4))*sigma(1)
       C*dt*0.5*(A(5,1)*et(1)+A(5,2)*et(2)+A(5,4)*et(4))*sigma(1)
       C*dt*0.5*(A(6,1)*et(1)+A(6,2)*et(2)+A(6,4)*et(4))*sigma(1)
          %-----second column-----------
       C*dt*0.5*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))*sigma(2)
       C*dt*0.5*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))*sigma(2)
       C*dt*0.5*(A(3,1)*et(1)+A(3,2)*et(2)+A(3,4)*et(4))*sigma(2)
       C*dt*0.5*(A(4,1)*et(1)+A(4,2)*et(2)+A(4,4)*et(4))*sigma(2)
       C*dt*0.5*(A(5,1)*et(1)+A(5,2)*et(2)+A(5,4)*et(4))*sigma(2)
       C*dt*0.5*(A(6,1)*et(1)+A(6,2)*et(2)+A(6,4)*et(4))*sigma(2)
          %-----third column-----------
       C*dt*0.5*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))*sigma(3)
       C*dt*0.5*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))*sigma(3)
       C*dt*0.5*(A(3,1)*et(1)+A(3,2)*et(2)+A(3,4)*et(4))*sigma(3)
       C*dt*0.5*(A(4,1)*et(1)+A(4,2)*et(2)+A(4,4)*et(4))*sigma(3)
       C*dt*0.5*(A(5,1)*et(1)+A(5,2)*et(2)+A(5,4)*et(4))*sigma(3)
       C*dt*0.5*(A(6,1)*et(1)+A(6,2)*et(2)+A(6,4)*et(4))*sigma(3)
          %-----forth column-----------
       C*dt*0.5*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))*sigma(4)
       C*dt*0.5*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))*sigma(4)
       C*dt*0.5*(A(3,1)*et(1)+A(3,2)*et(2)+A(3,4)*et(4))*sigma(4)
       C*dt*0.5*(A(4,1)*et(1)+A(4,2)*et(2)+A(4,4)*et(4))*sigma(4)
       C*dt*0.5*(A(5,1)*et(1)+A(5,2)*et(2)+A(5,4)*et(4))*sigma(4)
       C*dt*0.5*(A(6,1)*et(1)+A(6,2)*et(2)+A(6,4)*et(4))*sigma(4)
```

```
      %-----fifth column-----------
   0
   0
   0
   0
   0
   0
      %-----sixth column-----------
   0;
   0;
   0;
   0;
   0;
   0;
      %-----------------------------
   ]+D_1;

N=[C*dt*(A(1,1)*et(1)+A(1,2)*et(2)+A(1,4)*et(4))
   C*dt*(A(2,1)*et(1)+A(2,2)*et(2)+A(2,4)*et(4))
   C*dt*(A(3,1)*et(1)+A(3,2)*et(2)+A(3,4)*et(4))
   C*dt*(A(4,1)*et(1)+A(4,2)*et(2)+A(4,4)*et(4))
   C*dt*(A(5,1)*et(1)+A(5,2)*et(2)+A(5,4)*et(4))
   C*dt*(A(6,1)*et(1)+A(6,2)*et(2)+A(6,4)*et(4))];


Dt_tot=[B(1,1)+N(1)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
        B(2,1)+N(2)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
        B(3,1)+N(3)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
        B(4,1)+N(4)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
        B(5,1)+N(5)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
        B(6,1)+N(6)/(Q)*(0.5*et(1)*B(1,1)+0.5*et(2)*B(2,1)+0.5*et(4)*B(4,1))
           %-----second column-----------
        B(1,2)+N(1)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
        B(2,2)+N(2)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
        B(3,2)+N(3)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
        B(4,2)+N(4)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
        B(5,2)+N(5)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
        B(6,2)+N(6)/Q*(0.5*et(1)*B(1,2)+0.5*et(2)*B(2,2)+0.5*et(4)*B(4,2))
           %-----third column-----------
        B(1,3)+N(1)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
        B(2,3)+N(2)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
        B(3,3)+N(3)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
        B(4,3)+N(4)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
        B(5,3)+N(5)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
        B(6,3)+N(6)/Q*(0.5*et(1)*B(1,3)+0.5*et(2)*B(2,3)+0.5*et(4)*B(4,3))
           %-----fourth column-----------
        B(1,4)+N(1)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
        B(2,4)+N(2)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
        B(3,4)+N(3)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
        B(4,4)+N(4)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
        B(5,4)+N(5)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
```

```
                B(6,4)+N(6)/Q*(0.5*et(1)*B(1,4)+0.5*et(2)*B(2,4)+0.5*et(4)*B(4,4))
                   %-----fifth column-----------
                B(1,5)+N(1)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                B(2,5)+N(2)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                B(3,5)+N(3)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                B(4,5)+N(4)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                B(5,5)+N(5)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                B(6,5)+N(6)/Q*(0.5*et(1)*B(1,5)+0.5*et(2)*B(2,5)+0.5*et(4)*B(4,5))
                   %-----sixth column-----------
                B(1,6)+N(1)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))
                B(2,6)+N(2)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))
                B(3,6)+N(3)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))
                B(4,6)+N(4)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))
                B(5,6)+N(5)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))
                B(6,6)+N(6)/Q*(0.5*et(1)*B(1,6)+0.5*et(2)*B(2,6)+0.5*et(4)*B(4,6))];

        Dt(:,:,j)=[Dt_tot(1,1) Dt_tot(1,2) Dt_tot(1,3) Dt_tot(1,4)
                   Dt_tot(2,1) Dt_tot(2,2) Dt_tot(2,3) Dt_tot(2,4)
                   Dt_tot(3,1) Dt_tot(3,2) Dt_tot(3,3) Dt_tot(3,4)
                   Dt_tot(4,1) Dt_tot(4,2) Dt_tot(4,3) Dt_tot(4,4)];


    else
        Dt(:,:,j)=E_new(j,1)/((1+ny)*(1-2*ny))*D_red;

    end

end


%-------------------------------------------------------%
%                                                       %
%                   Plot fuction                        %
%                   By: Magnus Harrysson                %
%                                                       %
%-------------------------------------------------------%

function plotfuncel(nelm,Edof_plot,ndof_plot,E,Ex,Ey) figure(3)
axis([-120 155 0 250])

for i=1:nelm
    if E(i)>=22.5e3  &  E(i)<2.5e4
        C=[0 0 0];
    end

    if E(i)>=20e3  &  E(i)<22.5e3
        C=[0.1 0.1 0.1];
    end

    if E(i)>=17.5e3  &  E(i)<20e3
        C=[0.2 0.2 0.2];
```

```
    end

    if E(i)>=15e3  &  E(i)<17.5e3
        C=[0.3 0.3 0.3];
    end

    if E(i)>=12.5e3  &  E(i)<15e3
        C=[0.4 0.4 0.4];
    end

    if E(i)>=10e3  &  E(i)<12.5e3
        C=[0.5 0.5 0.5];
    end

    if E(i)>=7.5e3  &  E(i)<10e3
        C=[0.6 0.6 0.6];
    end

    if E(i)>=5e3  &  E(i)<7.5e3
        C=[0.7 0.7 0.7];
    end

    if E(i)>=2.5e3  &  E(i)<5e3
        C=[0.8 0.8 0.8];
    end

    if E(i)>=0e3  &  E(i)<2.5e3
        C=[1 1 1];
    end

    hold on
    fill(Ex(i,:),Ey(i,:),C)
end

xp=[100 120 120 100]; for i=1:9
    yp=[230-i*10 230-i*10 230-i*10-10 230-i*10-10];
    fill(xp, yp,[-0.1+i*0.1 -0.1+i*0.1 -0.1+i*0.1])
end fill(xp,[140 140 130 130], [1 1 1]);

text(98, 235, 'E (MPa)') text(125, 130, '0.0')
text(125,220,'2.5*10^4')
```