Implementation of an MPFA/MPSA-FV Solver for the Unsaturated Flow in Deformable Porous Media

MASTER'S THESIS IN APPLIED AND COMPUTATIONAL MATHEMATICS

JHABRIEL VARELA

Department of Mathematics
University of Bergen

May, 2018

Acknowledgments

Foremost, I would like to thank my family for the unconditional support I have received throughout these years.

There are no words to express my gratefulness towards my supervisors: Jan Martin Nordbotten, Eirik Keilevgalen, and Sarah Gasda, who relentlessly and patiently guided me through my Master's research.

I must acknowledge the generous financial support of the National Postgraduate Scholarships Program: Carlos Antonio López (BeCAL); this thesis would not have been possible without the grant they gratified me.

I would also like to thank my former supervisors Juan Pablo Nogues and Christian Schaerer for introduced me to the field of flow in porous media and encouraged me to pursue a postgraduate program.

On a more personal note, I would like to thank all my friends in Paraguay that despite the distance inspired me to keep going ahead. Also, I was fortunate getting my new family at the Fantoft Student Hostel, where countless dinners and parties had taken place. Many of them, will be hard to forget. Finally, I would like to thank my girlfriend Patricia Maldonado, without your love, patient and support it would not have been possible to get to the finish line.

Jhabriel Varela
May, 2018

Abstract

The Unsaturated Flow In Deformable Porous Media (UFIDPM) plays a crucial role in several academic and industrial applications such as; cracks induced by desiccation, collapsing soils, ground movement involving expansive soils, lateral earth surfaces, stability of vertical excavations, natural slopes subjected to environmental changes, construction and operation of a dam, etc. Typically, studying these systems with experimental techniques is impractical. On the other hand, numerical simulations allow us to investigate several scenarios in a short time and with reduced costs. At the moment, only Finite Element Method (FEM) based codes were available. FEM often shows conservative issues and when is applied to UFIDPM instabilities in the limit of incompressibility have been reported. Due to this issues, we were motivated to develop the first Cell Centered Finite Volume Method (CCFVM) based code for solving UFIDPM.

The governing equations are derived following the extended Biot's theory of three-dimensional consolidation which results in a coupled hydro-mechanical system. For the flow problem, we use the Richards' equation whereas the mechanical problem is modeled using the linear elasticity equations. For the spatial discretization, we use multi-points approximations schemes. Specifically, Multi Point Flux Approximation (MPFA) for the flow problem and Multi Point Stress Approximation (MPSA) for the mechanical problem. Moreover, the time discretization of the equations was obtained using Backward Euler (BE). The code was implemented in MATLAB R2017b were two core toolboxes (MRST and FV-BIOT) were used. The resulting non-linear set of equations was solved using the Newton method together with Automatic Differentiation (AD).

To test the capability of the code several sub-problems were validated. Furthermore, we present a numerical application where we focus on the desiccation process of a clayey soil in a Petri-dish. In this experiment, the water content reduction is caused by instantaneous water evaporation controlled by atmospheric conditions. Finally, by carefully post-processing the resulting stress field, the zones of tensile stress concentration, which corresponds to the areas where cracks are more likely to initiate, are identified.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, the interest in studying the unsaturated flow in deformable porous materials has increased considerably, primarily due to its broad range of application in hydro-geological systems. Within the most important applications we can mention: the determination of critical zones of crack formation in desiccating clayey soils, the construction and operation of dams, the stability analysis of vertical and near vertical excavations, the ground movement involving expansive soils, the consolidation of unsaturated soils, among several others (Fredlund & Rahardjo 1993).

Numerical simulations allow us to study these type of systems in reasonable time spans and reduced attached costs, in contrast to laboratory experimentation. Therefore, there is an imperative need to count with robust, flexible and computationally-efficient codes. In this context, extensive research has been done in developing FEM-based solvers. Nevertheless, FEM intrinsically lacks conservative properties, and moreover, instabilities in the limit of solid incompressibility have been reported (Nagtegaal et al. 1974, Brezzi & Fortin 2012). In contrast to FEM, FVM shows an inherent conservative property whereas the same advantages are preserved, e.g. flexibility in representing complex domains. However, FVM-based codes for simulating the unsaturated flow in deformable porous media were not yet available. Motivated by filling this gap, in this thesis we present the first Cell Centered Finite Volume Method capable of simulating these type of systems.

The governing equations for the unsaturated flow in deformable porous media are derived following the extended Biot's theory of three-dimensional consolidation (Lewis & Schrefler 1998), which results in a coupled hydro-mechanical system. Instead of using a full two-phase formulation to model the flow problem, we use the Richards' assumption of inviscid air where only the water motion is taken into account. On the other hand, the mechanical responses are described using the linear elasticity equations.

As we already mentioned, the spatial discretization of the resulting governing equations is made using a cell-centered finite volume scheme. In particular, for the flow problem, we use Multi Point Flux Approximation (MPFA), whereas the linear elastic problem is discretized using Multi Point Stress Approximation (MPSA). Moreover, for the time discretization, we use Backward Euler.

The code is implemented in Matlab R2017b where we use two core toolboxes: Matlab Reservoir Simulation Toolbox (MRST) and FV-Biot. MRST provides the basic data structure, mesh generation capabilities and Automatic Differentiation (AD) that is being used together with the Newton method to solve the resulting non-linear set of discrete equations. Besides, FV-Biot provides the MPFA/MPSA discretization and the coupling terms to solve the unsaturated poroelastic system.

From a modeling standpoint, this thesis is divided into four parts, i.e., the unsaturated flow in non-deformable porous media (Richards' equation), the linear elasticity equations (Navier-Lamé equations), the saturated flow in deformable porous media (Biot equations) and the unsaturated flow in deformable porous medium. The primary purpose of structuring the thesis this way is to present the concepts, governing equations and discretization procedure in increasing order of complexity until we arrive at the final model, which in turn, can be seen as a generalization of the previous three parts.

In the following, we present a summary of the chapters covered in this thesis

Chapter 2: covers the fundamental concepts of the flow in unsaturated porous media, the linear elastic behavior of solid bodies and the saturated/unsaturated linear poromechanics.

Chapter 3: focuses on the derivation of the governing equations for each of the four parts of the thesis, using the concepts introduced in Chapter 2.

Chapter 4: deals with the discretization and implementation aspects. We start giving a brief description of the MPFA and MPSA techniques. Then, the computational vectorial representation of the state variables and the MPFA/MPSA discrete operators are introduced. Finally, the set of discretized equations are derived in details using the governing equations from Chapter 3. In regard to the implementation aspects, we present the general implementation framework and explain the iterative-based scheme for solving the non-linear set of equations. We conclude the chapter introducing a simulator based on Automatic Differentiation for solving the Richards' equation.

Chapter 5: concentrates in the numerical validations and demonstrations of the first three parts of the thesis. Chapter 5 is subdivided into three parts, in each part one numerical validation and one numerical demonstration is covered.

Chapter 6: focuses on a numerical application of the unsaturated flow in deformable porous media. In this numerical application, we simulate the desiccation process of clayey soil in a Petri-dish. The primary motivation to study desiccation in clayey soils is the formation of cracks, which are often seen in mudcracks. The desiccation process is driven by instantaneous evaporation of water at the surface of the Petri-dish, which in turn, depends on the atmospheric conditions. Moreover, we are particularly interested in the mechanical responses within the domain, i.e., resulting displacement and stress fields. By carefully post-processing the stress field, we can identify the zones of tensile stress concentration, and consequently infer the areas where cracks are more likely to initiate.

Chapter 7: establishes the conclusions of this thesis. Also, we propose further investigations based on the results achieved in this work.

Even though this thesis has been focused on hydro-geological systems, with some modifications, applications from other fields can be studied. For example, biological systems such as tissues (Hodneland et al. 2016) and bones (Giorgio et al. 2016) have been investigated using the Biot equations.

Chapter 2

Fundamental concepts

In this chapter, we present the fundamental concepts that later will be needed to derive the equations that govern the problems considered on each part of this thesis. This chapter aims to provide the essential information related to the flow and linear elasticity problems. One of the challenges of writing this thesis was to develop a unified (and unique) nomenclature system without moving too far away from the standard nomenclature found in the literature. The chapter is divided as follows.

In section 2.1 we present the fundamental concepts related to the flow in porous media. Several concepts such as porosity, REV, saturation and water content, potentials, relative velocity and Darcy's law are formally introduced. Moreover, we pay particular attention to the unsaturated system and present the Richards' assumption, which will be crucial for the derivation of the mathematical models.

The fundamental concepts of linear elasticity are presented in section 2.2. In this section, we introduce concepts such as normal and shear stresses, the stress tensor, the traction vector, the relations between deformation, displacement and strain, the Hooke's law and the Poisson ratio. Moreover, we cover the plane stress and plane strain assumptions while remarking their scope of validity. Finally, we present other important elastic constants often found in the literature and the relations among them.

Lastly, in section 2.3 we focus on the fundamental concepts of linear poroelasticity for both, saturated and unsaturated systems. We start by introducing the concept of pore pressure. Then, we state the principle of effective stress while presenting the Biot's coefficient. Finally, using the Richards' assumption, we extend the principle of effective stress to unsaturated systems.

2.1 Concepts of unsaturated flow in porous media

In this section, we present the concepts of the unsaturated flow in porous media. We start by introducing basic concepts such porosity, representative elementary volume, and saturation. Then, we focus on the different types of potentials that rule the flow of fluids in porous media, i.e., hydraulic, pressure and elevation potentials. We proceed by defining the concept of relative velocity, which later will be of remark importance for the derivation of the mass conservation equations. Coming up next is the multiphase version of Darcy's law, which gives a quantitative measure of the water velocity as a function of its driven forces. Finally, we describe the unsaturated zone on the pore and macro scales and introduce the Richards' assumption.

2.1.1 Porosity

The fundamental property that defines a porous media is the *porosity*. The porosity n is defined as the ratio between the void space (space occupied by the fluids) V_f and the total volume of the porous medium V, which is equivalent to the sum of the void space and the volume occupied by the solid grains V_s (Pinder & Celia 2006)

$$n = \frac{V_f}{V} = \frac{V_f}{V_s + V_f}. \tag{2.1}$$

2.1.2 Representative elementary volume (REV)

The *representative elementary volume (REV)* is a concept used to describe the fluid flow in porous media. Applying volume averaging techniques the micro-scale properties of a porous media can be represented at a larger scale by introducing new properties. In essence, the REV should be sufficiently large to avoid fluctuations of the averaged properties but small enough to represent the spatial dependence of such properties (Dietrich et al. 2005). A schematic representation of the concept of REV and its dependency on a generic property, i.e porosity is shown in Fig. 2.1.

2.1.3 Saturation and water content

Whenever more than one fluid phase is filling the void space, we have a *multiphase system*, i.e., water and air. If this is the case, it is common to use the concept of *saturation* (or degree of saturation). The saturation of water is given as the ratio between the pore space which is occupied by water and the total pore volume in a representative elementary volume (REV) (Lewis & Schrefler 1998, Bear 2013, Bachmat & Bear 1987). Hence, we have

$$S_w = \frac{V_w}{V_f} = \frac{V_w}{V_w + V_a}, \tag{2.2}$$

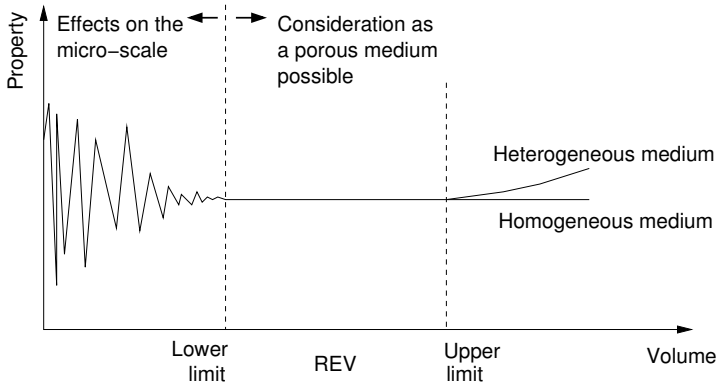


Figure 2.1 Representative elementary volume. Borrowed from *Flow and Transport in Fractured Porous Media by Peter Dietrich et al, p. 27.*

where V_w and V_a are the volumes of water and air.

Naturally, we could also define the saturation of air

$$S_a = \frac{V_a}{V_f} = \frac{V_a}{V_w + V_a}. \tag{2.3}$$

It should be evident that both saturation must add to one, i.e., $S_w + S_a = 1$. In soil mechanics and hydrogeology, the concept of *water content* (or moisture content) is often preferred over the saturation. The water content θ_w is defined as the product between the saturation of water and the porosity (Pinder & Gray 2008)

$$\theta_w = nS_w. \tag{2.4}$$

Note that the water content represents the amount of water contained in a porous material and is bounded between 0 and n.

2.1.4 Hydraulic head and fluid potential

Hubbert (1940) in his famous article *The theory of ground-water motion* defines *potential* as "a physical quantity, capable of measurement at every point in a flow system, whose properties are such that flow always occurs from regions in which the quantity has higher values to those in which it has lower, regardless of the direction in space" (p. 794). The concept of potential is crucial to the understanding of fluid flow in porous media, specially when we are dealing with gravity contributions. A detailed explanation of these concepts are beyond the scope of this thesis. However, we referred to (Freeze & Cherry 1979) for an excellent introduction.

In groundwater hydrology, it is common to work in gage pressures, i.e., setting the atmospheric pressure

equal to zero. Whenever this is the case, we can define the *water potential* Φ_w as

$$\phi_w = \frac{p_w}{\rho_w} + g\zeta, \tag{2.5}$$

where p_w is the water pressure, ρ_w is the water density, g is the modulus of the gravity acceleration [1] , and ζ is the elevation head, i.e. the height from a reference point to the measurement point (Freeze & Cherry 1979).

If we divide Eq. (2.5) by g we have

$$h_w = \frac{p_w}{\rho_w g} + \zeta = \psi_w + \zeta, \tag{2.6}$$

where h_w is the hydraulic head and ψ_w the pressure head of the water (Freeze & Cherry 1979). These concepts are schematically depicted for a laboratory manometer in Fig. 2.2.

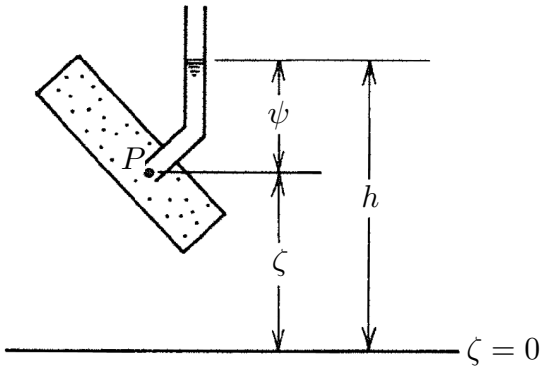


Figure 2.2 Hydraulic head h, pressure head ψ, and elevation head ζ for a laboratory manometer. Adapted from *Groundwater by R. Allan Freeze and John A. Cherry, p. 20.*

2.1.5 Relative velocity

Throughout this thesis, a material coordinate system is employed for the solid phase and a spatial coordinate system for the fluid phases, following the classical continuum mechanics approach (Lewis & Schrefler 1998). In addition, we always consider isothermal equilibrium and negligible inertial forces. Due to the convenient choice of reference systems, the water velocity can be referred to the solid phase using the relative velocity

$$\underline{v}_{ws} = \underline{v}_w - \underline{v}_s. \tag{2.7}$$

2.1.6 Darcy's law

For a non-deformable saturated porous media, the momentum conservation is often stated in the form of the Darcy's law (Darcy 1856). This law suggests a linear relationship between the fluid velocity and the

[1] To be precise, $g = \left\|\underline{g}\right\|$. However, for simplicity we avoid the formal notation.

gradients in pressure and/or elevation (Chen et al. 2006). However, to account for the simultaneous flow of two or more fluid phases, we must use the extended multiphase version of the Darcy's law

$$\underline{q}_w = nS_w\underline{v}_{ws} = -\underline{\underline{k}}\frac{k_{rw}}{\mu_w}\left(\nabla p_w - \rho_w \underline{g}\right), \tag{2.8}$$

where $\underline{q}_w$ is the Darcy's water velocity (often called specific discharge), $\underline{k}$ is the intrinsic permeability of the medium, generally a second order tensor dependent only on the porous medium (Freeze & Cherry 1979). The relative permeability k_{rw} is included to account for the simultaneous flow and it is often modeled as a function of the water saturation (Chen et al. 2006). The viscosity is represented by μ_w and $\underline{g}$ is the gravity acceleration, considered positive downwards (Lewis & Schrefler 1998).

In subsurface hydrology, it is common to express Darcy's law in terms of the hydraulic head and the *hydraulic conductivity* $\underline{\underline{K}}^w_{sat} = \underline{\underline{k}}\rho_w g/\mu_w$

$$\underline{q}_w = -\underline{\underline{K}}^w_{sat} k_{rw} \nabla h_w. \tag{2.9}$$

Some authors, recognize the product $\underline{\underline{K}}^w_{sat} k_{rw}$ as the unsaturated hydraulic conductivity $\underline{\underline{K}}_w$. Note that, both, the saturated and unsaturated hydraulic conductivity now depend not only on the properties of the porous media but the properties of the fluid as well (Pinder & Celia 2006).

2.1.7 The unsaturated zone

Groundwater is often extracted from aquifers, where all the available pore space is filled with water. Nevertheless, there are certain zones where water and air coexist, those zones are referred as unsaturated (or vadose) zones (see Fig. 2.3a). Unlike the saturated zone, unsaturated systems are characterized by additional boundaries, which separates the different phases, i.e., water, air, and solid grains. A schematic representation of such system can be seen in Fig. 2.3b (Pinder & Celia 2006). For a fairly extensive introduction to aquifers and its respective zones, we referred to (Bear 2013).

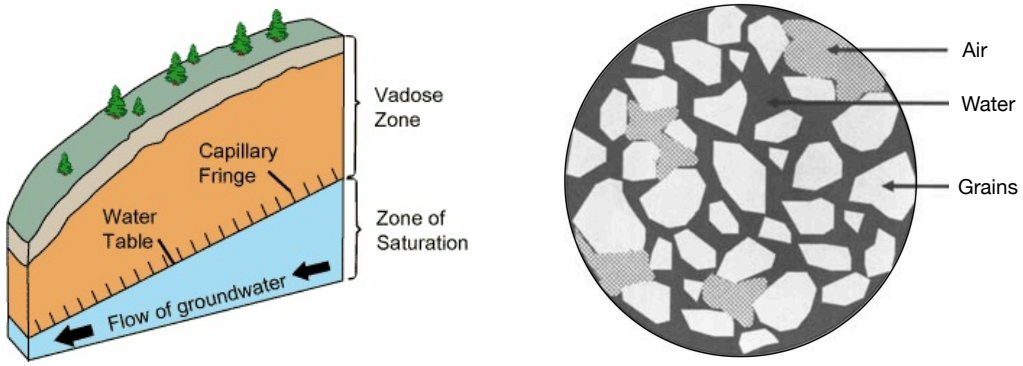


Figure 2.3 a) Cross-section of a hillslope depicting the vadose zone, capillary fringe, water table, and phreatic or saturated zone. Borrowed from *United States Geological Survey.* b) Schematic representation of a unsaturated system at the pore scale. Adapted from *Subsurface Hydrology by George F. Pinder and Michael A. Celia, p. 405.*

The unsaturated zone is relevant for several reasons, such as: transmission of water from the atmosphere to the saturated zone via infiltration or precipitation, the support of almost all kinds of plants and the active return of water from the subsurface to the atmosphere, just to mention a few of them (Pinder & Celia 2006). Among the current fields of research we can cite: colloid and colloid-facilitated transport, biogeochemical transport, multiphase flow and remediation, non-equilibrium and preferential flow, agricultural applications, constructed wetlands and inverse problems (Šimůnek & Bradford 2008).

2.1.8 The Richards' assumption

To adequately describe the simultaneous flow of water and air in a non-deformable porous media, we must use the well known two-phase flow equations (Chen et al. 2006). However, the disparity in physical properties between the water and air allow us to make some simplifications. Generally speaking, the air can be considered three orders of magnitude less dense than water, and two orders of magnitude less viscous. As the flow in a porous media is inversely proportional to the viscosity of the fluid, to obtain the same flow, the air needs a pressure gradient 100 times smaller than the water (Pinder & Gray 2008). Thus, we can assume that the air is inviscid with respect to the water. This simplification is often referred as the Richards' assumption, first proposed by Richards (1931). We must emphasize that the Richards' assumption does not imply a static air phase. On the contrary, it is infinitely mobile and can move without any significant gradients.

2.2 Concepts of linear elasticity

All types of materials describe at some extent elastic properties, i.e., if the application of external forces producing some structural deformation cease, the material restores its original configuration. In this section, we present the fundamental concepts that describe the elastic responses of a perfect solid body, i.e., the body recovers its identical original shape after the external sources of deformation disappear (Timoshenko & Goodier 1951). We start by introducing the two possible types of stresses, normal and shear. Then, we formally defined the stress tensor. A concept which is closely related to the stress, the traction vector is presented later. The next concepts cover the relationship between the deformation, displacement, and strain. After that, the link between stresses and strains is introduced progressively. We start by studying the uniaxial and shear versions of the Hooke's law and then extend them to derive the general stress-strain relationships for a three-dimensional isotropic elastic body. We also cover the particular cases where either stresses or strains can be neglected in a specific direction, giving place to the well-known plane stress and plane strain states. Finally, we present other important elastic constants and a table where these constants are related one another.

2.2.1 Normal and shear stresses

Perfect elastic bodies which are subject to external forces can be seen of as transmitting these forces from one point to another. The transmission of such forces generates internal forces, the magnitude, of these forces are characterized by the material stress. Generally speaking, we can differentiate two types of stresses, i.e., normal stress and shear stress (Lubliner & Papadopoulos 2016).

To illustrate the concept of normal stress, let us consider a small region of area ΔA, with center at c as depicted in Fig 2.4a. Moreover, let r be a characteristic dimension of this region, so that the area will vanish when r approaches to zero. Finally, let ΔF_n be the total normal force acting in this region. The *normal stress* σ_n at the point c on the cross-sectional area A is defined as

$$\sigma_n = \lim_{r \to 0} \frac{\Delta F_n}{\Delta A}. \tag{2.10}$$

An interesting way to interpret Eq. (2.10) is to see σ_n as the force per unit area acting on a small circular region as it shrinks to the point c. It is important to state that if ΔF_n is a pull force, then σ_n is called *tensile stress*. On the other hand, if ΔF_n is a push force, then σ_n is a *compressive stress* (Lubliner & Papadopoulos 2016). In Fig. 2.4b we show a normal stress field acting on the $y-z$ plane whose resultant force is F_n.

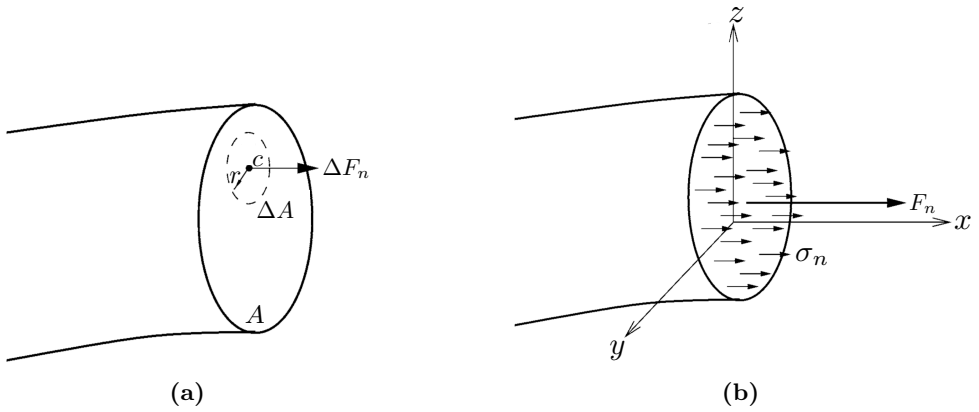


Figure 2.4 a) Force ΔF_n acting on a small region of area ΔA centered at point c on the cross section. b) Normal stress field $\sigma_n(y, z)$ and its resultant F_n. Adapted from *Introduction to solid mechanics by J. Lubliner and P. Papadopoulus, p. 157-158.*

To introduce the concept of shear stress, let us consider a shear force $\underline{F_s}$ acting on a planar cross-section of area A. Note that, we are representing the shear force as vector to reflect the fact that its direction is not specified, even though we know it is acting on the plane of cross-sectional area (see Fig. 2.5a). This is an essential difference with normal, where the direction is always known. The shear force $\underline{F_s}$ can be seen as the resultant of distributed forces per unit area acting at different points of the cross-section (Lubliner & Papadopoulos 2016). Again, we must remark that these forces do not necessarily have the same direction, as we can see in Fig. 2.5b.

Analogously to Eq. (2.10) we can define the *shear stress* $\underline{\sigma}_s$ at a point c of the cross-section as

$$\underline{\sigma}_s = \lim_{r \to 0} \frac{\Delta \underline{F}_s}{\Delta A}. \tag{2.11}$$

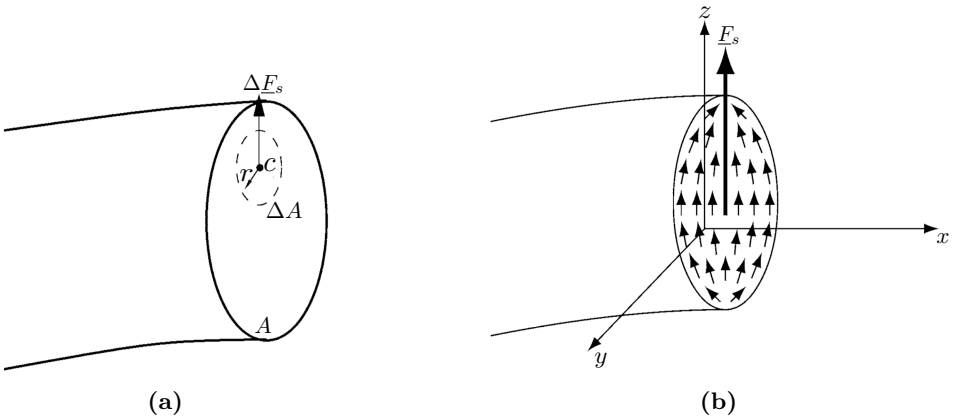


Figure 2.5 a) Force $\Delta \underline{F}_s$ acting on a circular region of area ΔA centered at point c on the cross section. b) Stresses acting on a cross section with a resultant shear force $\underline{F}_s$. Adapted from *Introduction to solid mechanics by J. Lubliner and P. Papadopoulus, p. 163-164.*

2.2.2 Stress tensor

So far, we have introduced the concepts of normal stress and shear stress, representing them on an arbitrary circular region. Let us now, properly identify the stresses that are acting on a volume element. For simplicity, we will consider a cube in Cartesian coordinates (see Fig. 2.6). It is assumed that the volume element is small enough so we can neglect any variations in the stresses to its extent. In Fig. 2.6 we show all the stresses acting on each positive faces of the cuboid. We can recognize both normal and shear stresses acting on each face. Note that two subscripts are necessary to fully describe each stress,[2] the first subscript indicates the direction of the stress vector and the second subscript refers to the direction normal to the cross-section (Lubliner & Papadopoulos 2016).

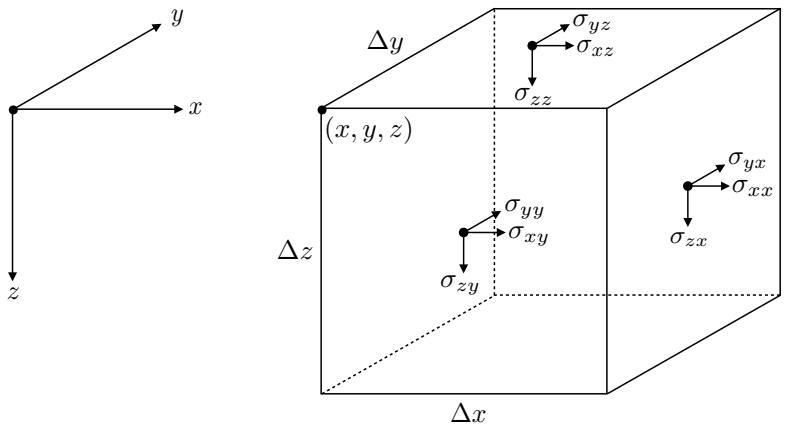


Figure 2.6 Stresses acting on a three-dimensional volume element.

[2] As a matter of fact, the normal stresses can be represented using only one subscript. Nevertheless, to avoid any inconsistency we are always going to use two subscripts.

It is important to notice that if no forces are acting on the interior of the element, the stresses on the negative faces are equal in magnitude but opposite in direction to the ones that we are showing, consequently it is said that the volume element is in force equilibrium (Lubliner & Papadopoulos 2016).

We can conveniently express all the components of the stress using matrix notation to define the *stress tensor*

$$\underline{\underline{\sigma}} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}. \tag{2.12}$$

2.2.3 Traction vector

To introduce the concept of traction vector and lay emphasis on the tensorial nature of the stress, let us consider a two-dimensional wedge in the $x-z$ plane as shown in the Fig. 2.7. We assume that the size of the wedge is infinitesimal so we can neglect any body forces that may act in it. The area of the wedge is dA, and a resultant force $\underline{T}\,dA$ is acting on the body. Note that there are both normal and shear forces acting on each leg of the wedge (Lubliner & Papadopoulos 2016).

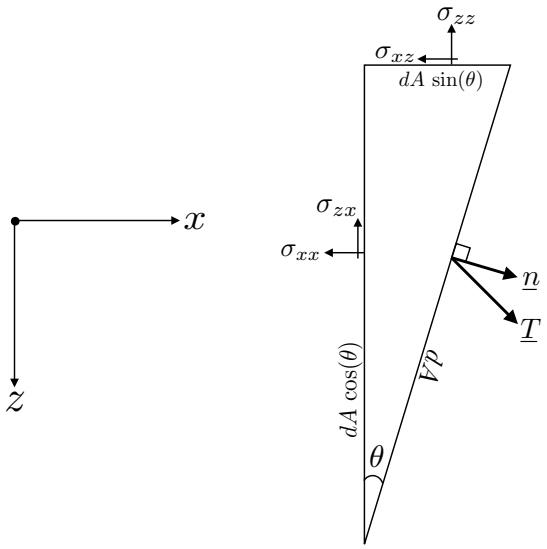


Figure 2.7 Traction vector acting on a wedge.

If this system is in force equilibrium, the forces acting on each spatial coordinate must sum to zero. That is

$$T_x dA = \sigma_{xx} dA \cos\theta + \sigma_{xz} dA \sin\theta, \tag{2.13}$$

$$T_y dA = \sigma_{zx} dA \cos\theta + \sigma_{zz} dA \sin\theta. \tag{2.14}$$

Dividing Eqs. (2.13) and (2.14) by dA and noting that $n_x = \cos\theta$ and $n_y = \sin\theta$ we have

$$T_x = \sigma_{xx} n_x + \sigma_{xz} n_y,$$
$$T_y = \sigma_{zx} n_x + \sigma_{zz} n_y.$$

The last equations can be expressed conveniently in matrix form as

$$\begin{Bmatrix} T_x \\ T_y \end{Bmatrix} = \begin{bmatrix} \sigma_{xx} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{zz} \end{bmatrix} \begin{Bmatrix} n_x \\ n_y \end{Bmatrix}.$$

This result can be generalized to a three-dimensional case with almost no effort by considering a tetrahedron instead of a planar wedge. Indeed, this generalization carries the name of *Cauchy Tetrahedron* after the famous French mathematician. In matrix form, the three-dimensional traction vector can be written as (Lubliner & Papadopoulos 2016)

$$\begin{Bmatrix} T_x \\ T_y \\ T_z \end{Bmatrix} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \begin{Bmatrix} n_x \\ n_y \\ n_z \end{Bmatrix},$$

or alternatively as

$$\underline{T} = \underline{\underline{\sigma}} \cdot \underline{n}. \tag{2.15}$$

Even though the components of the traction vector have the same units than the components of the stress, they are not stress components, since they lack a reference to the axes defined by the inclined plane (Lubliner & Papadopoulos 2016). Understanding the difference between traction vector and stress is crucial in elasticity and solid mechanics, especially while defining boundary conditions in numerical simulations.

2.2.4 Deformation, displacement and strain

Deformation can be defined as a change in the distances between material points which leads to changes in shape and/or size of a body. All materials undergo at least some deformation under the influence of forces. We can quantify the deformation of a solid relative to some reference frame by introducing the concept of strain (Lubliner & Papadopoulos 2016).

Perhaps, the easiest way to observe a deformation is by analyzing the elongation of an isotropic bar under the application of tensile forces acting at both ends as shown in Fig. 2.8 (Lubliner & Papadopoulos 2016).

The relative stretch λ can be defined as

$$\lambda = \frac{L + \Delta L}{L} = 1 + \frac{\Delta L}{L}. \tag{2.16}$$

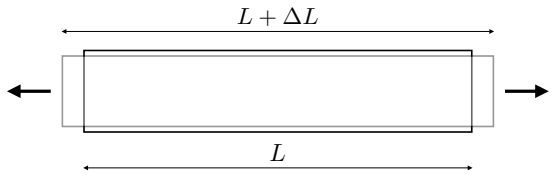


Figure 2.8 Elongation of a bar under applied tensile forces at both ends. Adapted from *Introduction to solid mechanics by J. Lubliner and P. Papadopoulus, p. 215.*

Note that λ is always a positive number, if the body suffers an elongation $\lambda > 0$, if the body suffers a contraction $\lambda < 0$ and if the material is rigid $\lambda = 1$.

If we divide the bar under consideration in Fig. 2.8 into two identical half-bars, applying the same tensile force on each of them, we will see that each bar is stretched by $\Delta L/2$ (see Fig. 2.9).

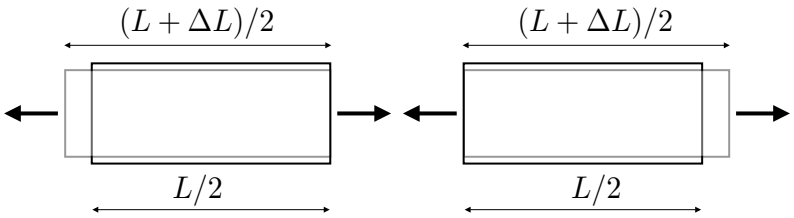


Figure 2.9 Elongation of bar represented as two-half bars in series. Adapted from *Introduction to solid mechanics by J. Lubliner and P. Papadopoulus, p. 216.*

We could progressively subdivide the original bar from Fig. 2.8 into an arbitrary number of bars in series, where the elongation ΔL will be proportional to the respective bar length. However, the elongation per unit length $\Delta L/L$ is independent of the bar length (Lubliner & Papadopoulos 2016). This quantity is usually referred to as the longitudinal strain and denoted by ε_L

$$\varepsilon_L = \frac{\Delta L}{L}. \tag{2.17}$$

If the bar lengthens $\varepsilon_L > 0$, if the bar shortens $\varepsilon_L < 0$ and if the bar is rigid $\varepsilon_L = 0$. Note that λ is easily related with ϵ_L by

$$\lambda = 1 + \varepsilon_L. \tag{2.18}$$

Unfortunately, the definition of longitudinal strain is not unique. In fact, nothing stops us from defining the strain as the ratio between ΔL and the *final* length $L' = L + \Delta L$. To avoid confusions, let us define this strain as ϵ_L

$$\epsilon_L = \frac{\Delta L}{L + \Delta L} = \frac{\varepsilon_L}{1 + \varepsilon_L}. \tag{2.19}$$

The important thing to remark here is that the difference between this two strains is small as long as the elongation ΔL is small *relative* to the original length L. In other words, if the strain satisfy

$$|\varepsilon_L| << 1. \tag{2.20}$$

If the inequality Eq. (2.20) is satisfied, we say that the strain under consideration is *infinitesimal*, otherwise, we say that we are dealing with *finite* strains. This is a crucial point in the derivation of the linear elasticity equations, where it is assumed that all bodies under consideration satisfy Eq. (2.20) (Lubliner & Papadopoulos 2016).

To consider more arbitrary deformations, we could start by comparing the positions of some point in a body before and after it has been stretched in some way as in Fig. 2.10. We represent the position of the point by a vector a located at the undeformed solid, which is moved to a new position a' in the deformed solid. We can define the *displacement* $\underline{u}$ as

$$\underline{u}(a) = a' - a, \tag{2.21}$$

which describes the motion of the point after the deformation has taken place (Goehring et al. 2015).

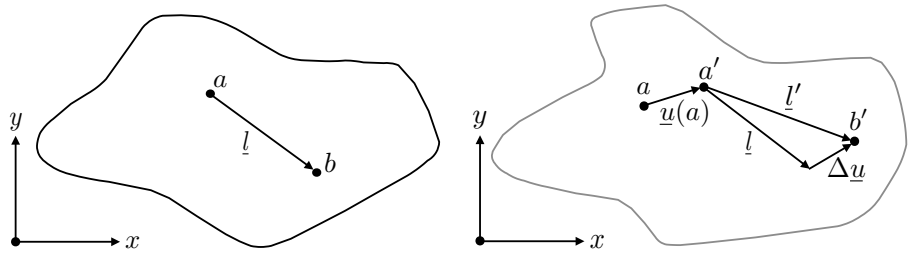


Figure 2.10 An elastic body before (left) and after (right) some deformation in the $x-y$ plane. Adapted from *Desiccation Cracks and Their Patterns by L. Goehring et. al, p. 12.*

Note that, as in the case of the stretched bar, only the relative displacement of the point is important. Again, the relative motion is captured by the strain. Let us now derive an expression that relates the strain with the displacement field as presented in (Goehring et al. 2015). Consider the points a and b in Fig. 2.10 for the undeformed solid. The points are separated by a line $\underline{l} = b - a$. Due to the application of some force, the body is deformed and these points move relative to each other, to new positions a' and b' which, in turn, are separated by a new distance $\underline{l}' = b' - a'$. The change in the line element can be determined by

$$\Delta\underline{u}(a,b) = \underline{u}(b) - \underline{u}(a) = \underline{l}' - \underline{l}. \tag{2.22}$$

Using Einstein's notation, we can represent the length L of the undeformed vector l as

$$L = \|\underline{l}\| = \sqrt{l_i l_i}. \tag{2.23}$$

Similarly, the length L' of the deformed vector $\underline{l}'$ is

$$L' = \|\underline{l}'_i\| = \sqrt{l'_i l'_i} = \sqrt{(l+\Delta u)_i (l+\Delta u)_i} = \sqrt{L^2 + 2l_i\Delta u_i + \Delta u_i \Delta u_i}. \tag{2.24}$$

Assuming an infinitesimal deformation, i.e. $\|\Delta\underline{u}\| << \|l\|$, we can Taylor-expand L' around the equilibrium length such that

$$L' = L + \frac{1}{2L}(2l_i\Delta u_i) + \mathcal{O}(\|\Delta\underline{u}\|^2) \tag{2.25}$$

Neglecting the higher order terms and rearranging the terms in Eq. (2.25) we have

$$\frac{L' - L}{L} = \frac{1}{L^2}(l_i \Delta u_i). \tag{2.26}$$

Now, we can Taylor-expand Δu_i such that

$$\Delta u_i = \frac{\partial u_i}{\partial x_j} l_j + \mathcal{O}(\|\Delta \underline{u}\|^2). \tag{2.27}$$

Invoking the symmetry argument $(\partial u_i/\partial x_j) l_i l_j = (\partial u_j/\partial x_i) l_j l_i$ we can write the explicit symmetric form of the strain tensor

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \tag{2.28}$$

or, alternatively, in vector notation

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \tag{2.29}$$

Let us now consider all the possible deformations that an elastic body can describe as presented in (Goehring et al. 2015). To do this, we start by considering a two-dimensional elastic body of unit length as shown in Fig. 2.11a.

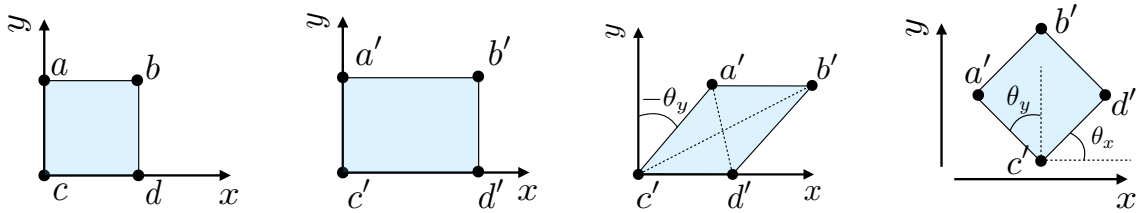


Figure 2.11 Possible deformations in a linear elastic body. **a)** Undeformed solid of unit length. **b)** Extension along the horizontal axis. **c)** Shear stress along the horizontal axis. **d)** Rotation and translation. Adapted from *Desiccation Cracks and Their Patterns by L. Goehring et. al, p. 13.*

As already discussed, a body subjected to tensile or compressive forces can stretch or shrink, respectively. In Fig. 2.11b we consider a uniaxial extension of the unit square along the horizontal axis. In this case, two edges are stretched from the equilibrium length $L = 1$ to a new length $L' = 1 + \alpha$. In the y-coordinate there is no deformation, whatsoever. Hence, the only displacement to consider is $u_{xx} = \alpha x$. The horizontal strain is therefore given by Eq. (2.28)

$$\varepsilon_{xx} = \frac{1}{2}\left(\frac{\partial u_x}{\partial x} + \frac{\partial u_x}{\partial x}\right) = \frac{1}{2}(\alpha + \alpha) = \alpha, \tag{2.30}$$

while $\varepsilon_{yy} = \varepsilon_{xy} = \varepsilon_{yx} = 0$.

Let us now, consider the case of a small shear in the horizontal direction (see Fig. 2.11c). Again, there is no displacement in the $y-$direction, hence $u_y = 0$. However, in the $x-$direction the displacement is given by $u_x = 2\varepsilon_s y$, where ε_s represents the shear strain. The only non-zero components of the strain

tensor are given by

$$\varepsilon_{xy} = \varepsilon_{yx} = \frac{1}{2}\left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}\right) = \varepsilon_s. \tag{2.31}$$

We can also determine the shear strain as a function of the angles between perpendicular axes. In Figs. 2.11c and 2.11c the angle θ_x represents the angle through which points along the horizontal axis are rotated. Similarly, θ_y represents the angle through which points along the vertical axis are rotated. Using the small angle identities we have

$$\frac{\partial u_y}{\partial x} = \tan\theta_x \approx \theta_x, \qquad \frac{\partial u_x}{\partial y} = -\tan\theta_y \approx -\theta_y. \tag{2.32}$$

Combining Eqs. (2.31) and (2.32) we arrive to an equation for a simple shear in terms of the angles θ_x and θ_y

$$\varepsilon_s = \frac{1}{2}\left(\theta_x - \theta_y\right). \tag{2.33}$$

The generalization to a three-dimensional elastic body is straightforward. In such case, the shear terms ε_{xy}, ε_{xz} and ε_{yz} are defined by the change in the angle between the corresponding pair of axes.

Lastly, whenever an elastic body is translated and/or rotated as in Fig. 2.11d, the strain tensor is always zero. Since there is no extension or shrinkage $\varepsilon_{xx} = \varepsilon_{yy} = 0$, and since $\theta_x = \theta_y$ by Eq. (2.33) we have that $\varepsilon_s = 0$.

As we can see, the components of strain tensor have a straightforward physical interpretation. The normal components ε_{xx}, ε_{yy} and ε_{zz} give us a measure of the relative distortion of the lengths with respect to the undeformed elastic body. On the other hand, the shear components ε_{xy}, ε_{xz} and ε_{yz} tell us about the relative distortion of the angles. Finally, for translation or rotation all the components of the strain tensor are zero (Goehring et al. 2015).

2.2.5 Uniaxial and shear Hooke's law

Now that we have adequately defined the concepts of stress and strain let us investigate the relationship between them. Let us start by introducing the Hooke's law for uniaxial and shear deformations.

Consider the axially loaded bar of Fig. 2.12 and assume that is linear elastic. At the left, the bar is fixed to the wall. Therefore no deformation is allowed at this end. At the right, a constant tensile force F_x is applied, elongating the bar a distance ΔL. Under this circumstances, the relative amount of elongation $\Delta L/L_0$ is proportional to the applied tensile stress F_x/A. The constant of proportionality is known as Young's modulus E of the material (Lubliner & Papadopoulos 2016). Altogether, it follows

$$\sigma_{ii} = E\varepsilon_{ii} \qquad \forall \quad i = \{x, y, z\}. \tag{2.34}$$

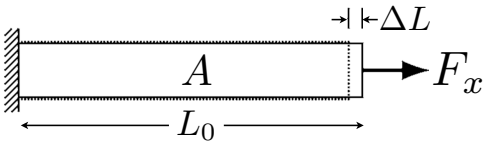


Figure 2.12 Axially loaded bar. Adapted from *Introduction to solid mechanics by J. Lubliner and P. Papadopoulus, p. 250.*

Since the strain is a non-dimensional quantity, the Young's modulus has the same units of the stress. Eq. (2.34) is referred to as the uniaxial Hooke's law. Analogously, there is a shear version of the Hooke's law that follows

$$\sigma_{ij} = 2G\varepsilon_{ij} \qquad \forall \quad i,j = \{x,y,z\}, \tag{2.35}$$

where G is known as the shear modulus (Lubliner & Papadopoulos 2016).

2.2.6 Poisson's ratio

Let us now introduce the concept of an important material parameter, the Poisson's ratio. When elastic bodies are compressed, it is common to observe that the material tends to expand in the direction perpendicular to the direction of compression. On the other hand, when elastic bodies are stretched, the material tends to contract in the directions transverse to the direction of stretching. This phenomenon is known as the Poisson effect, and the Poisson's ratio ν is a quantitative measure of this effect (Lubliner & Papadopoulos 2016).

It is formally defined as

$$\nu = -\frac{d\varepsilon_{\text{lat}}}{d\varepsilon_{\text{long}}}, \tag{2.36}$$

where ε_{lat} is the lateral (or transverse) strain, and $\varepsilon_{\text{long}}$ is the longitudinal (or axial) strain. A Poisson's ratio $\nu = 0$ means no lateral contraction. The higher the value of ν, the larger the lateral contraction. Materials with negative values of ν are quite rare, and called *auxetic*. A value of $\nu = 0.5$ denotes incompressibility.

See, for example the cube of side L showed in Fig. 2.13. The cube is stretched due to tensile forces acting along the $x-$axis, causing an increase of ΔL in the horizontal length, whereas in the y and $z-$directions the sides of the cube decrease a length $\Delta L'$.

If we assume that both, ΔL and $\Delta L'$ are small enough, the normal components of the strain tensor for the cube are given by

$$d\varepsilon_{\text{long}} = d\varepsilon_{xx} = \frac{dx}{x}, \qquad d\varepsilon_{\text{lat}} = d\varepsilon_{yy} = \frac{dy}{y}, \qquad d\varepsilon_{\text{lat}} = d\varepsilon_{zz} = \frac{dz}{z}. \tag{2.37}$$

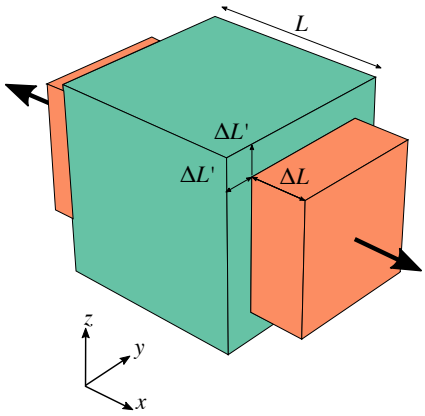


Figure 2.13 Diagram of an isotropic linear elastic material with Poisson ratio of 0.5 subject to axial forces along the x-axis only. *Downloaded under public domain from* https://commons.wikimedia.org/wiki/File:PoissonRatio.svg.

Moreover, if the Poisson's ratio is constant throughout the deformation we have

$$-\nu \int_L^{L+\Delta L} \frac{dx}{x} = \int_L^{L-\Delta L'} \frac{dy}{y} = \int_L^{L-\Delta L'} \frac{dz}{z}. \tag{2.38}$$

Computing the integrals and exponentiating we have

$$\left(1 + \frac{\Delta L}{L}\right)^{-\nu} = 1 - \frac{\Delta L'}{\Delta L}. \tag{2.39}$$

If we assume that $\Delta L << L$ and $\Delta L' << L$, we can use the property $\ln(1 \pm x) \approx \pm x \quad \forall \quad x << 1$ to have

$$\nu \approx \frac{\Delta L'}{\Delta L}. \tag{2.40}$$

Note that, this is exactly what the Poisson's ratio quantifies, the ratio between the relative deformations along the transverse and longitudinal directions.

2.2.7 Generalized Hooke's law

Now we are in position to present the general version of the Hooke's law, or in other words, the stress-strain relationship when all the components of the strain may be non-zero in a volume element.

Let us assume an *isotropic* volume element as in Fig. 2.6 (Lubliner & Papadopoulos 2016). As a consequence of Eqs. (2.34) and (2.36) the stresses σ_{xx} alone produce the strains

$$\varepsilon_{xx} = \frac{\sigma_{xx}}{E}, \qquad \varepsilon_{yy} = -\nu\frac{\sigma_{yy}}{E}, \qquad \varepsilon_{zz} = -\nu\frac{\sigma_{zz}}{E}. \tag{2.41}$$

Similarly, the stresses σ_{yy} and σ_{zz} (acting independently) will produce the strains

$$\varepsilon_{xx} = -\nu\frac{\sigma_{xx}}{E}, \qquad \varepsilon_{yy} = \frac{\sigma_{yy}}{E}, \qquad \varepsilon_{zz} = -\nu\frac{\sigma_{zz}}{E}, \tag{2.42}$$

and

$$\varepsilon_{xx} = -\nu\frac{\sigma_{xx}}{E}, \qquad \varepsilon_{yy} = -\nu\frac{\sigma_{yy}}{E}, \qquad \varepsilon_{zz} = \frac{\sigma_{zz}}{E}. \tag{2.43}$$

Invoking the principle of superposition (which allows the additivity of strains produced by stresses acting independently at each direction) we can express Eqs. (2.41) - (2.43) as

$$\begin{aligned}
\varepsilon_{xx} &= \frac{1}{E}\left(\sigma_{xx} - \nu\sigma_{yy} - \nu\sigma_{zz}\right), \\
\varepsilon_{yy} &= \frac{1}{E}\left(\sigma_{yy} - \nu\sigma_{zz} - \nu\sigma_{xx}\right), \\
\varepsilon_{zz} &= \frac{1}{E}\left(\sigma_{zz} - \nu\sigma_{xx} - \nu\sigma_{yy}\right).
\end{aligned} \tag{2.44}$$

The shear strains are given by Eq. (2.35) and it follows that

$$\begin{aligned}
\varepsilon_{xy} &= \frac{1}{2G}\sigma_{xy}, \\
\varepsilon_{yz} &= \frac{1}{2G}\sigma_{yz}, \\
\varepsilon_{zx} &= \frac{1}{2G}\sigma_{zx}.
\end{aligned} \tag{2.45}$$

We can invert Eqs. (2.44) and (2.45) as long as $\nu \neq -1 \neq 0.5$ to express the dependency between the stresses as a function of the strains

$$\begin{aligned}
\sigma_{xx} &= \frac{E}{(1+\nu)(1-2\nu)}\left[(1-\nu)\varepsilon_{xx} + \nu\varepsilon_{yy} + \nu\varepsilon_{zz}\right], \\
\sigma_{yy} &= \frac{E}{(1+\nu)(1-2\nu)}\left[(1-\nu)\varepsilon_{yy} + \nu\varepsilon_{zz} + \nu\varepsilon_{xx}\right], \\
\sigma_{zz} &= \frac{E}{(1+\nu)(1-2\nu)}\left[(1-\nu)\varepsilon_{zz} + \nu\varepsilon_{xx} + \nu\varepsilon_{yy}\right],
\end{aligned} \tag{2.46}$$

and

$$\begin{aligned}
\sigma_{xy} &= 2G\varepsilon_{xy}, \\
\sigma_{yz} &= 2G\varepsilon_{yz}, \\
\sigma_{zx} &= 2G\varepsilon_{zx}.
\end{aligned} \tag{2.47}$$

Representing Eqs. (2.46) and (2.47) as linear system in matrix notation is a rather difficult task due to the fact that stresses and strains are tensors. Alternatively, we can use a *column-matrix* notation where stresses and strains are represented as 6×1 column matrices exploiting the symmetry of the shear

components (Lubliner & Papadopoulos 2016). Using this notation, the stresses are

$$\underline{\underline{\sigma}} = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{Bmatrix},$$

while the strains[3] are

$$\underline{\underline{\varepsilon}} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ 2\varepsilon_{yz} \\ 2\varepsilon_{zx} \\ 2\varepsilon_{xy} \end{Bmatrix}.$$

In compact form, the general stress-strain relation is

$$\underline{\underline{\sigma}} = \mathfrak{C}\, \underline{\underline{\varepsilon}}, \tag{2.48}$$

where $\mathfrak{C}$ is the *stiffness tensor*, a four-rank tensor with components

$$\mathfrak{C} = \frac{E}{1+\nu} \begin{bmatrix} \frac{1-\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}.$$

On the other hand, if we are interested in the relationship between the strain as a function of the strain, we could multiply Eq. (2.48) to the left by $\mathfrak{C}^{-1}$ to get

$$\underline{\underline{\varepsilon}} = \mathfrak{C}^{-1} \underline{\underline{\sigma}}, \tag{2.49}$$

where $\mathfrak{C}^{-1}$ is often referred as *the compliance matrix* and its components are given by (Lubliner &

[3] Note that the shear strains appear multiplied by a factor of two. This is done because it is easier to express the generals stress-strain and strains-stress relations this way. Other authors use $\gamma_{ij} = 2\varepsilon_{ij}$, where γ is referred as the *engineering shear strain*.

Papadopoulos 2016)

$$\mathfrak{C}^{-1} = \frac{1}{E}\begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix}.$$

2.2.8 Plane stress and plane strain

The plane stress and plane strain states can be seen as special cases of the general stress-strain and strain-stress relationships where the z-components of the stresses and strains are zero, respectively. In other words, the plane stress assumption considers $\sigma_{zz} = \sigma_{yz} = \sigma_{zx} = 0$. Similarly, the plane strain assumption considers $\varepsilon_{zz} = \varepsilon_{yz} = \varepsilon_{zx} = 0$. The natural consequence of these assumptions is the reduction of dimensionality from three to two dimensions.

The plane stress assumption is usually valid for thin plates with stresses acting along its plane. Under this circumstances, it is assumed that there are no stresses acting perpendicular to the plane. The linear system that characterizes a plane stress state is (Lubliner & Papadopoulos 2016)

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ 2\varepsilon_{xy} \end{Bmatrix} = \frac{1}{E}\begin{bmatrix} 1 & -\nu & 0 \\ -\nu & 1 & 0 \\ 0 & 0 & 2(1+\nu) \end{bmatrix}\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix}.$$

On the other hand, a plane strain state can be idealized as a long wire with stresses acting perpendicular to the length of the wire. It is then assumed that the displacements along its length are null. Therefore, the $z-$components of the strain are zero. The linear system that characterizes a plane strain state is (Lubliner & Papadopoulos 2016)

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{Bmatrix} = \frac{E}{1+\nu}\begin{bmatrix} \frac{1-\nu}{1-2\nu} & \frac{\nu}{1-2\nu} & 0 \\ \frac{\nu}{1-2\nu} & \frac{1-\nu}{1-2\nu} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ 2\varepsilon_{xy} \end{Bmatrix}.$$

2.2.9 Relations among elastic constants

In the previous sections, we presented three elastic constants; the Young's modulus E, the shear modulus G, and the Poisson's ratio ν. However, there are other elastic constants, each of which, quantify a specific elastic response.

Let us start by introducing first and second *Lamé parameters* λ and μ (Lubliner & Papadopoulos 2016)

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \tag{2.50}$$

and

$$\mu = G. \tag{2.51}$$

Even though the first Lamé parameter lacks a straightforward physical interpretation, it is used with μ to stablish an alternative simplified formulation of the stress-strain relation known as the *Lamé formulation*. The task involves simple, but tedious algebraic transformations, which we prefer to avoid and simply state the final stress-strain relation in matrix-column notation (Lubliner & Papadopoulos 2016)

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{Bmatrix} = \begin{bmatrix} 2\mu+\lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2\mu+\lambda & \lambda & 0 & 0 & \\ \lambda & \lambda & 2\mu+\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ 2\varepsilon_{yz} \\ 2\varepsilon_{zx} \\ 2\varepsilon_{xy} \end{Bmatrix}.$$

The next physical parameter of physical relevance is the *bulk modulus* K, which characterizes the resistance of the material to volume change subject to hydrostatic load (Lubliner & Papadopoulos 2016). The bulk modulus can be expressed in terms of the Young's modulus and the Poisson's ratio as

$$K = \frac{E}{3(1-2\nu)} \tag{2.52}$$

We can see that K tends to infinity as ν approaches 0.5. In this limit, the material becomes incompressible as already we mentioned before. Note that $K = \infty$ suggests an infinite resistance of the elastic body to volume changes.

As a matter of fact, for isotropic materials, two elastic constants completely defined a stress-strain relation. The relation among them is presented in Table 2.1.

The relations in Table 2.1 are only valid for an isotropic material. In this thesis, we only consider isotropic materials. However, we referred to (Mehrabadi & Cowin 1990, Cowin & Mehrabadi 1992, Thomsen 1986) for an excellent discussion regarding anisotropic linear elasticity.

Table 2.1 Conversion formulas among elastic constants for isotropic linear materials.

	$K=$	$E=$	$\lambda=$	$G=\mu=$	ν
(K,E)	K	E	$\frac{3K(3K-E)}{9K-E}$	$\frac{3KE}{9K-E}$	$\frac{3K-E}{6K}$
(K,λ)	K	$\frac{9K(K-\lambda)}{3K-\lambda}$	λ	$\frac{3(K-\lambda)}{2}$	$\frac{\lambda}{3K-\lambda}$
(K,G)	K	$\frac{9KG}{3K+G}$	$K-\frac{2G}{3}$	G	$\frac{3K-2G}{2(3K+G)}$
(K,ν)	K	$3K(1-2\nu)$	$\frac{3K\nu}{1+\nu}$	$\frac{3K(1-2\nu)}{2(1+\nu)}$	ν
(E,λ)	$\frac{E+3\lambda+\sqrt{E^2+9\lambda^2+2E\lambda}}{6}$	E	λ	$\frac{E-3\lambda+\sqrt{E^2+9\lambda^2+2E\lambda}}{4}$	$\frac{2\lambda}{E+\lambda+\sqrt{E^2+9\lambda^2+2E\lambda}}$
(E,G)	$\frac{EG}{3(3G-E)}$	E	$\frac{G(E-2G)}{3G-E}$	G	$\frac{E}{2G}-1$
(E,ν)	$\frac{E}{3(1-2\nu)}$	E	$\frac{E\nu}{(1+\nu)(1-2\nu)}$	$\frac{E}{2(1+\nu)}$	ν
(λ,G)	$\lambda+\frac{2G}{3}$	$\frac{G(3\lambda+2G)}{\lambda+G}$	λ	G	$\frac{\lambda}{2(\lambda+G)}$
(λ,ν)	$\frac{\lambda(1+\nu)}{3\nu}$	$\frac{\lambda(1+\nu)(1-2\nu)}{\nu}$	λ	$\frac{\lambda(1-2\nu)}{2\nu}$	ν
(G,ν)	$\frac{2G(1+\nu)}{3(1-2\nu)}$	$2G(1+\nu)$	$\frac{2G\nu}{1-2\nu}$	G	ν

2.3 Concepts of linear poroelasticity for saturated and unsaturated systems

In this last section, we are going to present the concepts of linear poroelasticity. Many of the concepts presented in the previous section will still hold. However, some important distinctions must be made since now we are considering a porous material rather than a solid body. Since the essential characteristic of a porous medium is its ability to transmit and/or store a fluid, the interaction between the mechanical effects and the flow of the fluid must be taken into account (Merxhani 2016, Verruijt 2018). In this section, we start by introducing the concept of pore pressure. Then, we present the principle of effective stress while introducing the Biot's coefficient. Finally, we extend the principle of effective stress applied to saturated systems, to unsaturated systems.

2.3.1 Pore pressure

Typically, in a porous media, all the pores are mutually connected. Let us now assume that a fluid (usually water) is filling the pore space. When external or internal forces are acting on the porous media, a pressure is transmitted in the water body, possibly causing the water to flow through the pores. The pressure in the pores is usually denoted as *pore pressure* (Verruijt 2013).

2.3.2 The principle of effective stress and the Biot's coefficient

When stresses are applied to a saturated porous medium, some part act on the solid skeleton and some in the pore fluid. The stresses that act on the solid skeleton are responsible for the deformation of the

porous medium, therefore the name *effective* (Merxhani 2016). If we consider that tensile stresses are positive, and compressive stresses are negative, the principle of effective stress can be written as

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}}_e - \alpha p_w \underline{\underline{I}}, \tag{2.53}$$

where $\underline{\underline{\sigma}}_t$ is the total stress, $\underline{\underline{\sigma}}_e$ is the effective stress, p_w is the *pore* water pressure and $\underline{\underline{I}}$ is the identity matrix. The parameter α is known as the Biot's coefficient and it was first introduced by Biot (1941) as

$$\alpha = \frac{K}{H}, \tag{2.54}$$

where K is the bulk modulus (as in Eq. (2.52)) under drained conditions (the water is allow to escape through the boundaries of the porous medium), and $1/H$ is the poroelastic expansion coefficient, which describes the change of the *bulk volume* due to a pressure keeping the stress constant. While Eq. (2.54) is completely valid, it is somehow difficult to measure K and $1/H$. An alternative expression for α was proposed by Biot & Willis (1957)

$$\alpha = 1 - \frac{C_s}{C_m}, \tag{2.55}$$

where C_s and C_m are the compressibility of the solid grains and porous medium, respectively.

For soft soils, the value of α can be considered equal to one, reducing Eq.(2.53) to

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}}_e - p_w \underline{\underline{I}}. \tag{2.56}$$

Eq. (2.56) was used in the original work of von Terzaghi (1943) and is valid for most soils, since usually their skeleton is highly compressible, while their grains are virtually incompressible (Merxhani 2016). The Biot's coefficient can not take arbitrary values, it is bounded between $n \leq \alpha \leq 1$.

We must remark that the effective stress should not be confused with the average stress that is acting on the grains, which is obtained by subtracting the average pore pressure $-np_w\underline{\underline{I}}$ from $\underline{\underline{\sigma}}$ (Verruijt 2018). The stress-strain and strain-stress relations derived in the previous section for an isotropic solid are still completely valid. However they must be applied to the effective stress rather than to the total stress (Detournay & Cheng 1995).

2.3.3 The principle of effective stress for unsaturated systems

For unsaturated systems, we must consider the presence of air in the system. Under this premise, the pore pressure has now two contributions, the water pressure p_w and the air pressure p_a. Eq. (2.53) can be easily extended by considering each of these contributions (Lewis & Schrefler 1998)

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}}_e - \alpha \left[S_w p_w + (1 - S_w) p_a\right] \underline{\underline{I}}. \tag{2.57}$$

Nevertheless, if we use Richards' assumption of inviscid air presented in section 2.1.8, the air pressure

is constant and equal to the atmospheric pressure, which in turn, it is set to zero[4]. Hence, Eq. (2.57) simplifies to

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}}_e - \alpha S_w p_w \underline{\underline{I}}. \tag{2.58}$$

Eq. (2.58) clearly suggests that the pore water contribution is reduced by the presence of air by a factor of S_w. Note that the Richards' assumption now have an impact, not only on the flow, but also on the mechanical responses by assuming that air does not contribute to the total stress. Finally, it is obvious that we recover Eq. (2.53) when $S_w = 1$. Therefore, Eq. (2.58) can be seen as a generalization of the principle of effective stress.

[4]The practice of using the atmospheric pressure as a reference pressure is quite usual when dealing with unsaturated systems. The natural consequence of using gauge pressures instead of absolute pressures is that $p_w < 0$ if $S_w < 1$ and $p_w \geq 0$ if $S_w = 1$.

Chapter 3

Mathematical models and governing equations

In this chapter, we focus on the mathematical modeling and derivation of the governing equations for each part of this thesis. We aim to present the derivations as structured and detailed as possible. In view that some parts of this thesis are subsets (or particular cases) of others, we try our best to not fall into repetitiveness without compromising the clarity of the derivations. When we derive an important equation, a short explanation of the physical meaning of each term is given. At the end of each section, we present the *complete set of equations* for each of the problems under consideration while making a short analysis of the degrees of freedom (number of unknowns *vs.* number of equations) in order to prove the closeness of the set of equations. The chapter is structured as follows.

In section 3.1 we derive the three-dimensional, incompressible form of the Richards' equation. We start from the general statement of mass conservation and later combine it with the multiphase version of Darcy's law to derive the three versions of the Richards' equation, i.e., the pressure head, water content and mixed based form. Finally, we present the water retention curves that later are going to be used in the simulations.

The equations that govern the three-dimensional linear elasticity equations are derived in section 3.2. We present a detailed derivation of the equilibrium equations by performing a force and moment balance in an infinitesimal parallelepiped. The resulting equations are closed with the compatibility and stress-strain relations for an isotropic elastic body presented in the previous chapter.

We present the derivation of the governing equations for the saturated flow in deformable porous media in section 3.3. We start by extending the equilibrium equations to the case where solid particles and water interact mechanically within a porous medium by invoking the principle of effective stress. Later, we focus on the mass conservation equation, where we derive an expression for both, the solid and fluid phases. Finally, we derive with details the storage equation, which combines the two previous equations into one.

Using the Richards' assumption, in section 3.4 we present the governing equations that rule the unsaturated flow in deformable porous media. The derivation is quite similar to the saturated flow in deformable porous media. Therefore, we emphasize the differences whenever is needed, especially while deriving the unsaturated storage equation.

3.1 Equations for unsaturated flow in non-deformable porous media

In this section, we are going to present the equation that governs the incompressible-unsaturated flow in non-deformable porous media under the Richards' assumption. The Richards' assumption allow us to disregard the equations for the air by assuming inviscid conditions and consequently a constant air pressure equal to the atmospheric. We start our analysis by invoking the general statement of the water conservation. By combining the multiphase Darcy's law with the mass balance equation, we can derive the three versions of the incompressible Richards' equations, i.e., the mixed-based form, the pressure head based form and the water content based form. We present a short discussion of the advantages and disadvantages in the use of each of these versions. Finally, we show the water retention curves, which model the dependency of the water content, relative permeability and specific moisture capacity as a function of the water pressure head.

3.1.1 Mass balance equation

The mass balance equation of a fluid (in this case water) expresses the conservation of mass within the REV (Bear 2013). Can be stated generally as

$$\frac{\partial \left(\rho_w S_w n\right)}{\partial t} + \nabla \cdot \left(\rho_w S_w n \underline{v}_w\right) = \dot{m}_w, \tag{3.1}$$

where $\dot{m}_w$ is the rate of external addition (or subtraction) of mass of fluid per volume of REV.

Assuming that the porous media, as well as the water, are incompressible, we can represent the first term of Eq. (3.1) as

$$\frac{\partial \left(\rho_w S_w n\right)}{\partial t} = n\rho_w \frac{\partial S_w}{\partial t}. \tag{3.2}$$

Using the definition of the relative velocity (see Eq. (2.7)) and a vector identity[5] we can expand the second term of Eq. (3.1)

$$\begin{aligned} \nabla \cdot \left(\rho_w S_w n \underline{v}_w\right) &= \nabla \cdot \left(\rho_w S_w n \underline{v}_{ws}\right) + \nabla \cdot \left(\rho_w S_w n \underline{v}_s\right) \\ &= n S_w \underline{v}_{ws} \nabla \cdot \rho_w + \rho_w \nabla \cdot \left(S_w n \underline{v}_{ws}\right) \\ &= \rho_w \nabla \cdot \left(S_w n \underline{v}_{ws}\right). \end{aligned} \tag{3.3}$$

Note that in Eq. (3.3) we discarded the terms containing the solids velocity due to our assumption of non-deformable porous media. Moreover, we neglected the term expressing the gradient of the water density. The argument behind this assumption is based on the fact that water density does not change considerably in space and even so, its value is negligible compared to the divergence of the water velocity

[5] For a complete summary of vector identities we referred to (Bird 2002)

(Pinder & Celia 2006).

Using Eqs. (3.2) - (3.3) we can rewrite Eq. (3.1) as

$$n\rho_w \frac{\partial S_w}{\partial t} + \rho_w \nabla \cdot (S_w n \underline{v}_{ws}) = \dot{m}_w. \tag{3.4}$$

Recognizing the term $S_w n \underline{v}_{ws}$ as the Darcy velocity of the water $\underline{q}_w$ presented in Eq. (2.8) and dividing Eq. (3.4) by ρ_w we have

$$n\frac{\partial S_w}{\partial t} + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \tag{3.5}$$

Eq. (3.5) is a statement of the principle of mass conservation of water in an unsaturated porous medium in the case where the water and the porous medium are considered incompressible.

3.1.2 The Richard's equation and its different formulations

The Richards' equation is based upon the assumption of inviscid air. This allow us to consider the air pressure as constant and equal to the atmospheric pressure. Furthermore, if we consider both, the porous medium and the water as incompressible we can use Eq. (3.5) together with Eqs. (2.9) and (2.6) to express the mass conservation as

$$n\frac{\partial S_w}{\partial t} - \nabla \cdot \left(\underline{\underline{K}}^w_{sat} k_{rw} \left(\nabla \psi_w + \nabla \zeta\right)\right) = \frac{\dot{m}_w}{\rho_w}. \tag{3.6}$$

Instead of using the saturation, we can use the water content to rewrite Eq.(3.6) as (Pinder & Celia 2006)

$$\frac{\partial \theta_w}{\partial t} - \nabla \cdot \left(\underline{\underline{K}}^w_{sat} k_{rw} \left(\nabla \psi_w + \nabla \zeta\right)\right) = \frac{\dot{m}_w}{\rho_w}. \tag{3.7}$$

Eq. (3.7) is referred to as the incompressible *mixed form* of the Richards' equation. The "mixed form" implies that both, the moisture content and the water pressure head appear as primary variables (Pinder & Celia 2006). If we expand the water content as a function of the pressure head we have

$$\frac{\partial \theta_w}{\partial t} = \frac{d\theta_w}{d\psi_w}\frac{\partial \psi_w}{\partial t} = C(\psi_w)\frac{\partial \psi_w}{\partial t}, \tag{3.8}$$

where $C(\psi_w)$ is the *specific moisture capacity*.

Combining Eq. (3.8) and Eq. (3.7) we can write the mixed form of the Richards' equation entirely in terms of the pressure head

$$C(\psi_w)\frac{\partial \psi_w}{\partial t} - \nabla \cdot \left(\underline{\underline{K}}^w_{sat} k_{rw} \left(\nabla \psi_w + \nabla \zeta\right)\right) = \frac{\dot{m}_w}{\rho_w}. \tag{3.9}$$

Eq. (3.9) is called the incompressible *pressure head-based form* of the Richards' equation (Pinder & Celia 2006). Similarly, if instead of expanding the water content as a function of the pressure head, we express the gradient of the pressure head as a function of the water content we have

$$\nabla\psi_w = \frac{d\psi_w}{d\theta_w}\nabla\theta_w. \tag{3.10}$$

Combining Eqs. (3.10) and (3.7) we can express the Richards' equation solely in terms of the water content

$$\begin{aligned}
&\frac{\partial\theta_w}{\partial t} - \nabla\cdot\left[\underline{\underline{K}}^w_{sat}k_{rw}\left(\frac{d\psi_w}{d\theta_w}\nabla\theta_w + \nabla\zeta\right)\right] \\
&= \frac{\partial\theta_w}{\partial t} - \nabla\cdot\left[\left(\frac{\underline{\underline{K}}^w_{sat}k_{rw}}{d\theta_w/d\psi_w}\right)\nabla\theta_w + \underline{\underline{K}}^w_{sat}k_{rw}\nabla\zeta\right] \\
&= \frac{\partial\theta_w}{\partial t} - \nabla\cdot\left[D(\theta_w)\nabla\theta_w\right] - \nabla\cdot\left(\underline{\underline{K}}^w_{sat}k_{rw}\right)\nabla\zeta = \frac{\dot{m}_w}{\rho_w}.
\end{aligned} \tag{3.11}$$

The term $D(\theta_w)$ is called the *soil moisture diffusivity*. We refer Eq. (3.11) to as the *moisture content-based* form of the incompressible Richards' equation (Pinder & Celia 2006).

The three different forms of the Richards' equation have their advantages and disadvantages that are worth mentioning. For example, the θ-based is restricted only to the unsaturated zone. In the saturated zone, the water content equals the porosity and the driven force for the flow disappears. Hence, if saturated zones are a possible scenario, the ψ-based or mixed form should be used. Regarding complexity, it is evident that the mixed form is more challenging to implement than the ψ-based or θ-based forms. However, as we are going to see in the next chapter, the ψ-based form fails to conserve the mass. In a few cases, some noticeable errors have been reported. Due to this issue, the mixed form of the Richards' equation is the most reliable and currently preferred over the two other forms (Pinder & Celia 2006, Pinder & Gray 2008).

3.1.3 Water retention curves

So far, we have not mentioned how to determine the water content nor the relative permeability. In practice, they are determined via laboratory experiments for each specific type of soil in controlled conditions. However, some empirical correlations can be used to express a straightforward analytical dependency on the pressure head. These empirical correlations are known as *water retention curves*. The most used correlation is attributed to Van Genuchten (1980).

The water content is expressed as a function of the pressure head as

$$\theta(\psi_w) = \begin{cases} \dfrac{\theta^s_w - \theta^r_w}{\left[1 + (\alpha_v|\psi_w|)^{n_v}\right]^{m_v}} & \psi_w < 0 \\ \theta^w_s & \psi_w \geq 0 \end{cases} \tag{3.12}$$

where θ_w^s is the water content at saturated conditions, θ_w^r is the residual water content and α_v, n_v, m_v are fitting parameters that depend on the type of soil.

The relative permeability can also be calculated as a function of the pressure head

$$k_{rw}(\psi_w) = \begin{cases} \dfrac{\left\{1 - (\alpha_v|\psi|)^{n_v-1}\left[1 + (\alpha_v|\psi|)^{n_v}\right]^{-m_v}\right\}^2}{\left[1 + (\alpha_v|\psi|)^{n_v}\right]^{m_v/2}} & \psi_w < 0 \\ 1 & \psi_w \geq 0 \end{cases} \tag{3.13}$$

Finally, we can compute the derivative of the water content with respect to the pressure head to determine the specific moisture capacity

$$C(\psi_w) = \frac{\partial \theta_w}{\partial \psi_w} = \begin{cases} -\dfrac{m_v n_v \psi_w \left(\theta_w^s - \theta_w^r\right)(\alpha_v|\psi_w|)^{n_v}}{|\psi_w|^2\left[(\alpha_v|\psi_w|)^{n_v} + 1\right]^{m_v+1}} & \psi_w < 0 \\ 0 & \psi_w \geq 0 \end{cases} \tag{3.14}$$

As an example, we show in Fig. 3.1 the standard water retention curves for the data used in Celia et al. (1990) where $\alpha_v = 0.0335\,\mathrm{cm}^{-1}$, $\theta_w^s = 0.368$, $\theta_w^r = 0.102$, $n_v = 2$, $m_v = 0.5$ for a pressure head ranging from $-1000\,\mathrm{cm}$ to $-75\,\mathrm{cm}$.

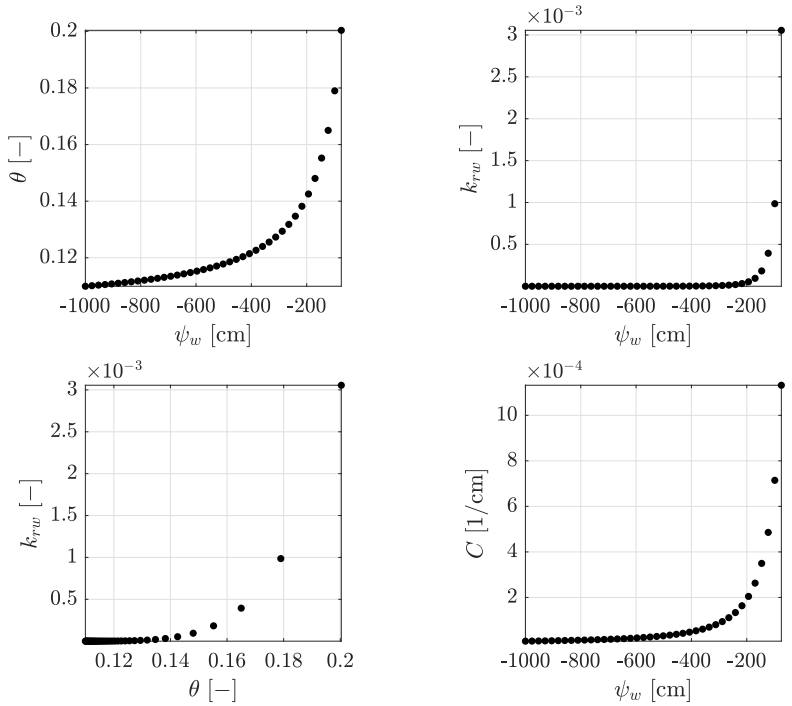


Figure 3.1 Water retention curves showing the interdependency between θ_w, k_{rw}, C and ψ_w.

3.2 Equations of linear elasticity

In this section, we present the derivation of the governing equations of linear elasticity. We start by deriving the local equilibrium equations for an infinitesimal parallelepiped. We consider the classical Taylor-expansion of stresses to approximate the stresses on opposite faces. Then, we perform a force balance within the infinitesimal element where normal stresses, shear stresses, and body density forces are taken into account. In view that, for a static system, not only the sum of forces, but also the moments must be zero, we perform a moment balance whose natural consequence results in the symmetry of the stress tensor. Finally, we present the complete set of equations that fully defines a three-dimensional linear elastic problem.

3.2.1 Local equilibrium equations

The local equilibrium equations can be seen as a particular case of the principle of momentum conservation for a static deformations. In this section, we are going to derive the general expression for a three-dimensional solid as in (Lubliner & Papadopoulos 2016). Let us consider a rectangular parallelepiped with its origin at $(\bar{x}, \bar{y}, \bar{z}$ as shown in Fig. 3.2).

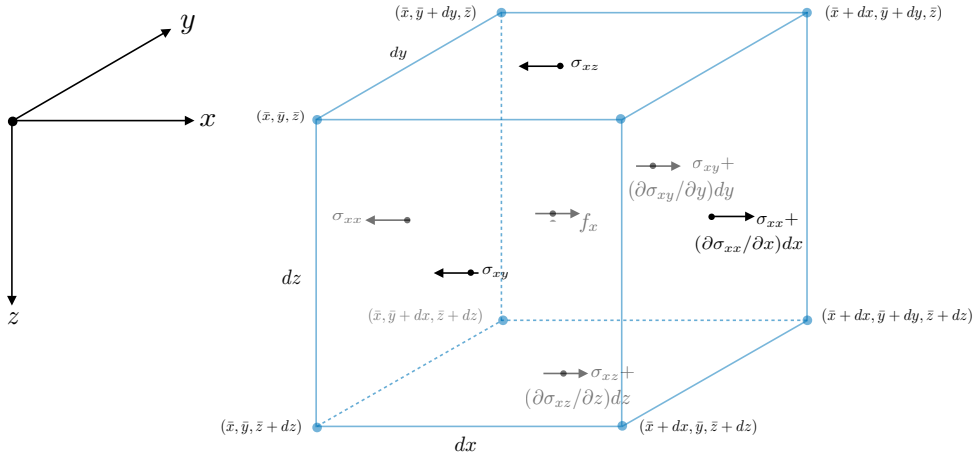


Figure 3.2 Free-body diagram for an infinitesimal rectangular parallelepiped with its origin in $(\bar{x}, \bar{y}, \bar{z})$

Let us suppose that a normal stress σ_{xx} is acting at a point $(\bar{x}, y, z)$ on the face with coordinates $x = \bar{x}$. Employing a Taylor expansion around $x = \bar{x}$ we can express the stress acting on the opposite face $x = \bar{x} + \Delta x$ as

$$\sigma_{xx}(\bar{x} + \Delta x, y, z) = \sigma_{xx}(\bar{x}, y, z) + \frac{\partial \sigma_{xx}}{\partial x}(\bar{x}, y, z)\Delta x + \frac{1}{2!}\frac{\partial^2 \sigma_{xx}}{\partial x^2}(\bar{x}, y, z)\Delta x^2 + ... \tag{3.15}$$

Taking the limit as Δx becomes zero, we can replace Δx with dx. Moreover, neglecting the higher order terms, we can express Eq. (3.15) as

$$\sigma_{xx}(\bar{x} + dx, y, z) = \sigma_{xx}(\bar{x}, y, z) + \frac{\partial \sigma_{xx}}{\partial x}(\bar{x}, y, z)dx. \tag{3.16}$$

A similar argument can be made for the shear stresses σ_{xz} and σ_{xy}, where

$$\sigma_{xz}(x, y, \bar{z} + dz) = \sigma_{xz}(x, y, \bar{z}) + \frac{\partial \sigma_{xz}}{\partial z}(x, y, \bar{z})dz, \tag{3.17}$$

and

$$\sigma_{xy}(x, \bar{y} + dy, z) = \sigma_{xy}(x, \bar{y}, z) + \frac{\partial \sigma_{xy}}{\partial y}(x, \bar{y}, z)dy. \tag{3.18}$$

The forces acting on each face are obtained by multiplying the stresses by their respective areas. Moreover, we consider an internal density force f_x acting on the centroid of the element. Since we are not allowing any kind of motion of the body, we can set the sum of all the forces acting along the x-direction equal to zero

$$\left[\sigma_{xx} + \frac{\partial \sigma_{xx}}{\partial x}dx\right](dydz) + \left[\sigma_{xy} + \frac{\partial \sigma_{xy}}{\partial y}dy\right](dxdz) + \left[\sigma_{xz} + \frac{\partial \sigma_{xz}}{\partial z}dz\right](dxdy) + f_x(dxdydz) -$$
$$\sigma_{xx}(dydz) - \sigma_{xy}(dxdz) - \sigma_{xz}(dxdy) = 0. \tag{3.19}$$

Cancelling terms and dividing by $dxdydz \neq 0$, Eq. (3.19) becomes

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} + f_x = 0. \tag{3.20}$$

Analogously, for the y and z-directions we have

$$\frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} + f_y = 0, \tag{3.21}$$

and

$$\frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + f_z = 0. \tag{3.22}$$

The second requirement for a body to be in static equilibrium is that the sum of moments acting at all points of the body must be zero. Note that only the shear stresses could induce a rotation or spin and, since the sum of forces is zero, we could choose arbitrarily any point, such as $(\bar{x} + dx, \bar{y} + dy, \bar{z})$. The moment about this point will be

$$\sigma_{yx}(dydz)dx - \sigma_{xy}(dxdy)dy = 0. \tag{3.23}$$

Hence, we have that

$$\sigma_{xy} = \sigma_{yx}. \tag{3.24}$$

A similar argument can be made for the components of the shear stress, where

$$\sigma_{xz} = \sigma_{zx}, \tag{3.25}$$

and

$$\sigma_{yz} = \sigma_{zy}. \tag{3.26}$$

The natural consequence of Eqs. (3.24)-(3.26) is that the stress tensor $\underline{\underline{\sigma}}$ is symmetric (Lubliner & Papadopoulos 2016). If we consider that the only internal force acting on the system is caused by the weight of the solid body, the internal density forces takes the form

$$\underline{f} = \rho_s \underline{g}, \tag{3.27}$$

where ρ_s is the density of the solid phase.

3.2.2 The complete set of equations

The equations of linear elasticity can be summarized by the following governing equations written in compact notation

- ◇ Equilibrium equations

$$\nabla \cdot \underline{\underline{\sigma}} + \rho_s \underline{g} = 0. \tag{3.28}$$

- ◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \tag{2.29 revisited}$$

- ◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}} = \mathfrak{C}\, \underline{\underline{\varepsilon}}. \tag{2.48 revisited}$$

Note that the total number of unknowns for a three-dimensional problem is 21 (9 stresses, 9 strains and 3 displacements) and the total number of equations is also 21 (6 equilibrium equations, 9 compatibility equations and 6 independent stress-strain relations).

3.3 Equations of saturated flow in deformable porous media

The equations that govern the saturated flow in deformable porous media are a combination of the linear elasticity equations and the mass conservation equation applied for both, the solid skeleton and the fluid. In this section, we will present a detailed derivation of the later. However, to avoid redundancy, we are simply going to state the final equations for the mechanics problem, since the only difference is that, instead of consider one single stress we must write the equations in terms of the total stress, which includes not only the contribution of the solid skeleton but also the fluid.

3.3.1 Linear momentum balance equation

The equilibrium equations applied to a saturated porous medium is given by (Lewis & Schrefler 1998)

$$\nabla \cdot \underline{\underline{\sigma}}_t + \underline{f} = 0. \tag{3.29}$$

As mentioned before, $\underline{\underline{\sigma}}_t$ represents the total stress (which now has two contributions) as presented in section 2.3.2

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}}_e - \alpha p_w \underline{\underline{I}}. \tag{2.53 revisited}$$

In virtually all the problems in porous medium, the internal density force $\underline{f}$ can be treated as a body (density) force, which in turn, has the contribution of the weight of the solid skeleton and the water

$$\underline{f} = [(1-n)\rho_s + n\rho_w]\,\underline{g}. \tag{3.30}$$

Combining Eqs. (3.29), (2.53), (3.30) and assuming α as constant, we have

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla p_w + [(1-n)\rho_s + n\rho_w]\,\underline{g} = 0. \tag{3.31}$$

Eq. (3.31) written in this way, clearly shows, that changes in momentum can be caused by three different mechanisms, the external forces acting on the boundaries (normal and shear *effective* stresses), the pore pressure exerted by the water, and the combined weight of the solid skeleton and the water.

The compatibility equations and the stress-strain relation can be written as

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla u + (\nabla u)^T\right), \tag{3.32}$$

and

$$\underline{\underline{\sigma}}_e = \mathfrak{C}\,\underline{\underline{\varepsilon}}. \tag{3.33}$$

3.3.2 Mass conservation

In this section, we are going to present a detailed derivation of the mass conservation for the saturated flow in deformable porous media as in (Lewis & Schrefler 1998). This is done by combining the mass conservation equation for the solid and fluid phase into another equation, called the *storage equation*. The storage equation together with Darcy's law, provide all the necessary information to close the system of equations.

Mass conservation equation for the fluid phase

The general mass conservation equation for an unsaturated was already presented in section 3.1.1. In the particular case when $S_w = 1$, Eq. (3.1) becomes

$$\frac{\partial\left(\rho_w n\right)}{\partial t} + \nabla\cdot\left(\rho_w n\underline{v}_w\right) = \dot{m}_w. \tag{3.34}$$

The accumulation term in Eq. (3.34) can be expanded using the chain rule as

$$\frac{\partial\left(\rho_w n\right)}{\partial t} = n\frac{\partial\rho_w}{\partial t} + \rho_w\frac{\partial n}{\partial t}. \tag{3.35}$$

Using the definition of the relative velocity (Eq. (2.7)) and neglecting the terms $\nabla\rho_w$ and $\nabla(\rho_w n)$ we can write the second term in Eq. (3.34) as

$$\begin{aligned}\nabla\cdot\left(\rho_w n\underline{v}_w\right) &= \nabla\cdot\left(\rho_w n\underline{v}_{ws}\right) + \nabla\cdot\left(\rho_w n\underline{v}_s\right),\\ &= \nabla\rho_w\cdot\left(n\underline{v}_s\right) + \rho_w\nabla\cdot\left(n\underline{v}_{ws}\right) + \nabla(\rho_w n)\cdot\underline{v}_s + \rho_w n\nabla\cdot\underline{v}_s,\\ &= \rho_w\nabla\cdot\left(n\underline{v}_{ws}\right) + \rho_w n\nabla\cdot\underline{v}_s.\end{aligned} \tag{3.36}$$

Assuming slight compressibility of the water, the time derivative of the water density can be written as

$$\frac{\partial\rho_w}{\partial t} = \frac{d\rho_w}{dp_w}\frac{\partial p_w}{\partial t} = \rho_w C_w\frac{\partial p_w}{\partial t}, \tag{3.37}$$

where C_w is the compressibility of the water.

Combining Eqs. (3.34) - (3.37) we have

$$C_w\rho_w n\frac{\partial p_w}{\partial t} + \rho_w\frac{\partial n}{\partial t} + \rho_w\nabla\cdot\left(n\underline{v}_{ws}\right) + \rho_w n\nabla\cdot\underline{v}_s = \dot{m}_w. \tag{3.38}$$

Dividing Eq. (3.38) by ρ_w gives

$$C_w n \frac{\partial p_w}{\partial t} + \frac{\partial n}{\partial t} + \nabla \cdot (n \underline{v}_{ws}) + n \nabla \cdot \underline{v}_s = \frac{\dot{m}_w}{\rho_w}. \tag{3.39}$$

We recognize the term $n\underline{v}_{ws}$ as the Darcy's velocity of the water (see section 2.1.6)

$$n \underline{v}_{ws} = \underline{q}_w = -\frac{\underline{\underline{k}}}{\mu_w} \left(\nabla p_w - \rho_w \underline{g} \right). \tag{3.40}$$

Making use of Eq. (3.40) the equation of water mass conservation can be written as

$$C_w n \frac{\partial p_w}{\partial t} + \frac{\partial n}{\partial t} + \nabla \cdot \underline{q}_w + n \nabla \cdot \underline{v}_s = \frac{\dot{m}_w}{\rho_w}. \tag{3.41}$$

Mass conservation equation for the solid phase

The general mass conservation equation for the solid phase follows

$$\frac{\partial \left[(1-n) \rho_s \right]}{\partial t} + \nabla \cdot \left[(1-n) \rho_s \underline{v}_s \right] = 0. \tag{3.42}$$

Note that the source term is equal to zero since no solid particles can enter or leave the system. The first term of Eq. (3.42) can be written as

$$\begin{aligned} \frac{\partial \left[(1-n) \rho_s \right]}{\partial t} &= (1-n) \frac{\partial \rho_s}{\partial t} + \rho_s \frac{\partial (1-n)}{\partial t}, \\ &= (1-n) \frac{\partial \rho_s}{\partial t} - \rho_s \frac{\partial n}{\partial t}. \end{aligned} \tag{3.43}$$

Neglecting the term $\nabla \left[(1-n) \rho_s \right]$, the second term of Eq. (3.42) is

$$\begin{aligned} \nabla \cdot \left[(1-n) \rho_s \underline{v}_s \right] &= \nabla \left[(1-n) \rho_s \right] \cdot \underline{v}_s + \left[(1-n) \rho_s \right] \nabla \cdot \underline{v}_s, \\ &= \rho_s \nabla \cdot \underline{v}_s - \rho_s n \nabla \cdot \underline{v}_s. \end{aligned} \tag{3.44}$$

Combining Eqs. (3.42) - (3.44) and dividing by ρ_s we get

$$\frac{(1-n)}{\rho_s} \frac{\partial \rho_s}{\partial t} - \frac{\partial n}{\partial t} + \nabla \cdot \underline{v}_s - n \nabla \cdot \underline{v}_s = 0. \tag{3.45}$$

The storage equation

Now we are in position to combine the expressions of mass conservation for the fluid and solid phases. We start by adding Eqs. (3.41) and (3.45) in order to eliminate the time derivative of the porosity

$$C_w n \frac{\partial p_w}{\partial t} + \nabla \cdot \underline{q}_w + \frac{(1-n)}{\rho_s}\frac{\partial \rho_s}{\partial t} + \nabla \cdot \underline{v}_s = \frac{\dot{m}_w}{\rho_w}. \tag{3.46}$$

The density of the solid can be expressed as a function of the isotropic total stress $\sigma_t := \mathbf{Tr}\left(\underline{\underline{\sigma}}_t\right)$ and the solid pressure p_s (which is equal to p_w for saturated systems)[6]

$$\begin{aligned}\frac{\partial \rho_s}{\partial t} &= \frac{\partial \rho_s}{\partial \sigma_t}\frac{\partial \sigma_t}{\partial t} + \frac{\partial \rho_s}{\partial p_s}\frac{\partial p_s}{\partial t},\\ &= \frac{\partial \rho_s}{\partial \sigma_t}\frac{\partial \sigma_t}{\partial t} + \frac{\partial \rho_s}{\partial p_s}\frac{\partial p_w}{\partial t}.\end{aligned} \tag{3.47}$$

The dependency of the solid density on the isotropic total stress can be written as

$$\frac{\partial \rho_s}{\partial \sigma_t} = -\frac{\rho_s C_s}{1-n}, \tag{3.48}$$

whereas the dependency on the solid pressure is

$$\frac{\partial \rho_s}{\partial p_s} = -\frac{\rho_s C_s}{1-n} n. \tag{3.49}$$

Combining Eqs. (3.47) - (3.49) we have

$$\begin{aligned}\frac{\partial \rho_s}{\partial t} &= -\frac{\rho_s C_s}{1-n}\frac{\partial \sigma_t}{\partial t} - n\frac{\rho_s C_s}{1-n}\frac{\partial p_w}{\partial t},\\ &= \frac{\rho_s C_s}{1-n}\left(-\frac{\partial \sigma_t}{\partial t} - n\frac{\partial p_w}{\partial t}\right).\end{aligned} \tag{3.50}$$

Making use of Eq. (3.50) we can express Eq. (3.46) as

$$C_w n \frac{\partial p_w}{\partial t} + \nabla \cdot \underline{q} + C_s\left(-\frac{\partial \sigma}{\partial t} - n\frac{\partial p_w}{\partial t}\right) + \nabla \cdot \underline{v}_s = \frac{\dot{m}_w}{\rho_w}. \tag{3.51}$$

Grouping common terms, Eq. (3.51) becomes

$$(C_w n - C_s n)\frac{\partial p_w}{\partial t} + C_s \frac{\partial \sigma_t}{\partial t} + \nabla \cdot \underline{v}_s + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w} \tag{3.52}$$

In this way, Eq. (3.52) is not really useful, since the isotropic total stress appears explicitly. An alternative

[6] This will not be the case for unsaturated systems as we will see in section 3.4

is to make use of the principle of effective stress to replace σ_t with $\sigma_e - \alpha p_w$. By doing this, we can express the time derivative of the isotropic total stress as

$$\frac{\partial \sigma_t}{\partial t} = \frac{\partial \sigma_e}{\partial t} - \alpha \frac{\partial p_w}{\partial t}. \tag{3.53}$$

Now, we could write the time derivative of the isotropic effective stress as a function of the volume strain $\varepsilon := \mathbf{Tr}(\underline{\underline{\varepsilon}})$

$$\frac{\partial \sigma_e}{\partial t} = \frac{\partial \sigma_e}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial t}. \tag{3.54}$$

The partial derivative of the isotropic effective stress w.r.t the volume strain can be written as

$$\frac{\partial \sigma_e}{\partial \varepsilon} = \frac{1}{C_m}. \tag{3.55}$$

The time derivative of the volume strain is the divergence of the velocity of the solid phase

$$\frac{\partial \varepsilon}{\partial t} = \nabla \cdot \underline{v}_s. \tag{3.56}$$

Now, by combining Eqs. (3.53) - (3.56) we can express the time derivative of the isotropic of the total stress as

$$\frac{\partial \sigma_t}{\partial t} = \frac{1}{C_m} \nabla \cdot \underline{v}_s - \alpha \frac{\partial p_w}{\partial t}. \tag{3.57}$$

Making use of Eq. (3.57) we can rewrite Eq. (3.52) as

$$(C_w n - C_s n) \frac{\partial p_w}{\partial t} - C_s \left(\frac{1}{C_m} \nabla \cdot \underline{v}_s - \alpha \frac{\partial p_w}{\partial t} \right) + \nabla \cdot \underline{v}_s + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \tag{3.58}$$

Using the definition of the Biot's coefficient (see Eq. (2.55)) and noting that $\nabla \cdot \underline{v}_s = \frac{\partial}{\partial t} (\nabla \cdot \underline{u})$ we can express Eq. (3.58) as

$$\left[(\alpha - n) C_s - n C_w \right] \frac{\partial p_w}{\partial t} + \alpha \frac{\partial}{\partial t} (\nabla \cdot \underline{u}) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \tag{3.59}$$

The term $[(\alpha - n) C_s - n C_w]$ is often called *storativity*, and it is a measure of the compressibility of the porous medium. Let us denote the storativity as S_ε, and present the final version of the storage equation

$$S_\varepsilon \frac{\partial p_w}{\partial t} + \alpha \frac{\partial}{\partial t} (\nabla \cdot \underline{u}) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \tag{3.60}$$

Note that there are four possible mechanisms that could cause of variation in the amount of water within the system:

1. Compressibility effects.
2. Deformation of the porous medium.
3. Flux of water due to changes in pressure and/or elevation.
4. External sources or sinks of water.

3.3.3 The complete set of equations

Let us now present the complete set of equations that governs the saturated flow in deformable porous media. These are

◇ Linear momentum balance equation

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla p_w + [(1-n)\rho_s + n\rho_w]\, \underline{g} = 0. \qquad \text{(3.31 revisited)}$$

◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \qquad \text{(2.29 revisited)}$$

◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}}_e = \mathfrak{C}\, \underline{\underline{\varepsilon}}. \qquad \text{(3.33 revisited)}$$

◇ Storage equation

$$S_\varepsilon \frac{\partial p_w}{\partial t} + \alpha \frac{\partial}{\partial t}(\nabla \cdot \underline{u}) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \qquad \text{(3.60 revisited)}$$

◇ Darcy's law

$$\underline{q}_w = -\frac{\underline{\underline{k}}}{\mu_w}\left(\nabla p_w - \rho_w \underline{g}\right). \qquad \text{(3.40 revisited)}$$

The total number of unknowns for a three-dimensional problem is 25 (9 effective stresses, 9 strains, 3 displacements, 3 Darcy's velocities and the water pressure) and the total number of equations is also 25 (6 linear momentum equations, 9 compatibility equations, 6 independent stress-strain relations, 3 Darcy's law equations and the storage equation).

3.4 Equations of unsaturated flow in deformable porous media

In this section, we are going to present the equations that govern the unsaturated flow in deformable porous media, under the Richards' assumption. The derivation structure is quite similar to the one presented for the saturated flow in deformable porous media. However, there exists some differences due to the presence of air in the system.

3.4.1 Linear momentum balance equation

The equilibrium equations applied to an unsaturated porous medium is given by

$$\nabla \cdot \underline{\underline{\sigma}}_t + \underline{f} = 0, \tag{3.61}$$

where $\underline{\underline{\sigma}}_t$ is given by Eq. (2.58) and the body (density) forces are given by

$$\underline{f} = [(1-n)\rho_s + nS_w\rho_w]\,\underline{g}. \tag{3.62}$$

Combining Eqs. (3.61), (2.58), (3.62) and assuming α as constant we have

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha\nabla\left(S_w p_w\right) + [(1-n)\rho_s + nS_w\rho_w]\,\underline{g} = 0. \tag{3.63}$$

Perhaps, the most important difference between a saturated and unsaturated system, regarding the momentum balance equation, is that the body forces are now a function of the saturation. Since the saturation is a non-linear function of the water pressure (or pressure head), the equilibrium conditions are also non-linear, in principle. More on this issue in the next chapter.

The compatibility equations and the stress-strain relation are exactly the same as in the previous section, see Eqs. (3.32) and (3.33).

3.4.2 Mass conservation

In this section, we present the derivation of the mass conservation for the unsaturated flow in deformable porous media as in (Lewis & Schrefler 1998). There are a lot of similarities between this derivation and the one presented for saturated systems. To avoid repetition, when similarities in the steps are encountered, we are directly going to refer them.

Mass conservation equation for the fluid phase

Let us revisit the general mass conservation equation for an unsaturated system presented in section 3.1.1

$$\frac{\partial\left(\rho_w S_w n\right)}{\partial t}+\nabla\cdot\left(\rho_w S_w n\underline{v}_w\right)=\dot{m}_w. \tag{3.1 revisited}$$

Expanding the accumulation term in Eq. (3.1) gives us

$$\frac{\partial\left(\rho_w S_w n\right)}{\partial t}=nS_w\frac{\partial\rho_w}{\partial t}+\rho_w S_w\frac{\partial n}{\partial t}+n\rho_w\frac{\partial S_w}{\partial t}. \tag{3.64}$$

Using the definition of the relative velocity (see Eq. (2.7)) and neglecting the terms $\nabla\rho_w$ and $\nabla\left(\rho_w S_w n\right)$ we can write the second term of Eq. (3.1) as

$$\begin{aligned}\nabla\cdot\left(\rho_w S_w n\underline{v}_w\right)&=\nabla\cdot\left(\rho_w S_w n\underline{v}_{ws}\right)+\nabla\cdot\left(\rho_w S_w n\underline{v}_s\right),\\&=\nabla\rho_w\cdot\left(S_w n\underline{v}_{ws}\right)+\rho_w\nabla\cdot\left(S_w n\underline{v}_{ws}\right)+\nabla\left(\rho_w S_w n\right)\cdot\underline{v}_s+\rho_w S_w n\nabla\cdot\underline{v}_s,\\&=\rho_w\nabla\cdot\left(S_w n\underline{v}_{ws}\right)+\rho_w S_w n\nabla\cdot\underline{v}_s.\end{aligned} \tag{3.65}$$

Combining Eqs. (3.1), (3.65) and (3.37) we have

$$C_w\rho_w S_w n\frac{\partial p_w}{\partial t}+\rho_w S_w\frac{\partial n}{\partial t}+\rho_w n\frac{\partial S_w}{\partial t}+\rho_w\nabla\cdot\left(nS_w\underline{v}_{ws}\right)+\rho_w S_w n\nabla\cdot\underline{v}_s=\dot{m}_w \tag{3.66}$$

Dividing Eq. (3.66) by $\rho_w S_w$ we get

$$C_w n\frac{\partial p_w}{\partial t}+\frac{\partial n}{\partial t}+\frac{n}{S_w}\frac{\partial S_w}{\partial t}+\frac{1}{S_w}\nabla\cdot\left(S_w n\underline{v}_{ws}\right)+n\nabla\cdot\underline{v}_s=\frac{\dot{m}_w}{\rho_w S_w}. \tag{3.67}$$

Using the multiphase Darcy's law (see Eq. (2.8)), Eq. (3.67) can be written as

$$C_w n\frac{\partial p_w}{\partial t}+\frac{\partial n}{\partial t}+\frac{n}{S_w}\frac{\partial S_w}{\partial t}+\frac{1}{S_w}\nabla\cdot\underline{q}_w+n\nabla\cdot\underline{v}_s=\frac{\dot{m}_w}{\rho_w S_w}. \tag{3.68}$$

Mass conservation for the solid phase

The mass conservation equation for the solid phase is invariant to the presence of multiple phases. Let us revisit its final version

$$\frac{(1-n)}{\rho_s}\frac{\partial\rho_s}{\partial t}-\frac{\partial n}{\partial t}+\nabla\cdot\underline{v}_s-n\nabla\cdot\underline{v}_s=0. \tag{3.45 revisited}$$

Unsaturated storage equation

Following the same steps as in the saturated case, we can add Eqs. (3.68) and (3.45) to eliminate the time derivative of the porosity to get

$$C_w n\frac{\partial p_w}{\partial t} + \frac{n}{S_w}\frac{\partial S_w}{\partial t} + \frac{1}{S_w}\nabla\cdot\underline{q}_w + \frac{(1-n)}{\rho_s}\frac{\partial \rho_s}{\partial t} + \nabla\cdot\underline{v}_s = \frac{\dot{m}_w}{\rho_w S_w}. \tag{3.69}$$

If we multiply Eq. (3.69) by S_w we have

$$C_w S_w n\frac{\partial p_w}{\partial t} + n\frac{\partial S_w}{\partial t} + \nabla\cdot\underline{q}_w + S_w\frac{(1-n)}{\rho_s}\frac{\partial \rho_s}{\partial t} + S_w\nabla\cdot\underline{v}_s = \frac{\dot{m}_w}{\rho_w}. \tag{3.70}$$

Once again, the density of the solid can be expressed as a function of the isotropic total stress σ_t and the solid pressure p_s. Unlike saturated systems, whenever more than one fluid phase is present, the density of the solid is given by the sum of the partial pressures of each fluid phase

$$\begin{aligned} p_s &= \sum_i S_i p_i \quad \forall \quad i = \{a, w\}, \\ &= S_w p_w + (1 - S_w)p_a, \\ &= S_w p_w. \qquad\qquad (p_a = 0 \text{ due to Richards}' \text{ assumption.}) \end{aligned} \tag{3.71}$$

Making use of Eq. (3.71) we have

$$\begin{aligned} \frac{\partial \rho_s}{\partial t} &= \frac{\partial \rho_s}{\partial \sigma_t}\frac{\partial \sigma_t}{\partial t} + \frac{\partial \rho_s}{\partial p_s}\frac{\partial p_s}{\partial t} \\ &= \frac{\partial \rho_s}{\partial \sigma_t}\frac{\partial \sigma_t}{\partial t} + \frac{\partial \rho_s}{\partial p_s}\left(S_w\frac{\partial p_w}{\partial t} + p_w\frac{\partial S_w}{\partial t}\right) \end{aligned} \tag{3.72}$$

The expressions for $\partial\rho_s/\partial\sigma_t$ and $\partial\rho_s/\partial p_s$ are given by Eqs. (3.48) and (3.49). Combining Eqs. (3.72), (3.48) and (3.49) we have

$$\frac{\partial \rho_s}{\partial t} = -\frac{\rho_s C_s}{1-n}\frac{\partial \sigma_t}{\partial t} - \frac{\rho_s C_s}{1-n}n\left(S_w\frac{\partial p_w}{\partial t} + p_w\frac{\partial S_w}{\partial t}\right) = \frac{\rho_s C_s}{1-n}\left(-\frac{\partial \sigma_t}{\partial t} - nS_w\frac{\partial p_w}{\partial t} - np_w\frac{\partial S_w}{\partial t}\right) \tag{3.73}$$

Using Eq. (3.73), we can express Eq. (3.70) as

$$C_w S_w n\frac{\partial p_w}{\partial t} + n\frac{\partial S_w}{\partial t} + \nabla\cdot\underline{q}_w + C_s S_w\left(-\frac{\partial \sigma_t}{\partial t} - nS_w\frac{\partial p_w}{\partial t} - np_w\frac{\partial S_w}{\partial t}\right) + S_w\nabla\cdot\underline{v}_s = \frac{\dot{m}_w}{\rho_w} \tag{3.74}$$

Grouping common terms, Eq. (3.74) becomes

$$\left(C_w S_w n - C_s S_w^2 n\right)\frac{\partial p_w}{\partial t} + \left(n - C_s S_w n p_w\right)\frac{\partial S_w}{\partial t} - C_s S_w\frac{\partial \sigma_t}{\partial t} + S_w\nabla\cdot\underline{v}_s + \nabla\cdot\underline{q}_w = \frac{\dot{m}}{\rho_w} \tag{3.75}$$

By making use of the principle of effective stress for unsaturated systems, we can express the time derivative of the isotropic total stress as

$$\frac{\partial \sigma_t}{\partial t} = \frac{\partial \sigma_e}{\partial t} - \alpha p_w \frac{\partial S_w}{\partial t} - \alpha S_w \frac{\partial p_w}{\partial t}. \tag{3.76}$$

Combining Eqs. (3.76), (3.54), (3.55) and (3.56) we get

$$\frac{\partial \sigma}{\partial t} = \frac{1}{C_m} \nabla \cdot \underline{v}_s - \alpha p_w \frac{\partial S_w}{\partial t} - \alpha S_w \frac{\partial p}{\partial t} \tag{3.77}$$

Combining Eq. (3.75) and (3.77) we get

$$\begin{aligned} &\left(C_w S_w n - C_s S_w^2 n\right) \frac{\partial p_w}{\partial t} + \left(n - C_s S_w n p_w\right) \frac{\partial S_w}{\partial t} \\ &- C_s S_w \left(\frac{1}{C_m} \nabla \cdot \underline{v}_s - \alpha p_w \frac{\partial S_w}{\partial t} - \alpha S_w \frac{\partial p_w}{\partial t} \right) + S_w \nabla \cdot \underline{v}_s + \nabla \cdot \underline{q}_w = \frac{\dot{m}}{\rho_w} \end{aligned} \tag{3.78}$$

By making use of the definition of the Biot's coefficient (see Eq. (2.55)), Eq. (3.78) becomes

$$\begin{aligned} &\left(C_w S_w n - C_s S_w^2 n + C_s S_w^2 \alpha\right) \frac{\partial p_w}{\partial t} + \left(n - C_s S_w n p_w - C_s S_w \alpha p_w\right) \frac{\partial S_w}{\partial t} \\ &+ \alpha S_w \nabla \cdot \underline{v}_s + \nabla \cdot \underline{q}_w = \frac{\dot{m}}{\rho_w} \end{aligned} \tag{3.79}$$

Finally, grouping common terms and noting that $\nabla \cdot \underline{v}_s = \frac{\partial}{\partial t} (\nabla \cdot \underline{u})$, Eq. (3.79) can be written as

$$\begin{aligned} &\left[(\alpha - n) C_s S_w^2 + n C_w S_w\right] \frac{\partial p_w}{\partial t} + \left[(\alpha - n) C_s S_w p_w + n\right] \frac{\partial S_w}{\partial t} \\ &+ \alpha S_w \frac{\partial}{\partial t} (\nabla \cdot \underline{u}) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \end{aligned} \tag{3.80}$$

Eq. (3.80) is commonly referred to as the unsaturated storage equation and as we can see, is highly non-linear. A remarkable characteristic of this equation is that we recover Eq. (3.60) by setting $S_w = 1$. In this respect, Eq. (3.80) can be seen as a generalization of Eq. (3.60).

3.4.3 The complete set of equations

Let us now present the complete set of equations that governs the unsaturated flow in deformable porous media. These are

- ◇ Linear momentum balance equation

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla p_w + \left[(1 - n)\rho_s + n\rho_w\right] \underline{g} = 0. \tag{3.63 revisited}$$

◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \qquad \text{(2.29 revisited)}$$

◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}}_e = \mathfrak{C}\, \underline{\underline{\varepsilon}}. \qquad \text{(3.33 revisited)}$$

◇ Unsaturated storage equation

$$\left[(\alpha - n)\, C_s S_w^2 + n C_w S_w\right] \frac{\partial p_w}{\partial t} + \left[(\alpha - n)\, C_s S_w p_w + n\right] \frac{\partial S_w}{\partial t} + \alpha S_w \frac{\partial}{\partial t}\left(\nabla \cdot \underline{u}\right) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \qquad \text{(3.60 revisited)}$$

◇ Multiphase Darcy's law

$$\underline{q}_w = -\underline{\underline{k}} \frac{k_{rw}}{\mu_w}\left(\nabla p_w - \rho_w \underline{g}\right) \qquad \text{(2.8 revisited)}$$

The total number of unknowns for a three-dimensional problem is the same as in the saturated flow in deformable porous media, i.e 25 (9 effective stresses, 9 strains, 3 displacements, 3 Darcy's velocities and the water pressure) and the total number of equations is also 25 (6 linear momentum equations, 9 compatibility equations, 6 independent stress-strain relations, 3 multiphase Darcy's law equations and the unsaturated storage equation). Nevertheless, note that we must provide an expression of S_w and k_{rw} as function of p_w via some soil water retention curve or similar empirical correlation.

3.5 Boundary and initial conditions

To solve the systems of partial differential equations, we must provide boundary and initial conditions for the elasticity and flow problems. For the elasticity problem, we can specify two types of conditions: displacement and traction vector. Similarly, for the flow problem, we can define pressures (or alternative hydraulic heads) and flows. Denoting Ω the domain of interest and $\partial\Omega$ its boundary, the possible boundary conditions are

$$\underline{u} = g_{\underline{u},D} \quad \text{on} \quad \Gamma_{\underline{u},D}, \tag{3.81}$$

$$\underline{\underline{\sigma}} \cdot \underline{n} = g_{\underline{u},N} \quad \text{on} \quad \Gamma_{\underline{u},N}, \tag{3.82}$$

$$p_w = g_{p,D} \quad \text{on} \quad \Gamma_{p_w,D}, \tag{3.83}$$

$$\underline{q}_w \cdot \underline{n} = g_{p_w,N} \quad \text{on} \quad \Gamma_{p,N}, \tag{3.84}$$

where $\underline{n}$ is the normal vector pointing outwards from the face boundary, and D and N denote Dirichlet and Neumann types of boundary conditions. Hence, the boundary of the domain is given by $\partial\Omega = \Gamma_D \cup \Gamma_N$ (Nordbotten 2016).

For the initial conditions, we must specify a displacement field and a pressure field at $t = 0$. Generally,

the displacement field is set initially to zero which corresponds to an undeformed system, and the water pressure field is considered to be constant (Verruijt 2018)

$$\underline{u} = \underline{u}_0 \qquad \text{for} \qquad t = 0, \tag{3.85}$$

$$p_w = p_{w_0} \qquad \text{for} \qquad t = 0. \tag{3.86}$$

Chapter 4

Discretization and implementation

In this chapter, we focus on the discretization and computational implementation. We start introducing the discretization methodology employed to approximate the solutions to the governing equations, i.e., MPFA and MPSA. We devote a section to introduce the vectorial notation used to represent the continuous variables in a discrete/computational sense. Then, we formally present the discrete MPFA/MPSA operators that later will be used to derive the discrete governing equations. We briefly present the Newton method which will be used to solve the resulting non-linear system of equations. Finally, we explain the general implementation framework and present a simulator based on automatic differentiation.

4.1 Multi Point Flux Approximation

Two Points Flux Approximation (TPFA) methods fail to converge if the grid directions are not aligned with the principal directions of the permeability tensor $\underline{\underline{K}}$. However, Multi Point Flux Approximation (MPFA) methods are designed to give a correct discretization of the flow equations for general non-orthogonal grids as well as for general orientation of the principal directions of the permeability tensor (Aavatsmark 2007). This entire section is primarily based on the work of Aavatsmark (2002, 2007).

Without loose of generality, a control-volume formulation involves the computation of the flux of a phase γ

$$Q_i^\gamma = -\int_{S_i} \left(\lambda^\gamma \underline{\underline{k}} \nabla p^\gamma\right) \cdot \underline{n}\, dS, \tag{4.1}$$

where S_i is the surface of a cell with volume V_i, λ^γ is the mobility of the phase γ defined as $k_{r\gamma}/\mu_\gamma$ and p_γ is some sort of potential of the phase γ, typically the pressure.

TPFA uses two points of information (see Fig. 4.1a) in order to approximate the flux Q_i

$$Q_i^\gamma \approx \lambda^\gamma \left(p_i^\gamma\right) T_i^r \left(p_1^\gamma - p_2^\gamma\right), \tag{4.2}$$

where T_i^r is the transmissibility evaluated at the face i. Typically, $\lambda^\gamma \left(p_i^\alpha\right)$ is evaluated at node 1 if $T_i^r \left(p_1^\gamma - p_2^\gamma\right) > 0$ and at node 2 otherwise. This criteria is referred to as *upstream weighting*. Other types of approximations are the harmonic and arithmetic average.

On the other hand, to approximate the flux at a face, MPFA uses not only the potentials of the adjacent

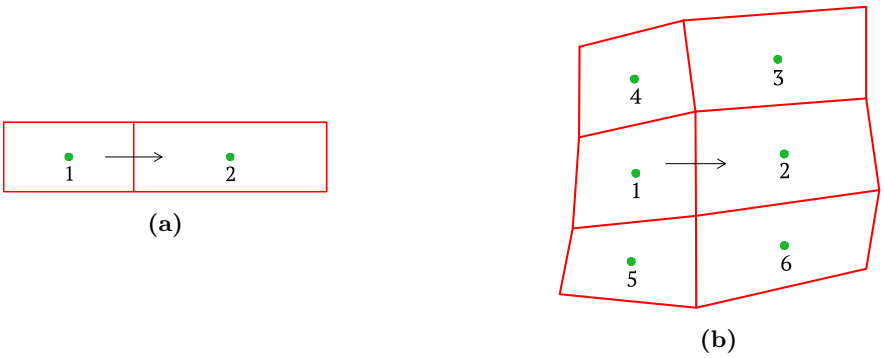


Figure 4.1 a) Two Point Flux Approximation (TPFA). **b)** Multi Point Flux Approximation (MPFA). Adapted from *An introduction to multipoint flux approximations for quadrilateral grids by I. Aavastsmark, p. 407.*

cells, but also the information of the neighbouring cells. The approximation of the flux can be written as

$$Q_i \approx \sum_{j \in J} t_{ij}^r p_j, \tag{4.3}$$

where $\sum_{j \in J} t_{ij}^r = 0$. The coefficients t_{ij}^r are called *transmissibilities* coefficients. Note that the set J will depend on the grid. For a two-dimensional *quadrilateral* grid as in Fig. 4.1b, J consists of the numbers of the six cells. Using Eq. (4.3) we can express Eq. (4.1) as

$$Q_i^\gamma \approx \lambda^\gamma \left(p_i^\gamma\right) \sum_{j \in J} t_{ij}^r p_j^\alpha, \tag{4.4}$$

where $\lambda^\gamma \left(p_i^\gamma\right)$ is evaluated at node 1 if $\sum_{j \in J} t_{ij}^r p_j^\gamma > 0$ and at node 2 otherwise.

If we assume that the potentials at the cell centers are known, both problems reduce to the determination of the transmissibilities. For the TPFA method, this is trivial since T_i^r can be determined by a simple harmonic average of the values transmissibilities between neighboring cells. However, for the MPFA method the task of determining the transmissibility coefficients t_i^r is more complicated.

A formal derivation of the MPFA discretization is beyond the scope of this thesis. However, we will sketch the general procedure involving the approximation of the fluxes for a two-dimensional problem using the *O-method* (more on this later). For an excellent introduction to MPFA we referred to (Aavatsmark 2002).

To calculate the transmissibility coefficients we use a dual grid as shown in Fig. 4.2. The cells within the dual grids are called *interaction regions*. Note that, each interaction region divides the cells interfaces in two parts, which are referred as *subinterfaces*. The core idea is that the local interaction between the cells within an interaction region can be used to calculate the transmissibility coefficients for all the subinterfaces inside the interaction region. Once we know the coefficients for all the subinterfaces, the transmissibility coefficients for the cell interfaces can be found by assembling the contributions from the subinterfaces that form a cell interface.

For simplicity, let us assume a single phase flow. Hence, to evaluate the expression of the flux

$$Q_i = -\underline{n}_i^T \underline{\underline{k}}_j \nabla P_j, \tag{4.5}$$

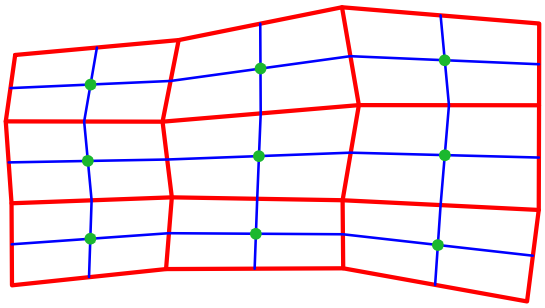


Figure 4.2 Cell grids (in red) and dual grid (in blue). Adapted from *An introduction to multipoint flux approximations for quadrilateral grids by I. Aavastsmark, p. 408.*

we need to compute the gradient ∇P_j and the normal vector $\underline{n}_i$ with length equal to the area of the cell.

In two dimensions the gradient ∇P has two components which are constant in a cell. If we let

$$\bar{p}_k = P\left(\bar{x}_k\right), \quad k = 1, 2, \tag{4.6}$$

then,

$$\nabla P \cdot \left(\bar{x}_k - x_0\right) = \bar{p}_k - p_0, \quad k = 1, 2.$$

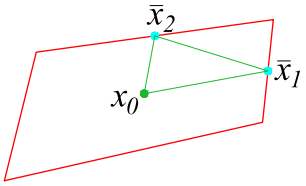


Figure 4.3 Cell center x_0 and continuity points $\bar{x}_1$, $\bar{x}_2$. Adapted from *An introduction to multipoint flux approximations for quadrilateral grids by I. Aavastsmark, p. 410.*

We can build a system of equations of the type

$$X \nabla P = \begin{bmatrix} \bar{p}_1 - u_0 \\ \bar{p}_2 - p_0 \end{bmatrix},$$

where

$$X = \begin{bmatrix} \left(\bar{x}_1 - x_0\right)^T \\ \left(\bar{x}_2 - x_0\right)^T \end{bmatrix}.$$

For convenience, we introduce the matrix R

$$R = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

Also, the determinant[7] of X is

$$H = \det X = \left(\bar{x}_1 - x_0\right)^T R \left(\bar{x}_2 - x_0\right).$$

[7] H represents twice the area of the triangle spanned by x_0, $\bar{x}_1$ and $\bar{x}_2$

To express the inverse of the matrix X, we introduce the vectors $\underline{\nu}_i$ with $i = 1, 2$ given by

$$\underline{\nu}_1 = R\left(\bar{x}_2 - x_0\right), \qquad \underline{\nu}_2 = -R\left(\bar{x}_1 - x_0\right).$$

Now, the inverse of the matrix X is given by

$$X^{-1} = \frac{1}{H}\left[\underline{\nu}_1, \underline{\nu}_2\right].$$

Finally, it follows

$$\nabla p = \frac{1}{H}\sum_{k=1}^{2} \underline{\nu}_k \left(\bar{p}_k - p_0\right). \tag{4.7}$$

Now, we have an expression of the gradient in terms of the geometry and the potentials at the continuity points. Having an expression for the flux (Eq. (4.5)) and the gradient (Eq. (4.7)) we may introduce the scalar quantity

$$\omega_{ijk} = -\frac{\underline{n}_i^T \underline{\underline{K}}_j \underline{\nu}_{jk}}{T_j}, \tag{4.8}$$

defined in an interaction region for the subinterface i, cell j and local cell surface k. The flux through the subinterface can be written in terms of the scalar ω_{ijk} and the potentials

$$f_i = \sum_{k=1}^{D} \omega_{ijk}\left(\bar{p}_{jk} - p_{j0}\right). \tag{4.9}$$

For a two dimensional problem (see Fig. 4.4) we have

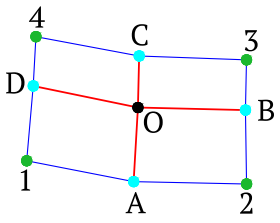


Figure 4.4 Two-dimensional interaction region with cell centers 1,2,3,4, edges midpoints A,B,C,D and common point O. Adapted from *An introduction to multipoint flux approximations for quadrilateral grids by I. Aavastsmark, p. 419.*

$$p(A) = \bar{p}_1,$$
$$p(B) = \bar{p}_2,$$
$$p(C) = \bar{p}_3,$$
$$p(D) = \bar{p}_4.$$

where for each subinterface there is an equation of the form $Q_{i,i-} = Q_{i,i+}$ equating the flux through the subinterface.

Using the notation from above, we can develop a system of equations for the interaction region

$$\begin{aligned}
\omega_{111}\left(\bar{p}_1 - p_1\right) + \omega_{112}\left(\bar{p}_4 - p_1\right) &= \omega_{121}\left(\bar{p}_2 - p_2\right) + \omega_{122}\left(\bar{p}_1 - p_2\right), \\
\omega_{221}\left(\bar{p}_2 - p_2\right) + \omega_{231}\left(\bar{p}_1 - p_2\right) &= \omega_{222}\left(\bar{p}_3 - p_3\right) + \omega_{232}\left(\bar{p}_2 - p_3\right), \\
\omega_{331}\left(\bar{p}_3 - p_3\right) + \omega_{332}\left(\bar{p}_2 - p_3\right) &= \omega_{341}\left(\bar{p}_4 - p_4\right) + \omega_{342}\left(\bar{p}_3 - p_4\right), \\
\omega_{441}\left(\bar{p}_4 - p_4\right) + \omega_{442}\left(\bar{p}_3 - p_4\right) &= \omega_{441}\left(\bar{p}_1 - p_1\right) + \omega_{412}\left(\bar{p}_4 - p_1\right).
\end{aligned}$$

We would like to express the potential at the midpoints $\underline{v} = [\bar{p}_1, \bar{p}_2, \bar{p}_3, \bar{p}_4]^T$ as a function of the potential at the cell centers $\underline{p} = [p_1, p_2, p_3, p_4]^T$. In order to this, we expand the LHS of the system of equations to get

$$\begin{aligned}
Q_1 &= \omega_{111}\bar{p}_1 + \omega_{112}\bar{p}_4 - \left(\omega_{111} + \omega_{112}\right) p_1, \\
Q_2 &= \omega_{222}\bar{p}_1 + \omega_{221}\bar{p}_2 - \left(\omega_{221} + \omega_{222}\right) p_2, \\
Q_3 &= \omega_{332}\bar{p}_2 + \omega_{331}\bar{p}_3 - \left(\omega_{331} + \omega_{332}\right) p_3, \\
Q_4 &= \omega_{442}\bar{p}_3 + \omega_{441}\bar{p}_4 - \left(\omega_{441} + \omega_{442}\right) p_4.
\end{aligned}$$

In matrix form we can write $\underline{Q} = C\underline{v} - D\underline{p}$

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} \omega_{111} & 0 & 0 & \omega_{112} \\ \omega_{222} & \omega_{221} & 0 & 0 \\ 0 & \omega_{332} & \omega_{331} & 0 \\ 0 & 0 & \omega_{442} & \omega_{441} \end{bmatrix} \begin{bmatrix} \bar{p}_1 \\ \bar{p}_2 \\ \bar{p}_3 \\ \bar{p}_4 \end{bmatrix} - \begin{bmatrix} \omega_{111} + \omega_{112} & 0 & 0 & 0 \\ 0 & \omega_{221} + \omega_{222} & 0 & 0 \\ 0 & 0 & \omega_{331} + \omega_{332} & 0 \\ 0 & 0 & 0 & \omega_{441} + \omega_{442} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix}.$$

We can also write the whole system like $A\underline{v} = B\underline{u}$ where

$$A = \begin{bmatrix} \omega_{111} - \omega_{121} & -\omega_{121} & 0 & \omega_{112} \\ \omega_{221} & \omega_{221} - \omega_{232} & -\omega_{231} & 0 \\ 0 & \omega_{332} & \omega_{331} - \omega_{342} & -\omega_{341} \\ -\omega_{441} & 0 & \omega_{441} & \omega_{441} - \omega_{412} \end{bmatrix},$$

and

$$B = \begin{bmatrix} \omega_{111} + \omega_{112} & -\omega_{111} - \omega_{122} & 0 & 0 \\ 0 & \omega_{221} + \omega_{222} & -\omega_{231} - \omega_{232} & 0 \\ 0 & 0 & \omega_{331} + \omega_{332} & -\omega_{341} - \omega_{342} \\ -\omega_{441} - \omega_{412} & 0 & 0 & -\omega_{441} - \omega_{442} \end{bmatrix}.$$

If $A\underline{v} = B\underline{p}$, then $\underline{v} = A^{-1}B\underline{p}$. Hence, we can express the flux as a function of the potentials at the centers as

$$\underline{Q} = \left(CA^{-1}B - D\right)\underline{p}.$$

Finally, the transmissibility coefficients of the subinterfaces are defined by

$$\underline{Q} = T^r \, \underline{p},$$

where $T^r = CA^{-1}B - D = \{t^r_{ij}\}$ is a 4×4 matrix. $T^r = \{t^r_{ij}\}$ are the transmissibility coefficients in one interaction region. To get the transmissibility coefficients of an entire interface, one has to solve for the two interaction regions which contain part of the interface, and add the fluxes of the subinterfaces.

In three dimensions there are eight cells in the interaction region and twelve subinterfaces between cells. For each interaction region we get twelve equations. The vector $\underline{v}$ has dimension 12, and the vector $\underline{p}$ has dimension 8. C and A are 12x12-matrices and D, B and T^r are 12x8-matrices. Note that inverting the matrix A in three-dimensions can represent a non-trivial cost depending on the level of grid refinement.

We conclude this section by remarking that the above method corresponds to a particular family of MPFA methods, the so called $O-$method. Other types of methods such as the $L-$method or enriched MPFA methods differ in the requirements imposed for the continuity conditions of fluxes and potentials within the interaction regions. For an introduction to the L-method we refer to (Aavatsmark 2007), and for a comparative analysis on the monotonicity of MPFA methods we refer to (Keilegavlen & Aavatsmark 2011).

4.2 Multi Point Stress Approximation

The development of cell-centered finite volume discretizations for deformation is motivated by the desire for a compatible approach with the discretization of fluid flow in deformable porous media. The Multi Point Stress Approximation (MPSA) attempt to generalize the established finite volume method for the flow equations (scalar equations) to linear elasticity (vector equations). This gave rise to a method which has cell-centered values of displacements as the only primary unknowns (Nordbotten 2014). This section is based on the articles (Nordbotten 2014, 2016).

Let us now briefly present the core idea behind the MPSA discretization as stated in (Nordbotten 2014). We consider a partition of the domain into a finite number of non-overlapping cells Ω_i. For two cells i and j, we denote the shared face as $e_{i,j}$. The control volume formulation of the *static* momentum balance equation for each cell holds

$$\int_{\Omega_i} \nabla \cdot \underline{\underline{\sigma}} \, dV + \int_{\Omega_i} \rho_s \underline{g} \, dV = \sum_j \int_{e_{i,j}} \underline{T} \, dA + \int_{\Omega_i} \rho_s \underline{g} \, dV = 0. \tag{4.10}$$

Computing the integrals, Eq. (4.10) can alternatively be expressed as

$$\frac{1}{|\Omega_i|} \sum_j |e_{i,j}| \underline{T}_{i,j} + \rho_s \underline{g}_i, \tag{4.11}$$

where $\rho_s \underline{g}_i$ should be interpreted as the volume averaged body forces and $\underline{T}_{i,j}$ as the surface averaged

traction vector.

The finite volume method is completed through the choice of a discrete expression for the stress on cell faces. This can be done using a suitable constitutive law that relates stresses with displacements, e.g. the generalized Hooke's law. In order to preserve the conservation property in Eq. (4.11), we require $\underline{T}_{i,j} = -\underline{T}_{i,j}$, where in each face the traction vector is linearly approximated as

$$\underline{T}_{i,j} = \sum_k t_{i,j,k}\, \underline{u}_k, \tag{4.12}$$

where $t_{i,j,k} = -t_{i,j,k}$ are the stress weight tensors, and can be seen as an analogous of the transmissibility coefficients for the MPFA.

Now the problem reduces to the calculation of the stress weight tensors $t_{i,j,k}$ for a face $e_{i,j}$. As in the flow problem, we use a dual grid that divides the faces into subfaces (see Fig. 4.2). The volume associated with a corner l and the cell i is called a subcell, and denoted as $\tilde{\Omega}_{i,l}$. Moreover, we will refer to the subfaces as $\tilde{e}_{i,j,l}$, where l is the corner associated with the subface. Now we can express the stress weight tensors associated to a face as the contributions of the stress weight tensors of each subface

$$t_{i,j,l} = \sum_l \tilde{t}_{i,j,l,k}. \tag{4.13}$$

Given a corner l, the number of subcells considered to determine the stress weight tensors for each subface will depend on the family of MPSA method adopted. The most popular, the $O-$method, uses an interaction region such that all subfaces and all subcells adajcents to the corner l are taken into account. Other methods, such the $L-$method and $U-$method consider other criterias to select the interaction regions.

We will now construct the subface stress weights $\tilde{t}_{i,j,l,k}$ for each interaction region, which together with Eqs. (4.11)-(4.13) will define the discretization method. Let us consider a linear approximation of the stress weights for the subface $\tilde{e}_{i,j,l}$ to the displacement within each subcell $\tilde{\Omega}_{i,l}$, which in turn are approximated by a multi-linear function of the spatial coordinates

$$\underline{u} \approx \underline{u}_i + \tilde{\nabla}_{i,j} \cdot (\underline{x} - \underline{x}_i) \qquad \text{for} \qquad \underline{x}_i \in \tilde{\Omega}_{i,l}, \tag{4.14}$$

where $\tilde{\nabla}_{i,j}$ is the gradient operator associated with the subcell $\tilde{\Omega}_{i,l}$.

Now, we impose the continuity requirements for the traction vector and displacement. The continuity for the traction vector can be expressed as

$$\left[\mathfrak{C}_i \left(\tilde{\nabla}_{i,l} + \tilde{\nabla}_{i,l}^T\right)\right] \cdot \bar{\underline{n}}_{i,m,l} - \left[\mathfrak{C}_j \left(\tilde{\nabla}_{j,l} + \tilde{\nabla}_{j,l}^T\right)\right] \cdot \bar{\underline{n}}_{i,m,l} = 0 \qquad \text{if} \qquad \tilde{e}_{i,m,l} \subset \tilde{e}_{i,j,l}, \tag{4.15}$$

where $\bar{\underline{n}}_{i,m,l}$ is the average normal vector of the subface $\tilde{e}_{i,m,l}$.

Generally, we can not require *full* continuity of both, traction vector and displacements since this will lead to an overdetermined system. Thus, we only require continuity of the displacements in specific points

$\tilde{x}_{i,m,l,n}$ for each subface $\tilde{e}_{i,m,l}$ such that

$$\tilde{\nabla}_{i,l} \cdot \left(\tilde{\underline{x}}_{i,m,l,n} - \underline{x}_i\right) - \tilde{\nabla}_{j,l} \cdot \left(\tilde{\underline{x}}_{i,m,l,n} - \underline{x}_j\right) = \underline{u}_j - \underline{u}_i. \tag{4.16}$$

If only one continuity point is adopted, the constrain is referred as weak continuity of displacement. If N points are chosen on the subface $\tilde{e}_{i,j,l}$ the constrain is referred as strong continuity of displacement.

The choice of different types of continuity constrains define the families of MPSA methods, i.e. $O-$method, $L-$method and U-method. Regardless the family of MPSA method adopted, the mathematical construction discussed above results in linear systems for the unknown gradients $\tilde{\nabla}_{j,l}$ within the interaction region under consideration. Once the gradients are known, the local stress weights can be determined from Eq. (4.15). Finally, the local stress weights must be assemble to determine the traction vectors associated at each face.

4.3 Discrete representation of continuous variables

Solving the set of governing equations using numerical methods, typically requires developing a computational code. The data structure chosen to represent the variables has a significant impact on the overall efficiency of an algorithm. In this thesis, we use a vectorial representation of the variables of interest, i.e. pressures, displacements, fluxes and traction vectors.

In the context of a cell centred finite volume discretization, the primary variables are evaluated at cell centres (i.e., pressures and displacements), and the fluxes and traction vectors are evaluated at the faces. For scalar equations, as in flow problems, there are two vector spaces of interest, the vector space of cell centres with dimension $\mathbb{R}^{\mathtt{N_c}}$ and the vector space of faces with dimension $\mathbb{R}^{\mathtt{N_f}}$, where $\mathtt{N_c}$ and $\mathtt{N_f}$ represent the total number of cell centres and faces. However, for vector equations, as in the linear elasticity equations, the dimensions of the vector spaces depend not only upon the number of faces and cells but also on the dimension of the problem. For example, for a plane stress problem, we will evaluate the displacements at the cells centres and the traction vectors at the faces. Nevertheless, both displacements and traction vectors have two components, one in the x-direction, and one in the y-direction. Hence, the vector spaces of interest are $\mathbb{R}^{\mathtt{N_d}\cdot\mathtt{N_c}}$ for the displacements, and $\mathbb{R}^{\mathtt{N_d}\cdot\mathtt{N_f}}$ for the traction vectors, where $\mathtt{N_d}$ can take the values, 1, 2 or 3 depending on the dimension of the problem.

We are now in position to introduce the discrete analogous of the variables of interest. We will use "bold" notation to represent vectors (in a computational sense) in order to avoid any confusion with the "underline " notation used for representing vectors and tensors in the continuous domain.

The water pressure is represented as a column vector and takes the following form

$$\boldsymbol{p_w} = \{p_{w_1}, \cdots, p_{w_{\mathtt{N_c}}}\}^T. \tag{4.17}$$

Analogously, the water flux can be represented as

$$\boldsymbol{Q_w} = \{Q_{w_1}, \cdots, Q_{w_{\mathtt{N_f}}}\}^T. \tag{4.18}$$

The displacement and traction vectors are denoted as

$$\begin{aligned}\mathbf{u} &= \{\mathbf{u}_1, \cdots, \mathbf{u}_{\mathtt{N_c}}\}^T \\ &= \{\{u_1, \cdots, u_{1\cdot\mathtt{N_d}}\}, \cdots, \{u_{\mathtt{N_c}\cdot\mathtt{N_d}-\mathtt{N_d}+1}, \ldots, u_{\mathtt{N_c}\cdot\mathtt{N_d}}\}\}^T,\end{aligned} \tag{4.19}$$

and

$$\begin{aligned}\boldsymbol{T} &= \{\boldsymbol{T}_1, \cdots, \boldsymbol{T}_{\mathtt{N_f}}\}^T \\ &= \{\{T_1, \cdots, T_{1\cdot\mathtt{N_d}}\}, \cdots, \{T_{\mathtt{N_f}\cdot\mathtt{N_d}-\mathtt{N_d}+1}, \cdots, T_{\mathtt{N_f}\cdot\mathtt{N_d}}\}\}^T.\end{aligned} \tag{4.20}$$

Note that, both displacement and stress vectors are constructed by concatenating their corresponding components. The number of components will obviously depend upon the dimensionality of the problem.

4.4 Discrete MPFA/MPSA operators

Instead of using an index notation to derive the discretized versions of the governing equations, we propose an alternative notation system inspired by Krogstad et al. (2015), which involves the use of *discrete operators*. These operators, as the name suggests, are the discrete counterparts of the continuous operators, such as gradient and divergence. When the discrete operators act on a specific vector field, they will transform, or in other words, map the vector from one vector space to another. The general mapping can be stated as

$$f : V \to W, \tag{4.21}$$

where f is a generic discrete operator, V and W are two vector spaces, typically (but not necessarily) of different dimensions.

The vector spaces of interest are the ones where pressures, fluxes, displacements and traction vectors are evaluated

$$\mathbb{P} \in \mathbb{R}^{\mathtt{N_c}}, \tag{4.22}$$

$$\mathbb{F} \in \mathbb{R}^{\mathtt{N_f}}, \tag{4.23}$$

$$\mathbb{U} \in \mathbb{R}^{\mathtt{N_d}\cdot\mathtt{N_c}}, \tag{4.24}$$

$$\mathbb{S} \in \mathbb{R}^{\mathtt{N_d}\cdot\mathtt{N_f}}. \tag{4.25}$$

The discrete operators are shown in Table 4.1. A brief description of the operators is presented below.

Table 4.1 Definition of MPFA/MSFA operators

Operator	Maps from	To	Matrix dimension
$\mathbf{F}(\cdot)$	$\mathbb{P}$	$\mathbb{F}$	$\mathtt{N_f} \times \mathtt{N_c}$
$\mathbf{boundF}(\cdot)$	$\mathbb{F}$	$\mathbb{F}$	$\mathtt{N_f} \times \mathtt{N_f}$
$\mathbf{divF}(\cdot)$	$\mathbb{F}$	$\mathbb{P}$	$\mathtt{N_c} \times \mathtt{N_f}$
$\mathbf{S}(\cdot)$	$\mathbb{U}$	$\mathbb{S}$	$(\mathtt{N_d} \cdot \mathtt{N_f}) \times (\mathtt{N_d} \cdot \mathtt{N_c})$
$\mathbf{boundS}(\cdot)$	$\mathbb{S}$	$\mathbb{S}$	$(\mathtt{N_d} \cdot \mathtt{N_f}) \times (\mathtt{N_d} \cdot \mathtt{N_f})$
$\mathbf{divS}(\cdot)$	$\mathbb{S}$	$\mathbb{U}$	$(\mathtt{N_d} \cdot \mathtt{N_c}) \times (\mathtt{N_d} \cdot \mathtt{N_f})$
$\mathbf{gradP}(\cdot)$	$\mathbb{P}$	$\mathbb{U}$	$(\mathtt{N_d} \cdot \mathtt{N_c}) \times \mathtt{N_c}$
$\mathbf{divU}(\cdot)$	$\mathbb{U}$	$\mathbb{P}$	$\mathtt{N_c} \times (\mathtt{N_d} \cdot \mathtt{N_c})$
$\mathbf{compat}(\cdot)$	$\mathbb{P}$	$\mathbb{P}$	$\mathtt{N_c} \times \mathtt{N_c}$

$\mathbf{F}(\cdot)$ computes the fluxes by taking the gradient of the pressures, which are evaluated at the cell centres. This operator takes into account the averaged transmissibilities, which are computed using a harmonic average.

$\mathbf{boundF}(\cdot)$ maps the boundary values imposed at the external faces of the domain, to the faces of the domain. Since a combination of boundary conditions can potentially be used (i.e., constant pressure, constant flux and their combinations), the purpose of this operator is to generalize their computation for all possible cases.

$\mathbf{divF}(\cdot)$ computes the divergence of a flux, mapping the values from the faces to the cell centres.

$\mathbf{S}(\cdot)$ computes the traction vector by taking the gradient of the displacement field according to a given stress-strain relationship, mapping each of the components of the displacement (evaluated at the cell centres) to the faces.

$\mathbf{boundS}(\cdot)$ maps the boundary values imposed at the external faces of the domain, to the faces of the domain. Analogously to the flow problem, this operator deals with all the possible elasticity boundary conditions (i.e., fixed displacement, constant traction force and their combinations).

$\mathbf{divS}(\cdot)$ computes the divergence of the stress, mapping the values from the faces to the cell centres.

$\mathbf{gradP}(\cdot)$ computes the gradient of the pressure, mapping the pressure at the cell centres to the displacement vector space $\mathbb{U}$.

$\mathbf{divU}(\cdot)$ computes the divergence of the displacement field, mapping the values at the cell centres to the pressure vector space $\mathbb{P}$.

$\mathbf{compat}(\cdot)$ provides stability for the coupled poro-elastic system. This operator comes naturally from the discretization procedure, and it is specially useful when small time steps are used to solve the Biot equations.

4.5 Discretization of the mathematical models

In this section, we present a detailed derivation of the discretization of the governing equations discussed in Chapter 3. The discretization is being made using the discrete operators introduced in section 4.4. For time-dependent problems, we will first present the time discretization and then use the resulting

semi-discrete set of equations to perform the spatial discretization. As we will see, the main advantage of using discrete operators against the conventional indexing notation is that set of discrete equations preserve their basic structure, resulting in a straightforward analogy with the governing equations of the continuous domain.

4.5.1 The discrete equations for the unsaturated flow in non-deformable porous media

We start by presenting the discretization procedure for the incompressible mixed-based form of the Richards' equation as derived in section 3.1

$$\frac{\partial \theta_w}{\partial t} + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}, \tag{4.26}$$

$$\underline{q}_w = -\frac{\underline{\underline{k}}\rho_w g}{\mu_w} k_{rw} \nabla h_w. \tag{4.27}$$

Time discretization

Due to the non-linear nature of the Richards' equation, an iterative strategy must be used to solve the discretized set of equations, i.e. Newton-Rhapson[8] or similar solvers. Hence, we will denote with the superscript $\mathtt{n}$ and $\mathtt{n+1}$ the *old* and *new* time levels, and with $\mathtt{m}$ and $\mathtt{m+1}$ the *old* and *new* iteration levels.

The time derivative of the water content of Eq. (4.26) can be approximated using backward Euler

$$\frac{\partial \theta_w}{\partial t} = \frac{[\theta_w]^{\mathtt{n+1,m+1}} - [\theta_w]^{\mathtt{n}}}{\tau^{\mathtt{n}}}, \tag{4.28}$$

where τ denotes the time step. The use of brackets in the variables is used only for readability purposes.

As suggested by Celia et al. (1990), in order to ensure mass conservation throughout the numerical computation, it is convenient to use the *modified Picard iteration*. This is typically done by Taylor-expanding the term ${\theta_w}^{\mathtt{n+1,m+1}}$ as a function of ψ

$$\begin{aligned}
[\theta_w]^{\mathtt{n+1,m+1}} &\approx [\theta_w]^{\mathtt{n+1,m}} + \left[\frac{\partial \theta}{\partial \psi_w}\right]^{\mathtt{n+1,m}} \left([\psi_w]^{\mathtt{n+1,m+1}} - [\psi_w]^{\mathtt{n+1,m}}\right), \\
&\approx [\theta_w]^{\mathtt{n+1,m}} + [C]^{\mathtt{n+1,m}} \left([\psi_w]^{\mathtt{n+1,m+1}} - [\psi_w]^{\mathtt{n+1,m}}\right), \\
&\approx [\theta_w]^{\mathtt{n+1,m}} + [C]^{\mathtt{n+1,m}} \left([h_w]^{\mathtt{n+1,m+1}} - [h_w]^{\mathtt{n+1,m}}\right),
\end{aligned} \tag{4.29}$$

where C is the specific moisture capacity given by Eq. (3.14). Note that a difference in pressure heads ψ is equal to a difference in hydraulic heads h, since the elevation head ζ does not change with time.

[8] The implementation of the solvers will be discussed with details in the next chapter.

Combining Eqs. (4.28) and (4.29) we have

$$\frac{\partial \theta_w}{\partial t} \approx \frac{[\theta_w]^{\mathtt{n}+1,\mathtt{m}} + [C]^{\mathtt{n}+1,\mathtt{m}} \left([h_w]^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]^{\mathtt{n}}}{\tau^{\mathtt{n}}}. \tag{4.30}$$

Even though there are no explicit time derivatives in Eq. (4.27), the hydraulic heads and the relative permeabilities are time-dependent. Hence, the semi-discrete version of the Darcy's law can be stated as

$$\underline{q}_w = -\frac{\underline{\underline{k}}\rho g}{\mu} [k_{rw}]^{\mathtt{n}+1,\mathtt{m}} \nabla [h]^{\mathtt{n}+1,\mathtt{m}+1}. \tag{4.31}$$

Spatial discretization

A typical control volume formulation, requires the integration over cell volumes. In this context, the accumulation term in Eq. (4.26) is given by

$$\begin{aligned} \int_\Omega \frac{\partial \theta_w}{\partial t} dV &= \int_\Omega \left(\frac{[\theta_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} + [C]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \left([h_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]_{\mathtt{c}}^{\mathtt{n}}}{\tau^{\mathtt{n}}} \right) dV, \\ &= \frac{V}{\tau^{\mathtt{n}}} \left([\theta_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} + [C]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \left([h_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]_{\mathtt{c}}^{\mathtt{n}} \right), \end{aligned} \tag{4.32}$$

where we used the subscript $\mathtt{c}$ to denote a cell-centred evaluated quantity.

Note that, since (4.32) holds for any arbitrary volume, it should also hold for all the cells of the domain

$$\begin{aligned} \int_{\Omega_1} \frac{\partial \theta_w}{\partial t} dV &= \frac{V_1}{\tau^{\mathtt{n}}} \left([\theta_w]_1^{\mathtt{n}+1,\mathtt{m}} + [C]_1^{\mathtt{n}+1,\mathtt{m}} \left([h_w]_1^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]_1^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]_1^{\mathtt{n}} \right), \\ \int_{\Omega_2} \frac{\partial \theta_w}{\partial t} dV &= \frac{V_2}{\tau^{\mathtt{n}}} \left([\theta_w]_2^{\mathtt{n}+1,\mathtt{m}} + [C]_2^{\mathtt{n}+1,\mathtt{m}} \left([h_w]_2^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]_2^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]_2^{\mathtt{n}} \right), \\ \vdots \quad &= \qquad\qquad \vdots \\ \int_{\Omega_{\mathtt{N_c}}} \frac{\partial \theta_w}{\partial t} dV &= \frac{V_{\mathtt{N_c}}}{\tau^{\mathtt{n}}} \left([\theta_w]_{\mathtt{N_c}}^{\mathtt{n}+1,\mathtt{m}} + [C]_{\mathtt{N_c}}^{\mathtt{n}+1,\mathtt{m}} \left([h_w]_{\mathtt{N_c}}^{\mathtt{n}+1,\mathtt{m}+1} - [h_w]_{\mathtt{N_c}}^{\mathtt{n}+1,\mathtt{m}} \right) - [\theta_w]_{\mathtt{N_c}}^{\mathtt{n}} \right), \end{aligned}$$

alternatively

$$\int_{\boldsymbol{\Omega}} \frac{\partial \boldsymbol{\theta}}{\partial t} dV = \frac{\boldsymbol{V}}{\tau^{\mathtt{n}}} \left([\boldsymbol{\theta_w}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} + [\boldsymbol{C}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \left([\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}+1} - [\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}} \right) - [\boldsymbol{\theta_w}]_{\mathtt{c}}^{\mathtt{n}} \right), \tag{4.33}$$

where the products between vectors must be interpreted as cell-wise multiplications.

Similarly, the source term in Eq. (4.26) can be represented as

$$\int_{\boldsymbol{\Omega}} \frac{\dot{m}_w}{\rho_w} dV = \boldsymbol{V} \frac{[\dot{\boldsymbol{m}}_w]_{\mathtt{c}}}{\rho_w}. \tag{4.34}$$

Let us now focus on the discretization of the fluxes. Invoking the divergence theorem, we can represent the volume integral of the divergence of the Darcy's velocity as a surface integral

$$\int_{\Omega} \nabla \cdot \underline{q}_w \, dV = \int_{\partial\Omega} \underline{q}_w \cdot \underline{n} \, dA, \tag{4.35}$$

where $\underline{n}$ is the normal vector, pointing outwards to the surface dA of the volume element dV. Generally, integrating over the whole surface of the cell is not practical. A common practice is to divide the total area into the areas A of the faces that compose the cell

$$\int_{\partial\Omega} \underline{q}_w \cdot \underline{n} \, dA = \sum_{\mathtt{f}=1}^{\mathtt{n_f(c)}} \int_{\partial\Omega} \underline{q}_{w_{\mathtt{f}}} \cdot \underline{n}_{\mathtt{f}} \, dA, \tag{4.36}$$

where $\mathtt{f}$ denotes an evaluation at the faces and $\mathtt{n_f}$ is the number of faces of a generic cell $\mathtt{c}$.

Using Eq. (4.31), the approximation of the flux takes the form

$$\int_{\partial\Omega} \underline{q}_{w_{\mathtt{f}}} \cdot \underline{n}_{\mathtt{f}} \, dA \approx -\frac{\rho_w g}{\mu_w} \left[A\right]_{\mathtt{f}} \left[\hat{k}\right]_{\mathtt{f}} \left[\breve{k}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}} \left(\tilde{\nabla}_f \left[h_w\right]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_{\mathtt{f}}, \tag{4.37}$$

where $\left[\hat{k}\right]_{\mathtt{f}}$ are evaluated using a harmonic average. On the other hand, the relative permeabilities $\left[\breve{k}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}}$ are approximated using either an upstream criteria or an arithmetic average. Finally, we denote with $\tilde{\nabla}_f(\cdot)$ the MPFA discrete gradient operator.

Since Eq. (4.37) holds for a generic face, it should also hold for all the faces conforming the computational grid

$$\begin{aligned}
\int_{\partial\Omega_1} \underline{q}_{w\mathtt{f}} \cdot \underline{n}_{\mathtt{f}} \, dA &\approx -\frac{\rho_w g}{\mu_w} \left[A\right]_1 \left[\hat{k}\right]_1 \left[\breve{k}_{rw}\right]_1^{\mathtt{n+1,m}} \left(\tilde{\nabla} \left[h_w\right]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_1, \\
\int_{\partial\Omega_2} \underline{q}_{w_{\mathtt{f}}} \cdot \underline{n}_{\mathtt{f}} \, dA &\approx -\frac{\rho_w g}{\mu_w} \left[A\right]_2 \left[\hat{k}\right]_2 \left[\breve{k}_{rw}\right]_2^{\mathtt{n+1,m}} \left(\tilde{\nabla} \left[h_w\right]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_2, \\
\vdots \quad &\approx \quad \vdots \\
\int_{\partial\Omega_{\mathtt{N_f}}} \underline{q}_{w_{\mathtt{f}}} \cdot \underline{n}_{\mathtt{f}} \, dA &\approx -\frac{\rho_w g}{\mu_w} \left[A\right]_{\mathtt{N_f}} \left[\hat{k}\right]_{\mathtt{N_f}} \left[\breve{k}_{rw}\right]_{\mathtt{N_f}}^{\mathtt{n+1,m}} \left(\tilde{\nabla} \left[h_w\right]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_{\mathtt{N_f}}
\end{aligned}$$

Or alternative, in vector form

$$\left[\boldsymbol{Q}_{\boldsymbol{w}_{\mathbf{int}}}\right]_{\mathtt{f}} = \int_{\boldsymbol{\partial\Omega}} \underline{q}_{w_{\mathtt{f}}} \cdot \underline{n}_{\mathtt{f}} \, dA \approx \frac{\rho_w g}{\mu_w} \left[\breve{\boldsymbol{k}}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}} \mathbf{F} \left(\left[\boldsymbol{h}_{\boldsymbol{w}}\right]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_{\mathtt{f}}, \tag{4.38}$$

for all $\mathtt{f} \in [1, \mathtt{N_f}]$, where the discrete operator $\mathbf{F}(\cdot) = -\left[\boldsymbol{A}\right]_{\mathtt{f}} \left[\hat{\boldsymbol{k}}\right]_{\mathtt{f}} \left(\tilde{\boldsymbol{\nabla}}_{\boldsymbol{f}} \left[\cdot\right]_{\mathtt{c}}\right)_{\mathtt{f}}$, and $\left[\boldsymbol{Q}_{\boldsymbol{w}_{\mathbf{int}}}\right]_{\mathtt{f}}$ represents the vector of fluxes evaluated at the *internal* faces.

In order to take into account the boundary fluxes, we make use of the discrete operator $\mathbf{boundF}(\cdot)$. Hence, the vector of boundary fluxes is given by

$$\left[\boldsymbol{Q}_{\boldsymbol{w}_{\mathbf{bound}}}\right]_{\mathtt{f}} = \frac{\rho g}{\mu} \left[\breve{\boldsymbol{k}}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}} \mathbf{boundF} \left(\left[\boldsymbol{b}_{\boldsymbol{f}}\right]_{\mathtt{f}}\right)_{\mathtt{f}}, \tag{4.39}$$

where $\boldsymbol{b_f}$ is the vector of boundary values, containing the information of both, Dirichlet and Neumann boundary conditions.

The vector which encompasses internal and boundary fluxes is given by

$$[\boldsymbol{Q_w}]_{\mathtt{f}} = [\boldsymbol{Q_{w_{int}}}]_{\mathtt{f}} + [\boldsymbol{Q_{w_{bound}}}]_{\mathtt{f}} = \frac{\rho_w g}{\mu_w} \left[\breve{\boldsymbol{k}}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}} \left(\mathbf{F}\left([\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_{\mathtt{f}} + \mathbf{boundF}\left([\boldsymbol{b_f}]_{\mathtt{f}}\right)_{\mathtt{f}}\right). \tag{4.40}$$

Similarly, if Eq. (4.36) holds for a generic cell, it should also hold for every cell

$$\begin{aligned}
\int_{\Omega_1} \nabla \cdot \underline{q}_w \, dV &= \sum_{\mathtt{f}=1}^{\mathtt{n_f(1)}} [Q_w]_{\mathtt{f}} \,, \\
\int_{\Omega_2} \nabla \cdot \underline{q}_w \, dV &= \sum_{\mathtt{f}=1}^{\mathtt{n_f(2)}} [Q_w]_{\mathtt{f}} \,, \\
\vdots \quad &= \quad \vdots \\
\int_{\Omega_{\mathtt{N_c}}} \nabla \cdot \underline{q}_w \, dV &= \sum_{\mathtt{f}=1}^{\mathtt{n_f(N_c)}} [Q_w]_{\mathtt{f}} \,.
\end{aligned}$$

In vectorial notation holds

$$\int_{\boldsymbol{\Omega}} \nabla \cdot \underline{q} \, dV = \mathbf{divF}\left([\boldsymbol{Q_w}]_{\mathtt{f}}\right)_{\mathtt{c}} , \tag{4.41}$$

where $\mathbf{divF}(\cdot)$ is the discrete divergence operator.

The final version of the discrete mass conservation equation is obtained by combining Eqs. (4.33), (4.41) and (4.34), whereas the discrete Darcy flux is given by Eq. (4.40).The complete set of discrete equations that governs the unsaturated flow in non-deformable porous media can be represented as

- ◇ Discrete mass conservation equation

$$\frac{\boldsymbol{V}}{\tau^{\mathtt{n}}} \left([\boldsymbol{\theta_w}]_{\mathtt{c}}^{\mathtt{n+1,m}} + [\boldsymbol{C}]_{\mathtt{c}}^{\mathtt{n+1,m}} \left([\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}} - [\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n+1,m}}\right) - [\boldsymbol{\theta_w}]_{\mathtt{c}}^{\mathtt{n}}\right) + \mathbf{divF}\left([\boldsymbol{Q_w}]_{\mathtt{f}}\right)_{\mathtt{c}} = \boldsymbol{V} \frac{[\dot{\boldsymbol{m}}_{\boldsymbol{w}}]_{\mathtt{c}}}{\rho_w}.$$

- ◇ Discrete multiphase Darcy flux

$$[\boldsymbol{Q_w}]_{\mathtt{f}} = \frac{\rho_w g}{\mu_w} \left[\breve{\boldsymbol{k}}_{rw}\right]_{\mathtt{f}}^{\mathtt{n+1,m}} \left(\mathbf{F}\left([\boldsymbol{h_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\right)_{\mathtt{f}} + \mathbf{boundF}\left([\boldsymbol{b_f}]_{\mathtt{f}}\right)_{\mathtt{f}}\right).$$

Note that all the terms in the mass conservation equation involve quantities either evaluated or mapped at the cell centres, whereas the discrete multiphase Darcy flux involves quantities either evaluated or mapped at the faces. Finally, we have decided to emphasize the vector of unknowns by using a colored version of the variable.

4.5.2 The discrete equations of linear elasticity

Since the linear elasticity equations are time independent, we only need a spatial discretization to derive the approximated governing equations. Let us start by revisiting the system of partial differential equations

- ◇ Equilibrium equations

$$\nabla \cdot \underline{\underline{\sigma}} + \rho_s \underline{g} = 0. \tag{3.28 revisited}$$

- ◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \tag{2.29 revisited}$$

- ◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}} = \mathfrak{C}\, \underline{\underline{\varepsilon}}. \tag{2.48 revisited}$$

Since the primary unknown is the displacement field, we can combine the compatibility equations and the stress-strain relationship to derive a direct relation between the stress and the displacement, or in other words, a *stress-displacement* relationship

$$\underline{\underline{\sigma}} = \frac{1}{2}\,\mathfrak{C}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \tag{4.42}$$

Eq. (4.42) can be seen as an analogous to the Darcy's law for the mechanics problem, where instead of having a Darcy's velocity we have a stress tensor and instead of pressures, displacement fields.

Following the same procedure as in the last section, we can integrate the divergence of the stress tensor over the volume of a cell and apply the divergence theorem to have

$$\int_{\Omega} \nabla \cdot \underline{\underline{\sigma}}\, dV = \int_{\partial\Omega} \underline{\underline{\sigma}} \cdot \underline{n}\, dA = \sum_{\mathtt{f}=1}^{\mathtt{n_f(c)}} \int_{\partial\Omega} \underline{\underline{\sigma}}_{\mathtt{f}} \cdot \underline{n}_{\mathtt{f}}\, dA = \sum_{\mathtt{f}=1}^{\mathtt{n_f(c)}} \int_{\partial\Omega} \underline{T}\, dA, \tag{4.43}$$

where $\underline{T}$ is the traction vector introduced in section 2.2.3.

Using Eq. (4.42) we can approximate the traction vector at each face of the domain

$$[\boldsymbol{T}]_{\mathtt{f}} = \mathbf{S}\left([\boldsymbol{u}]_{\mathtt{c}}\right)_{\mathtt{f}} + \mathbf{boundS}\left([\boldsymbol{b_m}]_{\mathtt{f}}\right)_{\mathtt{f}}. \tag{4.44}$$

In Eq. (4.44), the operator $\mathbf{S}(\cdot) = \frac{1}{2}\left[\mathfrak{C}\right]_{\mathtt{f}}\left(\tilde{\boldsymbol{\nabla}}_{\boldsymbol{m}}\left[\cdot\right]_{\mathtt{c}} + \left(\tilde{\boldsymbol{\nabla}}_{\boldsymbol{m}}\left[\cdot\right]_{\mathtt{c}}\right)^T\right)_{\mathtt{f}}$ acts on the displacement field, mapping each of the components of the displacement to the faces to compute the traction vector. On the other

hand, the operator $\mathbf{boundS}\,(\cdot)$ acts on the vector $\boldsymbol{b_m}$ which contains the information of the boundary conditions, i.e. fixed tractions and fixed displacements.

Since Eq. (4.43) holds for an arbitrary volume, it also holds for all the cells of the computational domain. Hence, we can write

$$\int_{\boldsymbol{\Omega}} \nabla \cdot \underline{\underline{\sigma}}\, dV = \mathbf{divS}\left([\boldsymbol{T}]_{\mathtt{f}}\right)_{\mathtt{c}}, \tag{4.45}$$

where the operator **divS** computes the divergence of the stress field by mapping the traction vector (defined at the faces) to the cell centers.

The body forces vector is obtained by integrating over each volume of the computational grid and concatenating them into a single vector

$$\int_{\Omega} \rho \underline{g}\, dV = \boldsymbol{V}\rho\,[\boldsymbol{g}]_{\mathtt{c}}\,. \tag{4.46}$$

The set of discrete equations can be summarized as

- ◇ Discrete equilibrium equations

$$\mathbf{divS}\left([\boldsymbol{T}]_{\mathtt{f}}\right)_{\mathtt{c}} + \boldsymbol{V}\rho\,[\boldsymbol{g}]_{\mathtt{c}} = 0. \tag{4.47}$$

- ◇ Traction vector-displacement relationship

$$[\boldsymbol{T}]_{\mathtt{f}} = \mathbf{S}\left([\boldsymbol{u}]_{\mathtt{c}}\right)_{\mathtt{f}} + \mathbf{boundS}\left([\boldsymbol{b_m}]_{\mathtt{f}}\right)_{\mathtt{f}}. \tag{4.48}$$

4.5.3 The discrete equations for the saturated flow in deformable porous media

In this section, we are going to present the discrete equations that rule the saturated flow in deformable porous. Since the basic structure of the discretization process is the same as in the previous cases, for the spatial discretization we will work only with the vectorized discrete variables to avoid repetitiveness. As usual, we start by revisiting the governing equations of the continuous domain

- ◇ Linear momentum balance equation

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla p_w + [(1-n)\rho_s + n\rho_w]\,\underline{g} = 0. \tag{3.31 revisited}$$

- ◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \tag{2.29 revisited}$$

◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}}_e = \mathfrak{C}\,\underline{\underline{\varepsilon}}. \qquad \text{(3.33 revisited)}$$

◇ Storage equation

$$S_\varepsilon \frac{\partial p_w}{\partial t} + \alpha \frac{\partial}{\partial t}\left(\nabla \cdot \underline{u}\right) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \qquad \text{(3.60 revisited)}$$

◇ Darcy's law

$$\underline{q}_w = -\frac{\underline{\underline{k}}}{\mu_w}\left(\nabla p_w - \rho_w \underline{g}\right). \qquad \text{(3.40 revisited)}$$

Time discretization

The semi-discrete linear momentum balance equation can be stated as

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla \left[p_w\right]^{\mathtt{n+1}} + \left[(1-n)\rho_s + n\rho_w\right]\underline{g} = 0, \tag{4.49}$$

where the water pressure is evaluated at the new time level $\mathtt{n}+\mathtt{1}$.

As in the elasticity case, instead of relating the effective stress with the strain, we express a direct dependency with the displacement field. In this context, the semi-discrete effective stress-displacement relationship holds

$$\underline{\underline{\sigma}}_e = \frac{1}{2}\mathfrak{C}\left(\nabla \left[\underline{u}\right]^{\mathtt{n+1}} + \left(\nabla \left[\underline{u}\right]^{\mathtt{n+1}}\right)^T\right), \tag{4.50}$$

where the displacements are also evaluated at the new time level $\mathtt{n}+\mathtt{1}$.

Applying backward Euler to $\partial p_w/\partial t$ and $\partial\left(\nabla \cdot \underline{u}\right)/\partial t$, the semi-discrete version of the storage equation becomes

$$S_\varepsilon \frac{\left[p_w\right]^{n+1} - \left[p_w\right]^n}{\tau^{\mathtt{n}}} + \alpha \frac{\nabla \cdot \left(\left[\underline{u}\right]^{\mathtt{n+1}} - \left[\underline{u}\right]^{\mathtt{n}}\right)}{\tau^{\mathtt{n}}} + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \tag{4.51}$$

Multiplying Eq. (4.52) by $\tau^{\mathtt{n}}$ we get

$$S_\varepsilon \left(\left[p_w\right]^{\mathtt{n+1}} - \left[p_w\right]^{\mathtt{n}}\right) + \alpha \nabla \cdot \left(\left[\underline{u}\right]^{\mathtt{n+1}} - \left[\underline{u}\right]^{\mathtt{n}}\right) + \tau^{\mathtt{n}} \nabla \cdot \underline{q}_w = \tau^{\mathtt{n}} \frac{\dot{m}_w}{\rho_w}. \tag{4.52}$$

Finally, the semi-discrete Darcy's law can be written as

$$\underline{q}_w = -\frac{\underline{\underline{k}}}{\mu_w} \nabla \left(\left[p_w\right]^{\mathtt{n+1}} - \rho_w g \zeta\right). \tag{4.53}$$

Spatial discretization

Using Eq. (4.49), the spatial discretization of the linear momentum balance equation follows

$$\mathbf{divS}\left([\boldsymbol{T_e}]_{\mathtt{f}}\right)_{\mathtt{c}} - \alpha\,\mathbf{gradP}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{c}} + \boldsymbol{V}\left[(1-n)\rho_s + n\rho_w\right][\boldsymbol{g}]_{\mathtt{c}} = 0, \tag{4.54}$$

where the discrete operator $\mathbf{gradP}(\cdot)$ computes the gradient of the water pressure, mapping one value per cell to $\mathtt{N_d}$ values per cell. We denote with $\boldsymbol{T_e}$ the *effective* traction vector.

The spatial discretization of the effective traction vector can be written as

$$[\boldsymbol{T_e}]_{\mathtt{f}} = \mathbf{S}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{f}} + \mathbf{boundS}\left([\boldsymbol{b_m}]_{\mathtt{f}}\right)_{\mathtt{f}}. \tag{4.55}$$

Employing Eq. (4.53), the approximated Darcy's flux can be written as

$$[\boldsymbol{Q_w}]_{\mathtt{f}} = \frac{1}{\mu_w}\left(\mathbf{F}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}} + \rho_w g\,[\boldsymbol{\zeta}]_{\mathtt{c}}\right)_{\mathtt{f}} + \mathbf{boundF}\left([\boldsymbol{b_f}]_{\mathtt{f}}\right)_{\mathtt{f}}\right). \tag{4.56}$$

Finally, utilizing Eq. (4.52), the spatial discretization of the storage equation holds,

$$\begin{aligned}&\alpha\,\mathbf{divU}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n+1}} - [\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n}}\right)_{\mathtt{c}} + \alpha^2\,\mathbf{compat}\left([\boldsymbol{p_w}]_{c}^{\mathtt{n+1}}\right)_{\mathtt{c}} + \boldsymbol{V}S_\varepsilon\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}} - [\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n}}\right) + \\ &\tau^{\mathtt{n}}\,\mathbf{divF}\left([\boldsymbol{Q_w}]_{\mathtt{f}}\right)_{\mathtt{c}} = \boldsymbol{V}\tau^{\mathtt{n}}\frac{[\dot{\boldsymbol{m}}_w]_{\mathtt{c}}}{\rho_w},\end{aligned} \tag{4.57}$$

where the operator $\mathbf{divU}(\cdot)$ computes the divergence of the displacement field, mapping their $\mathtt{N_d}$ components to one single value at the cell centres. The operator $\mathbf{compat}(\cdot)$ acts on the water pressure, providing stability when $\tau^{\mathtt{n}} << 1$.

As usual, let us present the summary of the discrete equations that rules the saturated flow in deformable porous media

- ◇ Discrete linear momentum balance equation

  $$\mathbf{divS}\left([\boldsymbol{T_e}]_{\mathtt{f}}\right)_{\mathtt{c}} - \alpha\,\mathbf{gradP}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{c}} + \boldsymbol{V}\left[(1-n)\rho_s + n\rho_w\right][\boldsymbol{g}]_{\mathtt{c}} = 0.$$

- ◇ Discrete effective traction vector-displacement relationship

  $$[\boldsymbol{T_e}]_{\mathtt{f}} = \mathbf{S}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{f}} + \mathbf{boundS}\left([\boldsymbol{b_m}]_{\mathtt{f}}\right)_{\mathtt{f}}.$$

- ◇ Discrete storage equation

  $$\begin{aligned}&\alpha\,\mathbf{divU}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n+1}} - [\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n}}\right)_{\mathtt{c}} + \alpha^2\,\mathbf{compat}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{c}} + \boldsymbol{V}S_\varepsilon\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}} - [\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n}}\right) + \\ &\boldsymbol{V}\tau^{\mathtt{n}}\,\mathbf{divF}\left([\boldsymbol{Q_w}]_{\mathtt{f}}\right)_{\mathtt{c}} = \tau^{\mathtt{n}}\frac{[\dot{\boldsymbol{m}}_w]_{\mathtt{c}}}{\rho_w}.\end{aligned}$$

◇ Discrete Darcy's flux

$$[\boldsymbol{Q_w}]_{\mathtt{f}} = \frac{1}{\mu_w}\left(\mathbf{F}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}} + \rho_w g\,[\boldsymbol{\zeta}]_{\mathtt{c}}\right)_{\mathtt{f}} + \mathbf{boundF}\left([\boldsymbol{b_f}]_{\mathtt{f}}\right)_{\mathtt{f}}\right).$$

Note that for this set of discrete equations we have two primary unknowns, the water pressure and the displacement. The system of equations is solved coupling the flow and elasticity parts, where the two unknowns are represented as a single vector.

4.5.4 The discrete equations for the unsaturated flow in deformable porous media

In this section, we present the discrete equations for the unsaturated flow in deformable porous media. Even in the discrete domain, we will see that the set of equations can be seen as a generalization of the saturated flow in deformable porous media. We completely recover the governing equations by assuming $S_w = 1$.

Let us now revisit the set of equations that governs the unsaturated flow in deformable porous media. These are

◇ Linear momentum balance equation

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla \left(S_w p_w\right) + \left[(1-n)\rho_s + n S_w \rho_w\right] \underline{g} = 0. \qquad \text{(3.63 revisited)}$$

◇ Compatibility equations

$$\underline{\underline{\varepsilon}} = \frac{1}{2}\left(\nabla \underline{u} + (\nabla \underline{u})^T\right). \qquad \text{(2.29 revisited)}$$

◇ Stress-strain relationship for isotropic materials

$$\underline{\underline{\sigma}}_e = \mathfrak{C}\, \underline{\underline{\varepsilon}}. \qquad \text{(3.33 revisited)}$$

◇ Unsaturated storage equation

$$\left[(\alpha - n)\, C_s S_w^2 + n C_w S_w\right] \frac{\partial p_w}{\partial t} + \left[(\alpha - n)\, C_s S_w p_w + n\right] \frac{\partial S_w}{\partial t} + \alpha S_w \frac{\partial}{\partial t}\left(\nabla \cdot \underline{u}\right) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w}. \qquad \text{(3.60 revisited)}$$

◇ Multiphase Darcy's law

$$\underline{q}_w = -\underline{\underline{k}} \frac{k_{rw}}{\mu_w}\left(\nabla p_w - \rho_w \underline{g}\right) \qquad \text{(2.8 revisited)}$$

Time discretization

The semi-discrete linear momentum balance equation holds

$$\nabla \cdot \underline{\underline{\sigma}}_e - \alpha \nabla \left([S_w]^{\mathtt{n}} [p_w]^{\mathtt{n+1,m+1}} \right) + [(1-n)\rho_s + n [S_w]^{\mathtt{n}} \rho_w] \underline{g} = 0. \tag{4.58}$$

Note that we are evaluating the water saturation at the old time level $\mathtt{n}$ in order to reduce the non-linearities. Further investigations must be carried out to determine the numerical impact of the level at which the saturations are evaluated, since we could have also evaluated it at $\mathtt{n+1,m+1}$, or even at $\mathtt{n+1,m}$.

The time discretization of the effective stress-displacement follows

$$\underline{\underline{\sigma}}_e = \frac{1}{2} \mathfrak{C} \left(\nabla [\underline{u}]^{\mathtt{n+1,m+1}} + \left(\nabla [\underline{u}]^{\mathtt{n+1,m+1}} \right)^T \right). \tag{4.59}$$

The semi-discrete extended Darcy's velocity in terms of pressure can be written as

$$\underline{q}_w = -\frac{1}{\mu_w} [k_{rw}]^{\mathtt{n+1,m}} \left(\nabla [p_w]^{\mathtt{n+1,m+1}} + \rho_w g \nabla \zeta \right). \tag{4.60}$$

Let us now focus on the time discretization of the unsaturated storage equation. Applying backward Euler to $\partial p_w / \partial t$ and $\partial \left(\nabla \cdot \underline{u} \right) / \partial t$ we can write

$$\frac{\partial p_w}{\partial t} = \frac{[p_w]^{\mathtt{n+1,m+1}} - [p_w]^{\mathtt{n}}}{\tau^{\mathtt{n}}}, \tag{4.61}$$

and

$$\frac{\partial}{\partial t} \left(\nabla \cdot \underline{u} \right) = \frac{\nabla \cdot \left([\underline{u}]^{\mathtt{n+1,m+1}} - [\underline{u}]^{\mathtt{n}} \right)}{\tau^{\mathtt{n}}}. \tag{4.62}$$

Applying backward Euler to $\partial S_w / \partial t$ we can write

$$\frac{\partial S_w}{\partial t} = \frac{[S_w]^{\mathtt{n+1,m+1}} - [S_w]^{\mathtt{n}}}{\tau^{\mathtt{n}}}. \tag{4.63}$$

Using the modified Picard iteration and some algebraic manipulations, the term $[S_w]^{\mathtt{n+1,m+1}}$ can be expressed as

$$\begin{aligned} [S_w]^{\mathtt{n+1,m+1}} &= [S_w]^{\mathtt{n+1,m}} + \left[\frac{\partial S_w}{\partial \psi_w} \right]^{\mathtt{n+1,m}} \left([\psi_w]^{\mathtt{n+1,m+1}} - [\psi_w]^{\mathtt{n+1,m}} \right), \\ &= [S_w]^{\mathtt{n+1,m}} + \frac{1}{n\rho g} \left[\frac{\partial \theta_w}{\partial \psi_w} \right]^{\mathtt{n+1,m}} \left([p_w]^{\mathtt{n+1,m+1}} - [p_w]^{\mathtt{n+1,m}} \right), \\ &= [S_w]^{\mathtt{n+1,m}} + \frac{1}{n\rho g} [C]^{\mathtt{n+1,m}} \left([p_w]^{\mathtt{n+1,m+1}} - [p_w]^{\mathtt{n+1,m}} \right). \end{aligned} \tag{4.64}$$

Combining Eqs. (4.63) and (4.64) the time derivative of the saturation can be written as

$$\frac{\partial S_w}{\partial t} = \frac{1}{\tau^{\mathtt{n}}}\left([S_w]^{\mathtt{n}+1,\mathtt{m}} + \frac{1}{n\rho g}\,[C]^{\mathtt{n}+1,\mathtt{m}}\left([p_w]^{\mathtt{n}+1,\mathtt{m}+1} - [p_w]^{\mathtt{n}+1,\mathtt{m}} \right) - [S_w]^{\mathtt{n}} \right). \tag{4.65}$$

Using Eqs. (4.61), (4.62) and (4.65) the semi-discrete unsaturated storage equation becomes

$$\begin{aligned}
&\frac{1}{\tau^{\mathtt{n}}}\left[(\alpha - n)\, C_s \left[S_w^2\right]^{\mathtt{n}} + nC_w \left[S_w\right]^{\mathtt{n}} \right] \left([p_w]^{\mathtt{n}+1,\mathtt{m}+1} - [p_w]^{\mathtt{n}} \right) + \\
&\frac{1}{\tau^{\mathtt{n}}}\left[(\alpha - n)\, C_s \left[S_w\right]^{\mathtt{n}} \left[p_w\right]^{\mathtt{n}} + n \right] \left([S_w]^{\mathtt{n}+1,\mathtt{m}} + \frac{1}{n\rho g}\,[C]^{\mathtt{n}+1,\mathtt{m}}\left([p_w]^{\mathtt{n}+1,\mathtt{m}+1} - [p_w]^{\mathtt{n}+1,\mathtt{m}} \right) - [S_w]^{\mathtt{n}} \right) + \\
&\frac{1}{\tau^{\mathtt{n}}}\alpha \left[S_w\right]^{\mathtt{n}} \nabla \cdot \left([\underline{u}]^{\mathtt{n}+1,\mathtt{m}+1} - [\underline{u}]^{\mathtt{n}} \right) + \nabla \cdot \underline{q}_w = \frac{\dot{m}_w}{\rho_w},
\end{aligned} \tag{4.66}$$

where the compressibility-like terms are evaluated at the old time level $\mathtt{n}$.

Multiplying Eq. (4.66) by $\tau^{\mathtt{n}}$ we have the final version of the semi-discrete unsaturated storage equation

$$\begin{aligned}
&\left[(\alpha - n)\, C_s \left[S_w^2\right]^{\mathtt{n}} + nC_w \left[S_w\right]^{\mathtt{n}} \right] \left([p_w]^{\mathtt{n}+1,\mathtt{m}+1} - [p_w]^{\mathtt{n}} \right) + \\
&\left[(\alpha - n)\, C_s \left[S_w\right]^{\mathtt{n}} \left[p_w\right]^{\mathtt{n}} + n \right] \left([S_w]^{\mathtt{n}+1,\mathtt{m}} + \frac{1}{n\rho g}\,[C]^{\mathtt{n}+1,\mathtt{m}}\left([p_w]^{\mathtt{n}+1,\mathtt{m}+1} - [p_w]^{\mathtt{n}+1,\mathtt{m}} \right) - [S_w]^{\mathtt{n}} \right) + \\
&\alpha \left[S_w\right]^{\mathtt{n}} \nabla \cdot \left([\underline{u}]^{\mathtt{n}+1,\mathtt{m}+1} - [\underline{u}]^{\mathtt{n}} \right) + \tau^{\mathtt{n}} \nabla \cdot \underline{q}_w = \tau^{\mathtt{n}} \frac{\dot{m}_w}{\rho_w},
\end{aligned} \tag{4.67}$$

Spatial discretization

Using the semi-discrete equations, it is straightforward to perform the spatial discretization emplyoing the discrete MPFA/MPSA operators. To avoid being repetitive, let us simply present the summary of discrete equations that governs the unsaturated flow in deformable porous media

- ◇ Discrete linear momentum balance equation

$$\mathbf{divS}\left([\boldsymbol{T_e}]_{\mathtt{f}} \right)_{\mathtt{c}} - \alpha\, \mathbf{gradP}\left([\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\, [\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}+1} \right)_{\mathtt{c}} + \boldsymbol{V}\left[(1-n)\rho_s + [\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\, n\rho_w \right] [\boldsymbol{g}]_{\mathtt{c}} = 0.$$

- ◇ Discrete effective traction vector-displacement relationship

$$[\boldsymbol{T_e}]_{\mathtt{f}} = \mathbf{S}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n}+1,\mathtt{m}+1} \right)_{\mathtt{f}} + \mathbf{boundS}\left([\boldsymbol{b_m}]_{\mathtt{f}} \right)_{\mathtt{f}}.$$

◇ Discrete unsaturated storage equation

$$\alpha\,[\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\,\mathbf{divU}\left([\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}-[\boldsymbol{u}]_{\mathtt{c}}^{\mathtt{n}}\right)_{\mathtt{c}}+\alpha^2\,\mathbf{compat}\left([\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}\,[\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1}}\right)_{\mathtt{c}}+$$
$$\boldsymbol{V}\left[(\alpha-n)\,C_s\,[\boldsymbol{S_w^2}]_{\mathtt{c}}^{\mathtt{n}}+nC_w\,[\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\right]\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}-[\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n}}\right)+$$
$$\boldsymbol{V}\left[(\alpha-n)\,C_s\,[\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\,[\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n}}+n\right]\left([\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n+1,m}}+\frac{1}{n\rho g}\,[\boldsymbol{C}]_{\mathtt{c}}^{\mathtt{n+1,m}}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}-[\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1,m}}\right)-[\boldsymbol{S_w}]_{\mathtt{c}}^{\mathtt{n}}\right)+$$
$$\tau^{\mathtt{n}}\,\mathbf{divF}\left([\boldsymbol{Q_w}]_{\mathtt{f}}\right)_{\mathtt{c}}=\boldsymbol{V}\tau^{\mathtt{n}}\,\frac{[\dot{\boldsymbol{m}}_w]_{\mathtt{c}}}{\rho_w}.$$

◇ Discrete multiphase Darcy's flux

$$[\boldsymbol{Q_w}]_{\mathtt{f}}=\frac{1}{\mu_w}\,[\boldsymbol{k_{rw}}]_{\mathtt{c}}^{\mathtt{n+1,m}}\left(\mathbf{F}\left([\boldsymbol{p_w}]_{\mathtt{c}}^{\mathtt{n+1,m+1}}+\rho_w g\,[\boldsymbol{\zeta}]_{\mathtt{c}}\right)_{\mathtt{f}}+\mathbf{boundF}\left([\boldsymbol{b_f}]_{\mathtt{f}}\right)_{\mathtt{f}}\right).$$

4.6 Solving the system of equations

As we already mentioned in the last section, solving non-linear problems requires the implementation of solvers based on iterative schemes. In this context, the method of Newton-Rhapson is a potential candidate, due to its quadratic order of convergence. Other alternatives are the Picard method, the Picard/Newton method and the L-scheme. An interesting comparative analysis among these methods for solving the one-dimensional mixed-based form of the Richards' equation is carried out in (List & Radu 2016).

Possibly, the main disadvantage of using the Newton method is the effort involving the calculation and computation of the Jacobian matrix, especially when complex non-linear constitutive relations (as the van Genuchten water retention curves) are used. Nevertheless, the computation of the Jacobian matrix can be broken down to nested differentiation of elementary operations and functions, becoming a good candidate for using *Automatic Differentiation* (AD) (Krogstad et al. 2015). The use of AD together with the Newton method is then the preferred option for this thesis whenever a non-linear system of equations must be solved.

Let us now briefly present the iterative scheme for a generic problem as in (Lie 2014). If we collect all the discrete equations, we can write the resulting system of nonlinear equations in short vector form as

$$\mathcal{F}(\mathbf{x}^{\mathtt{n+1}};\mathbf{x}^{\mathtt{n}})=0, \tag{4.68}$$

where $\mathbf{x}^{\mathtt{n+1}}$ is the primary vector of unknowns at the *new* time level and $\mathbf{x}^{\mathtt{n}}$ is assumed to be known.

Given an initial guess $\mathbf{x}_0$, we want to move towards the correct solution, $\mathcal{F}(\mathbf{x})=0$. To this end, we write $\mathbf{x}=\mathbf{x}_0+\Delta\mathbf{x}$ and solve for $\mathbf{x}$ from

$$0=\mathcal{F}(\mathbf{x}_0+\boldsymbol{\Delta}\mathbf{x})\approx\mathcal{F}(x_0)+\nabla\mathcal{F}(\mathbf{x}_0)\delta\mathbf{x}. \tag{4.69}$$

This gives rise to the following iterative scheme

$$\frac{d\mathcal{F}}{d\mathbf{x}}(\mathbf{x}^{m})\delta\mathbf{x}^{m+1} = -\mathcal{F}(\mathbf{x}^{m}), \qquad \mathbf{x}^{m+1} \leftarrow \mathbf{x}^{m} + \delta\mathbf{x}^{m+1}, \tag{4.70}$$

where $\mathcal{J} = d\mathcal{F}/d\mathbf{x}$ is the Jacobian, and $\delta\mathbf{x}^{m+1}$ the Newton update.

In Fig. 4.5 we show a flowchart of the Newton-Rhapson iterative scheme applied to a general non-linear equation. First, we call the equation in its residual form `resEq(x_ad,x_0)` which depends on the AD vector of unknowns `x_ad` and the initial guess `x_0`. Then, we compute the Jacobian Matrix and compute the residual. The Newton update is calculated by solving the linear system `y = J \ -F`, where \ is the backslash operator. Finally, we update the vector of unknowns and check the tolerance. If the stipulated tolerance is achieved, we move to the next time level, otherwise, we repeat the process. The amount of times that we could repeat the process is limited by the maximum number of iterations, which is typically between 10 and 30 iterations.

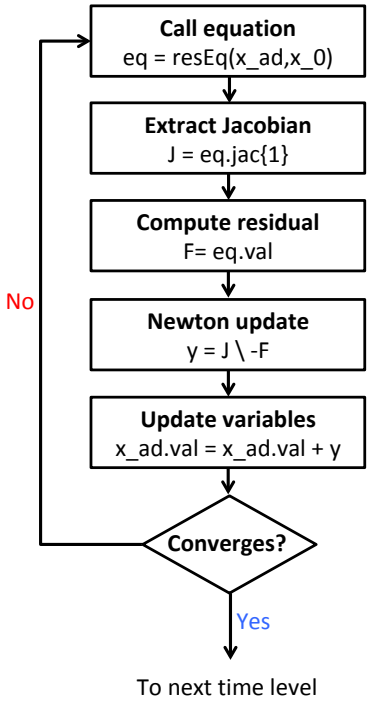


Figure 4.5 Schematic representation of the Newton method applied to a generic non-linear equation.

4.7 Computational implementation

Now that we presented the discrete equations and the general approach to solving them let us briefly introduce the general implementation framework and one example of a simulator based on automatic differentiation for solving the unsaturated flow in non-deformable porous media.

4.7.1 General implementation framework

The computational codes are implemented in MATLAB R2017b where two cores toolboxes are used, MRST 2016a and FV-Biot. MRST is a free open-source software for reservoir modeling and simulation and is mainly intended as a toolbox for rapid prototyping and demonstration of new simulation methods and modeling concepts (*MATLAB Reservoir Simulation Toolbox* 2018). The FV-Biot package implements the discretization of poro-mechanics by cell centered finite volumes methods, providing the discretization of the Darcy flow using MPFA, linear elasticity using MPSA and the Biot equations, i.e. coupling terms for the combined system (*FV-Biot* 2018). In Fig. 4.6 we show the general implementation framework highlighting the central capabilities exploited from these two toolboxes.

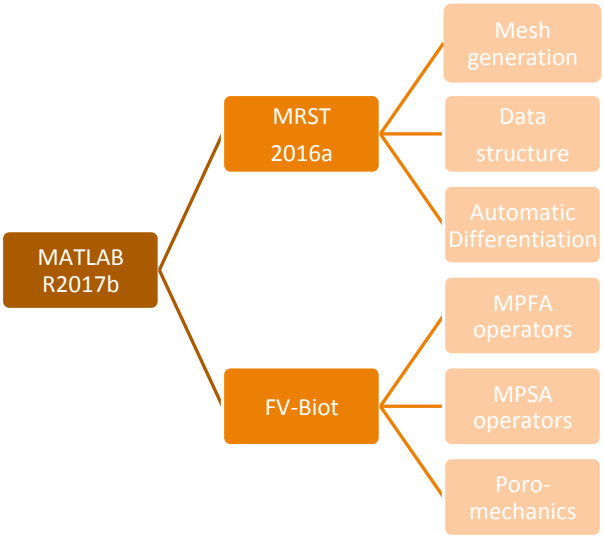


Figure 4.6 General implementation framework.

4.7.2 A simulator based on Automatic Differentiation

In this section, we present the script for solving the mixed-based form of the incompressible Richards' equation using the hydraulic head as the primary variable.

The first step involves setting up the grid. In this case, we use a Cartesian grid with 30 cells in the z-direction and one cell in the x and $y-$directions. The lengths are assumed to be equal to $100\,cm$ for all the axes. The `G` structure contains all the information related to the grid, where we used `cartGrid()` to generate the Cartesian grid and `computeGeomeotry()` to compute the geometry. The volumes of each cell can be extracted from the object `G`.

```
%% Setting up the Grid
nx = 1;                 % Cells in x-direction
ny = 1;                 % Cells in y-direction
nz = 30;                % Cells in z-direction
Lx = 100;               % Lenght in x-direction
Ly = 100;               % Length in y-direction
Lz = 100;               % Length in z-direction
```

```
G = computeGeometry(cartGrid([nx,ny,nz],[Lx,Ly,Lz])); % computing geometry
V = G.cells.volumes;  % Cell volumes
```

The next step consists in declaring the physical properties. First, we declare the parameters of the water. Then, given a saturated hydraulic conductivity, we determine the intrinsic permeability in order to create the `rock.perm` structure. Finally, we declare van Genuchten parameters that describe the water retention curves.

```
%% Fluid properties
rho = 1;                    % [g/cm^3] water density
mu = 0.01;                  % [g/cm.s] water viscosity
g = 980.6650;               % [cm/s^2] gravity acceleration

%% Rock properties
K_sat = 0.00922;            % [cm/s] saturated hydraulic conductivity
k = (K_sat*mu)/(rho*g);     % [cm^2] intrinsic permeability
rock.perm = repmat([k, k, k], [G.cells.num, 1]); % creating perm structure

%% vanGenuchten parameters
alpha = 0.0335;             % [1/cm] Equation parameter
nVan = 2;                   % [-]    Equation parameter
mVan = 1-(1/nVan);          % [-]    Equation parameter
theta_s = 0.368;            % [-]    Saturation soil moisture
theta_r = 0.102;            % [-]    Residual soil moisture
```

The boundary and initial conditions are declared next. First, we extract relevant grid information, and then we determine the indices of the boundaries, i.e. `x_min`, `x_max` and so on. Then, scalars are created to set the values at the boundary conditions. In this case, we assume no flux in all the sides of the domain, except at the top and bottom boundaries where constant hydraulic heads are imposed. The next step requires creating a boundary structure `bc` where the values and type are passed, i.e., flux or pressure. Finally, we create a vector of boundary values `bc_val` where we store the values using the boundary indices previously computed.

```
%% Boundary and Initial Conditions

% Extracting Grid information
fCentr = G.faces.centroids;             % extracting faces centroids
cCentr = G.cells.centroids;             % extracting cells centroids
zCells = cCentr(:,3);                   % centroids of cells in z-direction
zFaces = fCentr(:,3);                   % centroids of faces in z-direction
zetaCells = Lz - zCells;                % centroids of cells of elev. head
zetaFaces = Lz - zFaces;                % centroids of faces of elev. head

% Determining indices of boundaries
x_min = find(fCentr(:,1) == 0);             % indices of West boundary
x_max = find( fCentr(:,1) > 0.9999*Lx & ...
              fCentr(:,1) < 1.0001*Lx );    % indices of East boundary

y_min = find(fCentr(:,2) == 0);             % indices of South boundary
```

```
y_max = find( fCentr(:,2) > 0.9999*Ly & ...
              fCentr(:,2) < 1.0001*Ly );    % indices of North boundary

z_min = find(fCentr(:,3) == 0);             % indices of Top boundary
z_max = find( fCentr(:,3) > 0.9999*Lz & ...
              fCentr(:,3) < 1.0001*Lz );    % indices of Bottom boundary

% Boundary and initial values
fluxW = 0;                                  % [cm^3/s] West boundary
fluxE = 0;                                  % [cm^3/s] East boundary
fluxS = 0;                                  % [cm^3/s] South boundary
fluxN = 0;                                  % [cm^3/s] North boundary
psiT = -75;                                 % [cm] Top pressure head
psiB = -1000;                               % [cm] Bottom pressure head
hT = psiT + zetaFaces(z_min);               % [cm] Top hydraulic head
hB = psiB + zetaFaces(z_max);               % [cm] Bottom hydraulic head
h_init = psiB + zetaCells;                  % [cm] Initial condition

% Creating the boundary structure
bc = addBC([], x_min, 'flux', fluxW);       % setting West boundary
bc = addBC(bc, x_max, 'flux', fluxE);       % setting East boundary
bc = addBC(bc, y_min, 'flux', fluxS);       % setting South boundary
bc = addBC(bc, y_max, 'flux', fluxN);       % setting North boundary
bc = addBC(bc, z_min, 'pressure', hT);      % setting Top boundary
bc = addBC(bc, z_max, 'pressure', hB);      % setting Bottom boundary

% Setting the boundary values vector
bc_val = zeros(G.faces.num, 1);             % initializing
bc_val(x_min) = fluxW;                      % assigning West boundary
bc_val(x_max) = fluxE;                      % assigning East boundary
bc_val(y_min) = fluxS;                      % assigning South boundary
bc_val(y_max) = fluxN;                      % assigning North boundary
bc_val(z_min) = hT;                         % assigning Top boundary
bc_val(z_max) = hB;                         % assigning Bottom boundary
```

The discretization is made invoking the `mpfa()` routine, which uses the information of the grid structure `G`, the rock structure `rock` and the boundary conditions structure `bc`. The discrete MPFA-operators (`F()`,`boundF()` and `divF()`) are created as anonymous functions using the discretization information. The primary vector of unknown `h_ad`is initialized as a special type of object, called AD variable. Finally, the water retention variables are also declared as anonymous functions.

```
%% Calling MPFA routine
mpfa_discr = mpfa(G,rock,[],'bc',bc,'invertBlocks','matlab');

%% Creating discrete mpfa-operators
F = @(x) mpfa_discr.F * x;                  % flux discretization
boundFlux = @(x) mpfa_discr.boundFlux * x;  % boundary discretization
divF = @(x) mpfa_discr.div * x;              % divergence

%% Creating AD variable
```

```
h_ad = initVariablesADI(h_init);

%% Water retention curves

% Boolean function to determine if we are in the unsat or sat zone
isUnsat = @(x) x < 0;

% Water content
theta= @(x)      isUnsat(x) .* ((theta_s - theta_r) ./ ...
                 (1 + (alpha .* abs(x)).^nVan).^mVan + theta_r ) + ...
                 ~isUnsat(x) .* theta_s;

% Specific Moisture Capacity
cVan= @(x)       isUnsat(x) .* ((mVan .* nVan .* x .* (theta_r-theta_s) .* ...
                 alpha.^nVan .* abs(x).^(nVan-2) ) ./ ...
                 (alpha^nVan .* abs(x).^nVan + 1).^(mVan+1)) + ...
                 ~isUnsat(x) .* 0;

% Relative permeability
krw= @(x)        isUnsat(x) .* ((1 - (alpha .* abs(x)).^(nVan-1) .* ...
                 (1 + (alpha .* abs(x)).^nVan).^(-mVan)).^2 ./ ...
                 (1 + (alpha .* abs(x)).^nVan).^(mVan./2)) + ...
                 ~isUnsat(x) .* 1;
```

The time and printing parameters must be declared next. As recommended by (Simunek et al. 2005) we implement an adaptive time stepping algorithm. This algorithm will calculate the next time step size based on the number of iterations needed to reach the desired tolerance at the previous time level. If the number of iterations exceeds an upper optimal iteration range (i.e. 7) the time step is multiplied by an upper multiplication factor < 1 (i.e. 0.7) in order to decrease the time step size. On the other hand, if the number of iterations is less than a lower optimal iteration range (i.e. 3) the time step is multiplied by a lower multiplication factor > 1 (i.e. 1.3) to increase the time step size. However, the minimum and maximum time step sizes are limited by the user, i.e 0.01 and 10,000 s.

```
%% Time parameters
simTime = 72*3600;      % [s]
dt_init = 0.01;         % [s]
dt_min = 0.01;          % [s]
dt_max = 10000;         % [s]
lowerOptIterRange = 3;  % [iter] lower optimal iteration range
upperOptIterRange = 7;  % [iter] upper optimal iteration range
lowerMultFactor = 1.3;  % [-] lower multiplication factor
upperMultFactor = 0.7;  % [-] upper multiplication factor
dt = dt_init;           % [s] initializing time step
timeCum = 0;            % [s] initializing cumulative time
```

Now we are in position to create the discrete governing equations as presented in section 4.5.1. We can clearly see, not only similarities between the continuous and discrete equations but also a smooth analogy in the computational sense. As the reader may note, there are three discrete equations. The

first computes the relative permeabilities using an arithmetic average. The second, uses the relative permeabilities and the discrete operators `F()` and `boundF()` to determine the Darcy Flux. Finally, the mass conservation equation uses the Darcy fluxes and the `divF()` operator to solve for `h_ad`.

```
%% Discrete equations

% Arithmetic Mean
krwAr = @(h_m0) arithmeticMPFA_hydHead(h_m0,G,bc,zetaCells,zetaFaces,krw);

% Darcy Flux
v = @(h,h_m0)  (rho.*g./mu) .* krwAr(h_m0) .* (F(h) + boundFlux(bc_val));

% Mass Conservation
hEq = @(h,h_n0,h_m0,dt) (V./dt) .* (...
                                    theta(h_m0-zetaCells) + ...
                                    cVan(h_m0-zetaCells) .* (h - h_m0) - ...
                                    theta(h_n0-zetaCells) ...
                                    ) + divF(v(h,h_m0));
```

Finally, performing the simulation can be reduced to two while loops. The outer loop is effective as long as the cumulative time is less than the final simulation time. For each time level, the Newton parameters are prescribed, i.e., a maximum number of iterations and desired tolerance. The inner loop actually solves the discrete system for each time step using the Newton method explained in the last section. This loop is effective as long as the tolerance is greater than the desired tolerance *and* the number of iterations is less than the maximum number of iterations. The convergence information is printed in the console if convergence is achieved, otherwise the simulation stops and an error message is displayed. Whenever convergence is reached, a new time step size is determined using the adaptive time stepping algorithm.

```
%% Solving the system of equations

while timeCum < simTime

    % Newton parameters
    absTolPresHead = 100;           % [cm] initializing tolerance
    maxAbsTolPresHead = 1;          % [cm] max absolute tolerance
    iter = 1;                       % initializing iterations
    maxIter = 10;                   % maximum number of iterations
    h_n0 = h_ad.val;                % [cm] current time step h (n-index)
    timeCum = timeCum + dt;         % [s] cumulative time

    % Newton loop
    while (absTolPresHead > maxAbsTolPresHead) && (iter < maxIter)

        h_m0 = h_ad.val;                % current iteration step h (m-index)
        eq = hEq(h_ad,h_n0,h_m0,dt);    % calling equation
        R = eq.val;                     % determing residual
        J = eq.jac{1};                  % determing Jacobian
        y = J\-R;                       % Newton update
        h_ad.val  = h_ad.val + y;       % root for the k-th step
        absTolPresHead = max(abs(h_ad.val - h_m0));
```

```
        % Printing convergence information
        if (absTolPresHead <= maxAbsTolPresHead) && (iter <= maxIter)
             fprintf('Time: %.3f [s] \t Iter: %d \t Error: %d \n',...
                timeCum,iter,absTolPresHead);
        elseif (iter >= maxIter)
            error('Newton method did not converge!');
        end

        iter = iter + 1;                % iteration ++
    end

    % Calling for time stepping routine
    [dt] = timeStepRichards(dt,dt_min,dt_max,simTime,timeCum,iter,...
        lowerOptIterRange,upperOptIterRange,...
        lowerMultFactor,upperMultFactor);

end
```

We would like to finish this chapter pointing out that simulators used for solving coupled problems, i.e., between flow and elasticity, are slightly more complicated. However, the core structure of the implementation remains the same:

1. Create the computational grid.
2. Set the physical parameters.
3. Create the boundary structure and assign the boundary and initial conditions.
4. Discretize the problem using `mpfa` and `mpsa` routines.
5. Create the discrete `mpfa` and `mpsa` operators.
6. Initialize the Automatic Differentiation variable/s.
7. Write the discrete equations in residual form.
8. Set the time parameters.
9. Solve the system of equations using the Newton method for each time step.

Chapter 5

Numerical validations and demonstrations

In this chapter, we present a series of numerical validations and demonstrations in one, two and three dimensions. This chapter aims to show the capabilities of our numerical codes while gaining insight of the physical processes underlying each of the first three parts of this thesis. We validate our results whenever analytical solutions or well-regarded numerical codes are available. The chapter is subdivided into three sections, in each section we present two cases of increasing complexity.

In section 5.1 we study the unsaturated flow in non-deformable porous media. In Case 1, we start by solving the one-dimensional incompressible mixed-based form version of the Richards validating our results with Hydrus-1D (Simunek et al. 2005). In Case 2, we extend the one-dimensional problem to three dimensions where include heterogeneity in the porous media.

Section 5.2 covers two linear elastic problems. In Case 3, we compare the results of the MPSA discretization for an horizontal compression of an isotropic elastic rock (under the plane stress assumption) with the results from the Structural Mechanics PDE-Matlab Toolbox (Matlab 2018). In Case 4 we extend the two-dimensional problem covered in Case 5 to a three-dimensional isotropic elastic body using unstructured grids.

The problems of saturated flow in deformable porous media are presented in section 5.3. Again, we present two numerical examples where we focus on the consolidation process in rock formations, following the Biot's theory of consolidation (von Terzaghi et al. 1925). In Case 5, we solve the well-known one-dimensional Terzaghi's consolidation problem. Moreover, we validate our results with the analytical solution and perform a numerical convergence test. Lastly, in Case 6, we analyze the influence of impervious formations in the consolidation process for a three-dimensional system.

5.1 Unsaturated flow in non-deformable porous media

In this section, we present two numerical examples of the incompressible mixed-base form of the Richards' equation. In Case 1, we solve a one-dimensional benchmark problem validating our results with Hydrus-1D (Simunek et al. 2005), a well-regarded software in the hydrology community (Pinder & Celia 2006). In this test, we study a water infiltration problem for an initially dry soil presenting the water pressure and water content profiles for several times. We also show the information related to the number of iterations per time step and cumulative computational time. Finally, we perform a numerical convergence test where we compute the errors with respect to a Hydrus solution generated using a highly refined grid. In Case 2, we perform a three-dimensional simulation in a heterogeneous porous medium. The heterogeneities are included as impermeable formations within the system. By including theses impervious formations, we show how our code deals with significant differences in permeability. Very often, this represents a challenge for many numerical algorithms in terms of convergence. However, thanks to the mass conservative nature of the finite volume discretization, we achieve excellent results while keeping a low number of Newton iterations and reasonably small computational requirements.

5.1.1 Case 1: One-dimensional unsaturated water infiltration

In this numerical example, we simulate the one-dimensional incompressible version of the Richards' equation and compare our results against the well regarded Hydrus-1D code (Simunek et al. 2005). This test problem has been used extensively to test the capability of numerical codes. The domain is a one-dimensional initially dry soil column of 100 cm of length, after some time a water infiltration front develops from top to bottom due to the higher pressure head imposed at the top boundary. We present the pressure head and water content profiles for several times. Moreover, we show the number of iterations and computational time per time step and perform a numerical convergence test in space.

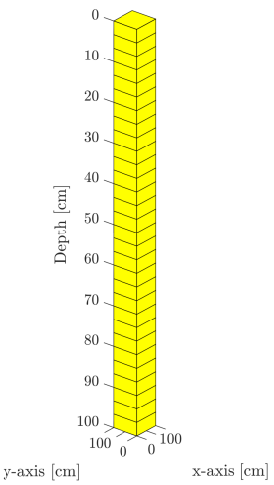


Figure 5.1 Computational grid for a *pseudo*-one-dimensional domain.

Table 5.1 Physical properties

k	9.4018×10^{-8}	cm^2
ρ_w	1	$g \cdot cm^{-3}$
μ_w	0.01	$g \cdot cm^{-1} \cdot s^{-1}$
g	980.66	$cm \cdot s^{-2}$
α_v	0.0335	cm^{-1}
θ_s	0.368	–
θ_r	0.102	–
m_v	0.5	–
n_v	2	–

In Fig. 5.1 we show the *pseudo*-one-dimensional grid that we use to emulate the actual one-dimensional grid. We use the term *pseudo* because we are using a three-dimensional grid with only one cell in the x and y-coordinates. The lengths are equal to 100 cm for all the axes. We use one cell in the x and y direction and 30 cells in the $z-$direction.

The physical parameters used in this problem corresponds to measurements made at a field site in New Mexico and had been taken from (Pinder & Celia 2006), see Table 5.1. The total simulation time is 3 days. Note that we also include the information related to the time stepping control and the iterations criteria in Table 5.2.

The boundary and initial conditions are shown in Table 5.3. We include the boundary conditions (no flux) corresponding to the East, West, South and North sides of the domain. These, of course, are set as no flux to resemble the one-dimensional infiltration process.

Table 5.2 Time and iteration parameters

Initial time	0	s
Final time	$259,200$	s
Initial time step	0.01	s
Minimum time step	0.01	s
Maximum time step	$10,000$	s
Max. number of iterations	10	–
Pressure head tolerance	1	cm
Lower optimal iteration range	3	–
Upper optimal iteration range	7	–
Lower optimal iteration range	3	–
Lower time step mult. factor	1.3	–
Upper time step mult. factor	0.9	–

Table 5.3 Boundary and initial conditions

East	$\underline{q} \cdot \underline{n}$	0	$cm^3 \cdot s^{-1}$
West	$\underline{q} \cdot \underline{n}$	0	$cm^3 \cdot s^{-1}$
South	$\underline{q} \cdot \underline{n}$	0	$cm^3 \cdot s^{-1}$
North	$\underline{q} \cdot \underline{n}$	0	$cm^3 \cdot s^{-1}$
Top	ψ	-75	cm
Bottom	ψ	-1000	cm
Initial	ψ	-1000	cm

In Fig. 5.2 we show the pressure head and water content profiles for four different times. We can see a good agreement with the profiles generated in Hydrus-1D. These curves show how the pressure head increases with time, reflecting the infiltration process and the resulting saturation of the soil column. This process will take place until we reach a hydrostatic equilibrium where the hydraulic head along the z-direction is constant. Once we reach this point, there is no longer any driven force for the flow to take place, and we will enter a steady-state regime.

We also include the graphics concerning the number of iterations per time step and cumulative CPU time in Fig. 5.3. Typically, the time stepping control algorithm will automatically decrease the time step size if the number of iterations is greater than seven in order to ensure a low number of iterations. Regarding the CPU time, we can only say that is practically insignificant, however we must remember that this is a straightforward test case in one dimension.

When analytical or reference solutions are available, it is recommended to perform numerical convergence tests. Unfortunately, analytical solutions for the Richards' equations are only available for a few specific

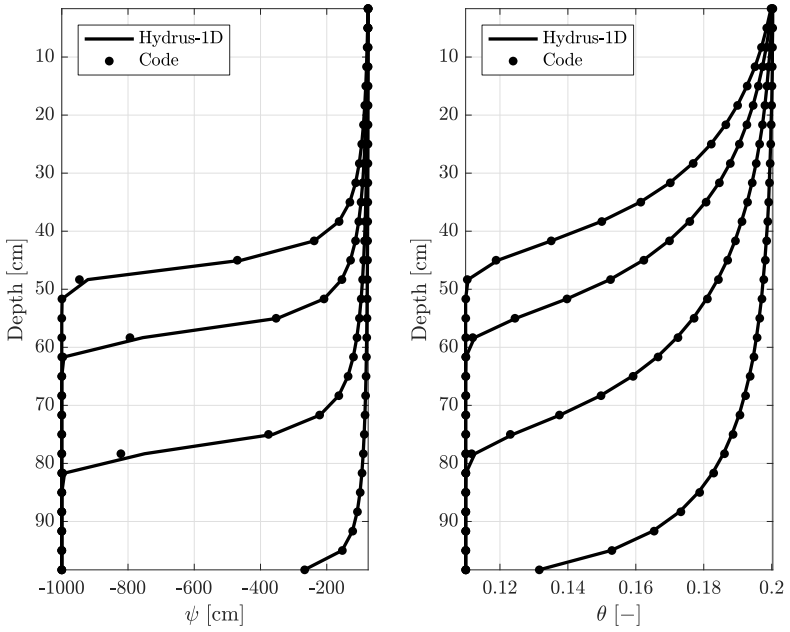


Figure 5.2 Validation with Hydrus-1D showing the pressure head (left) and water content (right) profiles for three days of water infiltration. From top to bottom, we show the profiles for $51,840$ s, $77,760$ s, $129,600$ s and $259,200$ s.

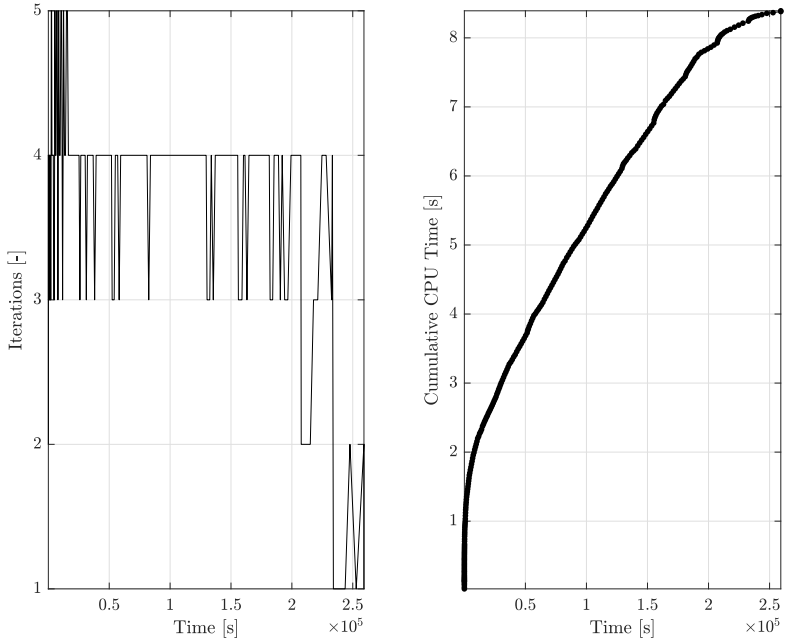


Figure 5.3 Left: Number of iterations per time step. Right: Cumulative CPU time.

forms of constitutive relations describing the soil water retention curves (Rogers et al. 1983, Barry et al. 1991, Sander et al. 1988, Barry & Sander 1991, Barry et al. 1993, Ross & Parlange 1994). However, the applicability of these solutions is limited because their functional forms are different from widely used water retention curves that represent real soils. Nevertheless, we can generate a reference solution in Hydrus-1D using a highly refined mesh, i.e., using 362 nodes.

In Fig. 5.4 (left) we show the pressure head profiles for progressively refined grids and the reference solution for a simulation time of $50,000\ s$. The numerical solution converges in space to the reference solution as the number of cells increases. To measure the error e between our solution and the reference solution, we use the infinite norm $\|\cdot\|_\infty$. Hence, we define the error measured in the infinity norm e_∞ as

$$e_\infty = \|\psi_{\text{app}} - \psi_{\text{ref}}\|_\infty = \sup_{z\in(z_{\min},z_{\max})} |\psi_{\text{app}}(z) - \psi_{\text{ref}}(z)|,$$

where ψ_{ref} and ψ_{app} are the reference and the approximated solution, respectively. Moreover, we say that the numerical method is of order p if there is a number C independent of δz such that

$$|\psi_{\text{app}} - \psi_{\text{ref}}| \leq C(\delta z)^p,$$

for a sufficiently small δz. If the error $\psi_{\text{app}} - \psi_{\text{ref}}$ depends smoothly on δz, then

$$\psi_{\text{app}} - \psi_{\text{ref}} = Ch^p + \mathcal{O}((\delta z)^{p+1}).$$

If we take logarithm at both sides we have

$$\log|\psi_{\text{app}} - \psi_{\text{ref}}| = \log|C| + p\log(\delta z) + \mathcal{O}(h).$$

By fitting a line and determining the slope of the log-log plot of the error *vs.* the step size we can determine the order of convergence of the numerical approximation. $p < 1$ denotes a sublinear convergence, $p = 1$ linear convergence, $p = 2$ quadratic convergence, and so on. As we can see in the right plot of Fig. (5.4) we have a sublinear convergence with $p = 0.6$. The low order of convergence may be caused by the modified Picard iteration implemented to ensure mass conservation and the highly non-linear water retention curves. Further analyses must be carried out to determine the range of convergence and stability of the Newton method in the context of MPFA-FV.

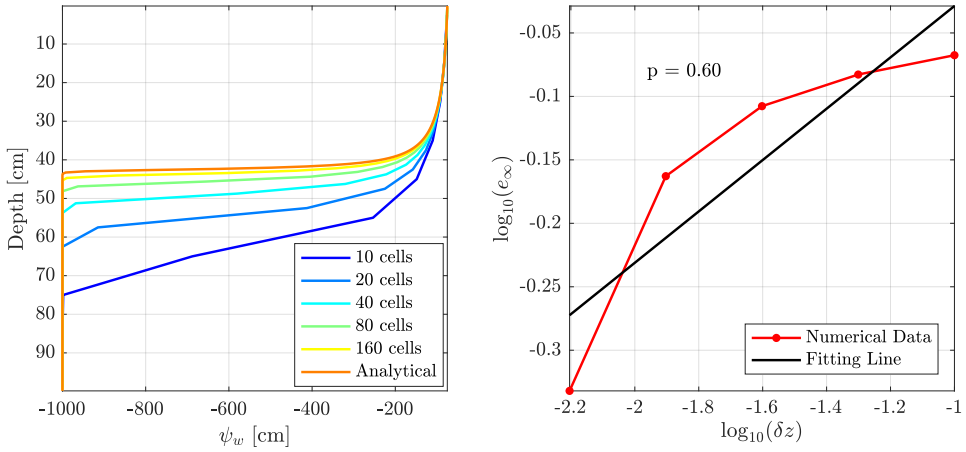


Figure 5.4 Left: Spatial convergence of pressure head w.r.t. the reference solution for $t = 50,000\ s$. Right: Log-log plot showing a sublinear order of convergence.

5.1.2 Case 2: Three-dimensional simulation for a heterogeneous porous media

For this case, we pretend to show the capabilities of the code to deal with heterogeneity within the domain. Therefore, we include two impermeable layers as shown in Fig.5.5. Since we are dealing with a three-dimensional domain, it would be hard to visualize the entire permeability field, consequently, we only show the impermeable layers, the rest of the layers are assumed to be permeable. For this case, we use a grid with 20 cells in the x and y-directions and 80 cells in the z-direction where the infiltration takes place.

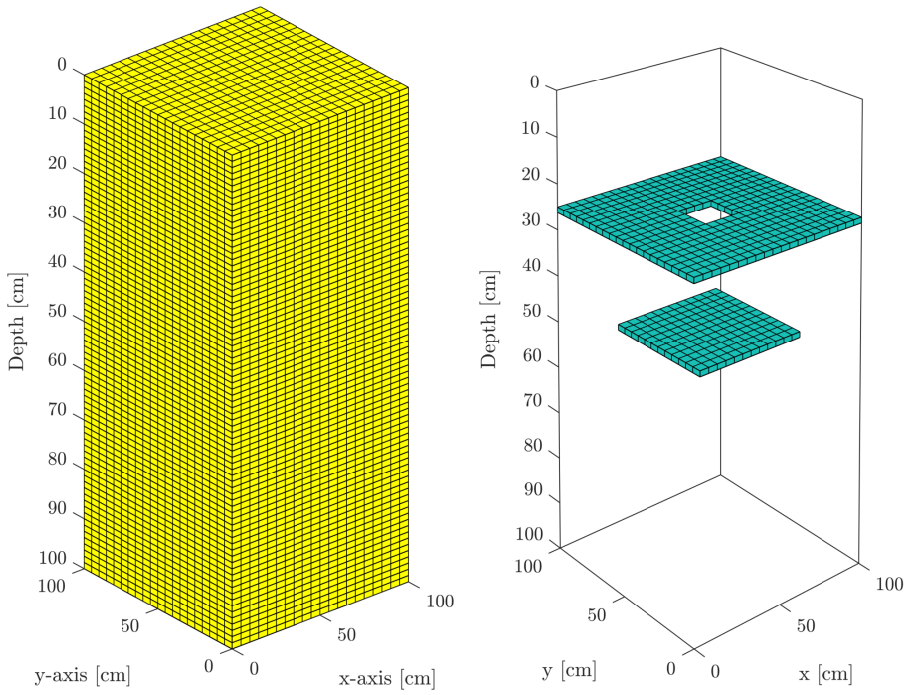


Figure 5.5 Computational grid (left) and permeability field (right) showing the location of the impermeable formations. The lengths of the domain are equal to 100 cm for all the axes. We use 20 cells in the x and $y-$ directions and 80 cells in the $z-$direction.

The physical parameters are the same as in Case 1, except for the permeability of the impermeable rocks which are set to 9.4018×10^{-28} cm^2. Note that we assigned a significantly low value to test the code. For the time stepping control algorithm, we use an initial and minimum time step of 10 s to speed up the simulation, the rest of the parameters are the same as in Table 5.2. Finally, the boundary conditions are the same as in Case 1, see Table 5.3.

Figure 5.7 shows the water content for six different times. For 7.20 h the water starts to infiltrate from the top and then deviate the zones of low permeability to pass through the permeable center hole. For 14.40 h the infiltration proceeds and more water enters the domain, always avoiding the impermeable layer. After 21.60 h the water front encounters another impermeable region and spreads out laterally to avoid it. Finally, for the rest of the simulation, the water front still develops symmetrically taking the high permeability paths. However, the domain is still dry at the bottom after $72,00$ h as a consequence of the inclusion of the impervious layers.

Unlike Case 1, for this three-dimensional case, both the number of iterations and the CPU time increased. We reached 7 iterations at the beginning of the simulation. This is a consequence of including heterogeneity within the domain and increasing the initial and minimum time steps. Regarding the CPU time, we can see an abrupt jump compared to Case 1. It takes an average of 3.5 s to solve each time step and nearly 1500 s to complete the simulation. This is clearly due to the higher requirement needed to compute the Jacobian and the size of the resulting linear system of a three-dimensional domain.

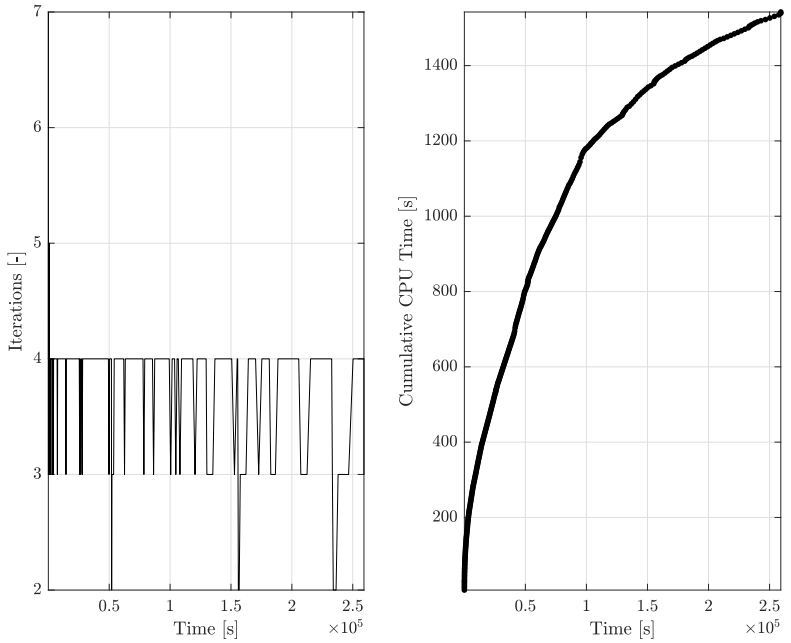


Figure 5.6 Number of iterations per time step (left) and cumulative CPU time (right).

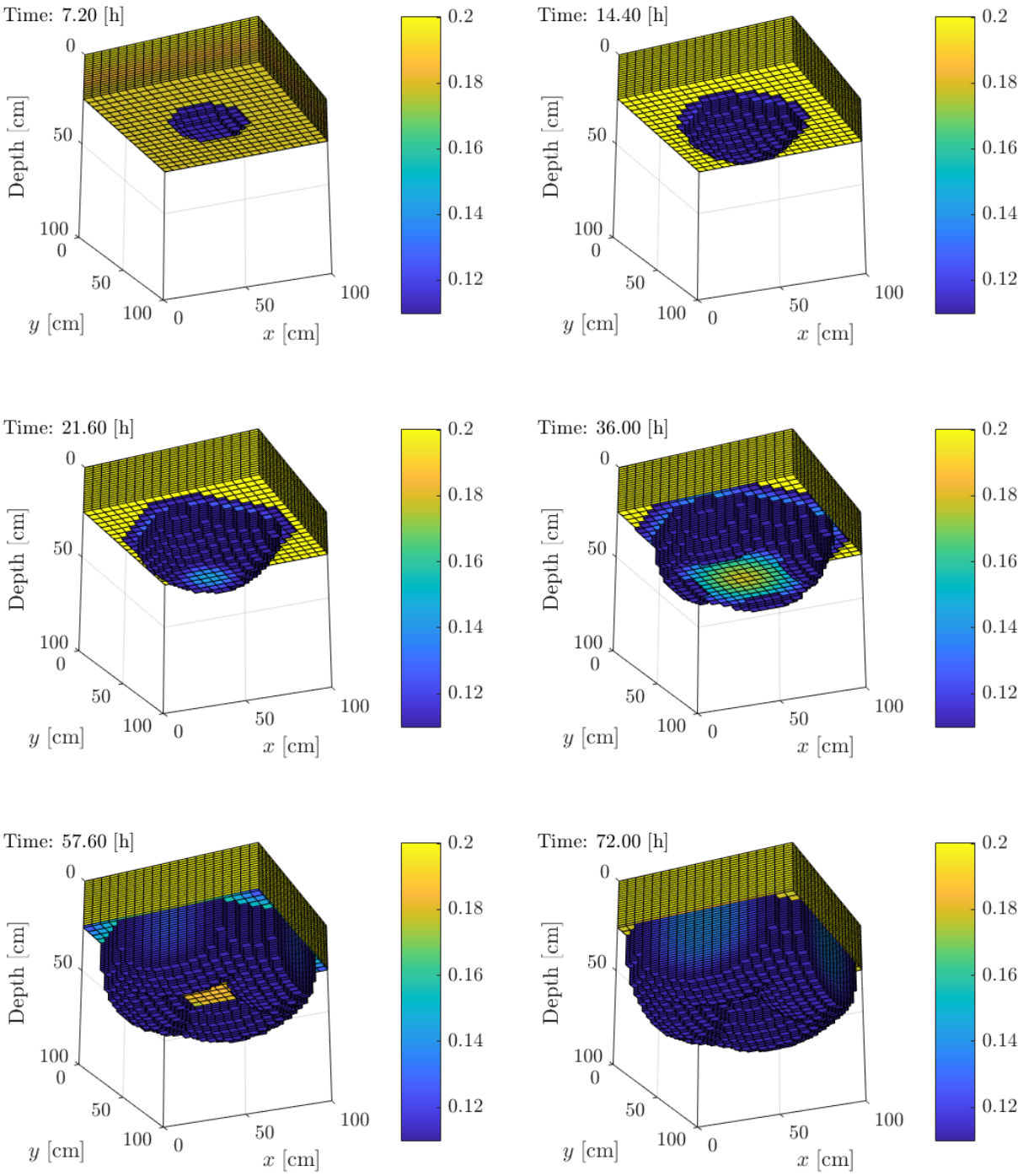


Figure 5.7 Water content for several time steps showing the water infiltration from the top.

5.2 Linear Elasticity

In this section, we present two numerical study cases of two and three-dimensional static linear elasticity problems.

In Case 3, we study a horizontal compression of an isotropic solid under the plane stress assumption, i.e., neglecting the contribution of the stress components in the $z-$direction. We present the horizontal and vertical displacement fields making a quantitative comparison with the results obtained from the PDE Toolbox - Structural Mechanics module from MATLAB R2018a (Matlab 2018). Since we use the same triangular grid generated by the Structural Mechanics module, we can compute the absolute error for the horizontal and vertical displacement by comparing them at the cell centers of each triangular element. This requires interpolation of the displacements obtained from the Structural Mechanics Module, since this module uses Finite Element Method, where the unknowns are located at the vertices of the triangles rather than at the cell centers. Despite this fact, we found a good agreement between the two results. Finally, in order to have an overall measure of the error, we compute the mean squared error for u_x and u_y using as the reference solution the interpolated values evaluated at the cell centers obtained from the Structural Mechanics Module. Finally, in Case 4, we simulate a horizontal compression of a three-dimensional isotropic solid. This numerical test aims to show the capability of the code to handle three-dimensional unstructured grids, i.e. extruded hexagonal grids. We show the displacement fields for the $x-y$, $x-z$ and $y-z$ planes. The results shows that under the simulated conditions, the symmetry is preserved.

5.2.1 Case 3: Horizontal compression under the plane stress assumption.

In this numerical example we analyze the horizontal compression of an elastic body under the plane stress assumption, i.e., the contribution of the stress components in the z-direction are zero. The elastic body is assumed to be isotropic, and we neglect any body forces contributions. To show the validity of our results we used the PDE Toolbox - Structural Mechanics module from MATLAB R2018a (Matlab 2018) where we computed the absolute error for the vertical and horizontal displacements at the cell centers. The objective of this numerical test is to demonstrate the applicability of the code rather than performing a convergence analysis. If the reader is interested in errors and convergence analysis using MPSA for the discretization of the linear elasticity equations we refer to Keilegavlen & Nordbotten (2017), where a series of rigorous tests have been performed to prove the robustness of the MPSA discretization.

The geometry of the domain is a unit square which has been discretized using the FEM mesh generator from MATLAB. The generated mesh has 732 cells, with a maximum element size of 0.0566 m and a minimum element size of 0.0283 m. The physical parameters are shown in Table 5.4. We assume that both Lamé parameters are constant throughout the domain. These parameters correspond to granite at 600 MPa and 20 °C.

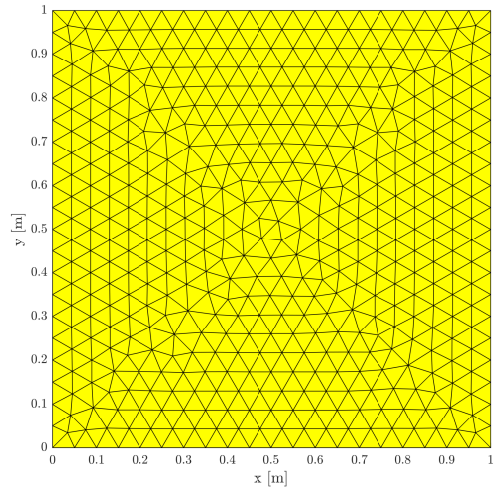


Figure 5.8 Computational grid.

Table 5.4 Physical properties

λ_s	3.44452×10^{10}	Pa
μ_s	3.1212×10^{10}	Pa
ρ_s	$2,700$	$kg \cdot m^{-3}$

We show the boundary and initial conditions in Table 5.5. Note that we stipulate a constant (positive) deformation along the $x-$coordinate in the East side of the domain which corresponds to a compression force, similarly, we imposed a constant (negative) deformation along the $x-$coordinate in the West side of the domain, which again represents a compression force. The initial conditions are needed just as an initial guess for the non-linear solver. This is not entirely necessary since the equations are linear and elliptic. However, we prefer to use an iterative solver (actually the same classic Newton scheme utilized in the previous section) since this strategy will be needed when we address the coupling between flow and deformation in the next sections.

Table 5.5 Boundary and initial conditions.

East	u_x	0.001	m
	u_y	0	m
West	u_x	-0.001	m
	u_y	0	m
South	T_x	0	Pa
	T_y	0	Pa
North	T_x	0	Pa
	T_y	0	Pa
Initial	u_x	0	m
	u_y	0	m

In Fig. 5.9 we show the deformation fields for the horizontal compression of the two-dimensional elastic body in the absence of body forces. We can see the agreement between the results of our code and the results from the Structural Mechanics Module from the PDE-Toolbox of MATLAB. The deformation near the East and West boundaries is maximum and minimum in the center of the domain. Note how in the absence of body forces the deformation field is perfectly symmetric.

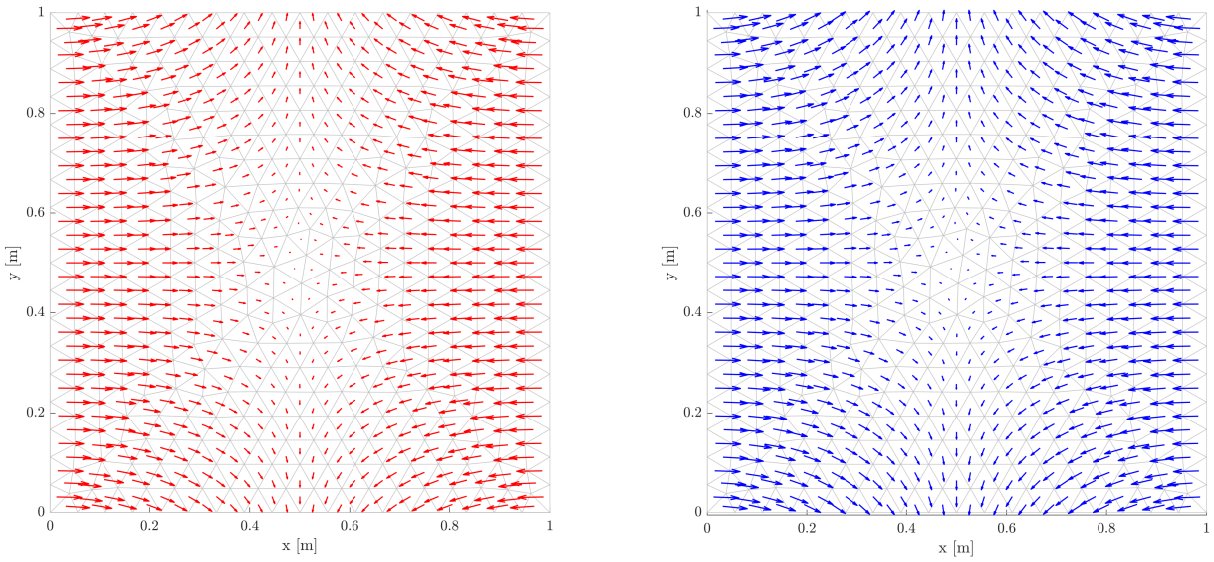


Figure 5.9 Displacement fields for a horizontal compression of a two-dimensional elastic body in the absence of body forces. Results from the MATLAB PDE-Toolbox (left) and our code (right).

In Fig. 5.10 we show the absolute errors for the horizontal and vertical displacements. The errors e_x and e_y are defined as

$$e_x = \left| u_{x_{\text{PDE}}}(\mathtt{c}) - u_{x_{\text{app}}}(\mathtt{c}) \right| \quad \forall \quad \mathtt{c} \in (1, \mathtt{N_c}),$$
$$e_y = \left| u_{y_{\text{PDE}}}(\mathtt{c}) - u_{y_{\text{app}}}(\mathtt{c}) \right| \quad \forall \quad \mathtt{c} \in (1, \mathtt{N_c}),$$

where $u_{x_{\text{PDE}}}(\mathtt{c})$ and $u_{x_{\text{app}}}(\mathtt{c})$ represent the values of the reference and approximated horizontal displacements evaluated at the cell centers, $u_{y_{\text{PDE}}}(\mathtt{c})$ and $u_{y_{\text{app}}}(\mathtt{c})$ are the values of the reference and approximated vertical displacements evaluated at the cell centers, and $\mathtt{N_c}$ the total number of cell centers.

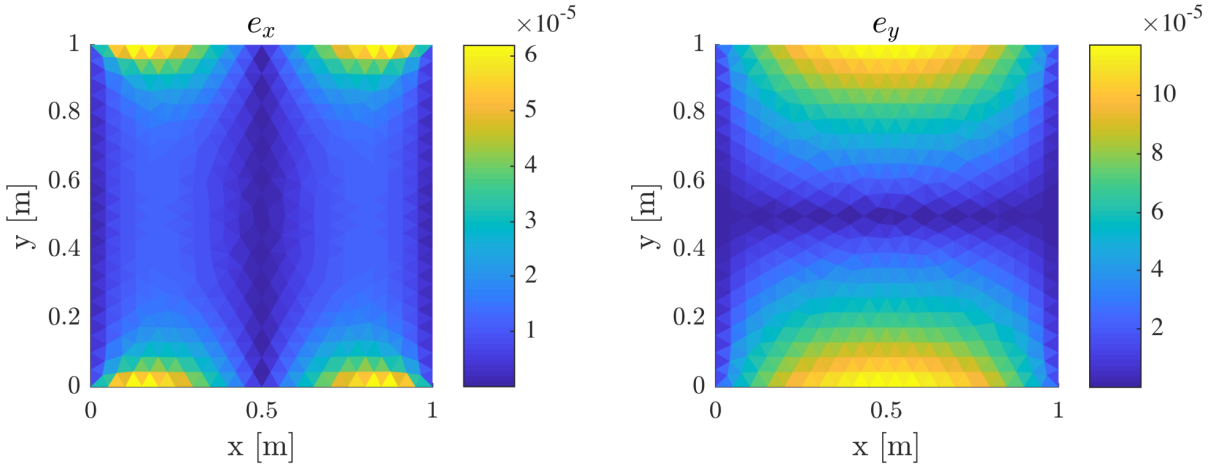


Figure 5.10 Absolute errors e_x and e_y for the horizontal and vertical displacements.

It can be noted that the biggest errors are located nearby the boundaries. The source of the differences could be a consequence of the interpolation necessary to evaluate the reference solution at the cell centers. The PDE-toolbox uses FEM to solve the governing equations and the primary unknowns are evaluated at the vertices of the elements (nodes) rather than at the cell centers.

To have a global measure of the error, we employ the mean squared error which is a measure of the quality of the estimation. Values closer to zero reflects a good approximation between the approximate solution and the reference solution. We define the mean squared errors as

$$\text{mse}_x = \frac{1}{N}\sum_{\mathsf{c}=1}^{N}\left(\left|u_{x_{\text{PDE}}}(\mathsf{c}) - u_{x_{\text{app}}}(\mathsf{c})\right|\right)^2$$
$$\text{mse}_y = \frac{1}{N}\sum_{\mathsf{c}=1}^{N}\left(\left|u_{y_{\text{PDE}}}(\mathsf{c}) - u_{y_{\text{app}}}(\mathsf{c})\right|\right)^2$$

where mse_x and mse_y are the mean squared errors of the horizontal and vertical displacements. The results show that $\text{mse}_x = 0.0341 \times 10^{-8}$ and $\text{mse}_y = 0.2865 \times 10^{-8}$ which suggest a very good approximation.

5.2.2 Case 4: Compression in an extruded hexagonal grid.

For the fourth case, we solve a three-dimensional linear elasticity problem in an extruded hexagonal grid as shown in Fig. 5.11. The purpose of this case is to show the flexibility of our code solving three-dimensional elasticity problems in unstructured grids. The domain has five cells in the $z-$direction and 132 cells in the $x-y$ plane. Note that most of the cells are complete regular hexagons. However there are some irregular quadrilaterals and pentagons at the corners and borders of the domain. The lengths of the domain are 10 m in the x and $y-$axes and 5 m in the $z-$direction.

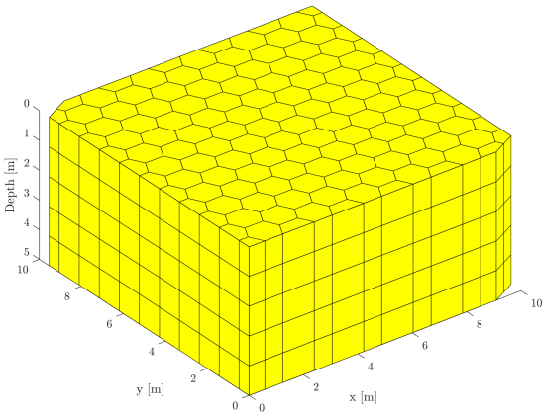


Figure 5.11 Computational grid.

Table 5.6 Boundary and initial conditions

East	u_x	0.25	m
	$u_y = u_z$	0	m
West	u_x	-0.25	m
	$u_y = u_z$	0	m
South	$T_x = T_y = T_z$	0	Pa
North	$T_x = T_y = T_z$	0	Pa
Bottom	$T_x = T_y = T_z$	0	Pa
Top	$T_x = T_y = T_z$	0	Pa
Initial	$u_x = u_y = u_z$	0	m

The physical parameters are shown in Table 5.4 whereas the boundary conditions are in Table 5.6. Note that in a three-dimensional problem we must specify the deformation and the traction force in the three axes x, y, and z. This case is a generalization of Case 4. We establish a constant horizontal deformation of equal magnitude and opposite sign at the East and West side of the domain, i.e., 0.25 m. For these sides, all the other deformations naming u_y, and u_z are set to zero. Moreover, the other sides of the domain are set as traction free, where T_x, T_y, and T_z are set to zero.

In Fig. 5.12 we show the deformation field for the horizontal compression (fixed displacement). In the left column, we can see the position of the original centroids (in gray) and the position of the deformed

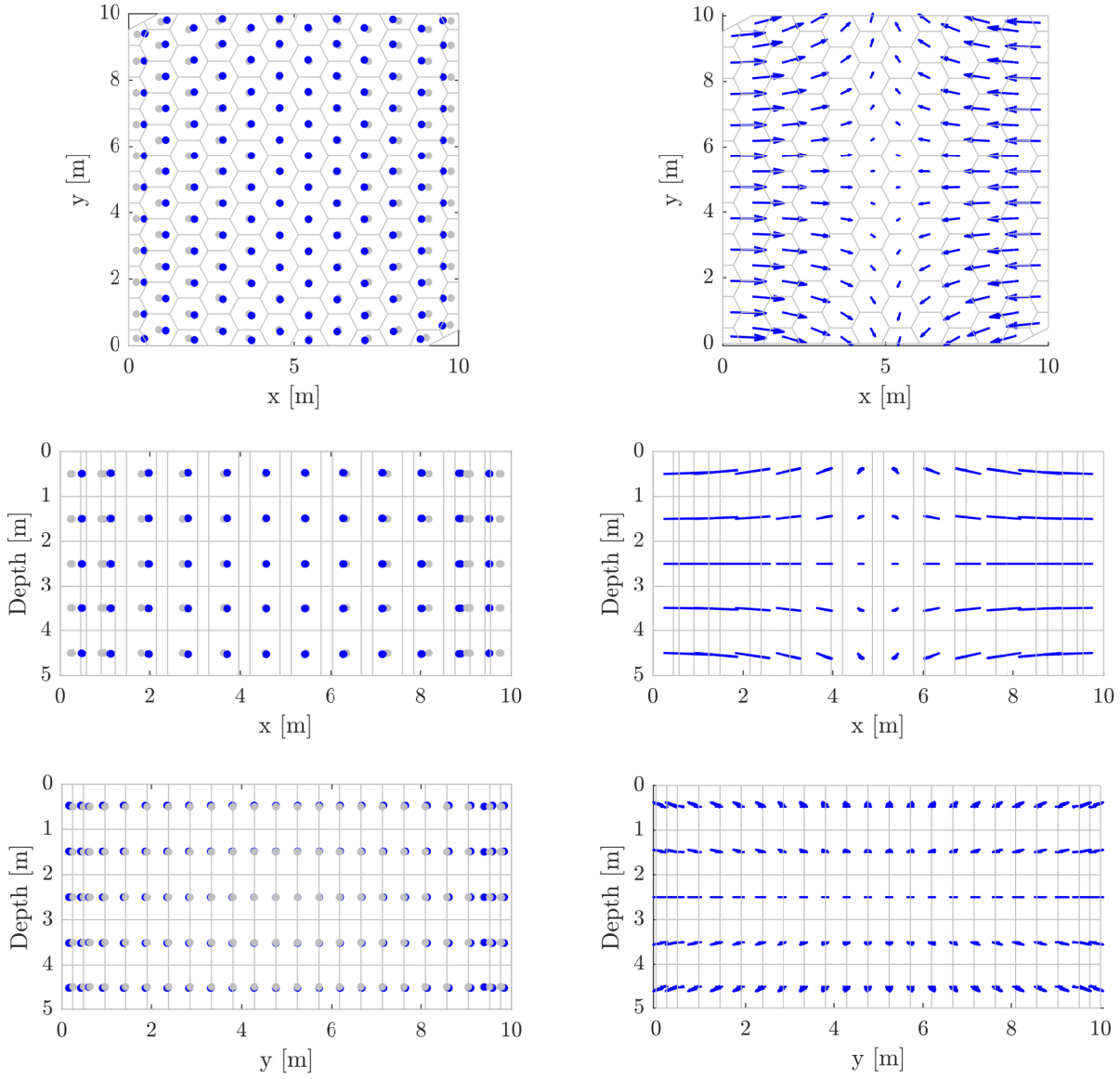


Figure 5.12 Displacement field showing the location of the original centroids (gray) and displaced centroids (blue) for the $y-x$, $z-x$ and $z-y$ planes (left column) and the quiver lines corresponding to the mentioned planes (right column).

centroids (in blue). Similarly, we show in the right column the quiver lines that obey the deformation field. For the $x-y$ plane, we can see how the centroids show their maximum displacement at the East and West sides of the domain. The nearest cells from the East and West sides follow an almost strict horizontal displacement, whereas its neighbors start to displace to the Top and Bottom sides progressively, where the vertical contribution increases as the horizontal contribution decrease. For the $x-z$ plane an almost identical situation described for the $x-y$ plane. Maybe we can point out how the symmetry is preserved by noting that the displacement corresponding to the third layer of the domain does not have a vertical contribution. Finally, the fields in the $y-z$ plane show that indeed a compressive force is acting on the body, note how all the centroids displaced from outside to inside the domain.

5.3 Saturated flow in deformable porous media

In this subsection, we study the saturated flow in deformable porous media following the Biot's theory of three-dimensional consolidation (Biot 1941). We can define consolidation as the process where a reduction in volume and a consequent expulsion of water can be verified when long-term static loads are applied to the soil.

In Case 5 we study Terzaghi's one-dimensional consolidation problem. If compressibility effects are neglected, Terzaghi's problem provides an exact solution for the consolidation process in terms of non-dimensional variables. Due to its simplicity, it has been used extensively to test the validity of numerical codes. We present the non-dimensional pore water pressure profiles for several non-dimensional times. Furthermore, we perform a numerical convergence test in space to show the order of convergence of our numerical method. In Case 6, we present a three-dimensional consolidation process for a heterogeneous porous medium accounting with body forces and compressibility. The system consists of two rock formations with marked differences in their physical properties, i.e., Bose sandstone and Tennessee marble. We show how the inclusion of marble layer located at the middle of the domain considerably retards the consolidation process.

5.3.1 Case 5: Terzaghi's consolidation problem

Terzaghi's problem is a well known one-dimensional consolidation process (von Terzaghi 1923, von Terzaghi et al. 1925). Generally, when soils are subjected to a vertical load, the porosity decreases, resulting in less available space for the pore water. The water within the pores can be expelled, however, in certain types of soils (especially clayey soils) this process may take some time due to their low permeability. The described process is called consolidation.

Since we are dealing with a one-dimensional process, we must emulate a one-dimensional computational grid with a two-dimensional grid. For flow problems, this can be done by setting no flux boundary conditions at the sides of the domain. However, for elasticity problems, we solve a vector equation rather than a scalar equation. Therefore, assuming zero displacement at the sides does not resemble a one-dimensional process. Nevertheless, there are two options to cope with this issue. The first option would be to set up roller boundary conditions, where the perpendicular stress contribution would be zero in the interested direction, i.e. the $x-$coordinate if we are solving a vertical consolidation problem. The second option is to assume a huge extension of the horizontal axis, by doing this we can use a zero displacement condition at the sides of the domain since the horizontal stress contribution at these sides will be negligible. In other words, the horizontal contribution will not have an impact on the vertical process since the boundaries are too far away. Between these two options, the preferred one should be the roller boundary conditions. However, to use them, we must mix a traction vector condition with a fixed displacement condition. Unfortunately, this option is not yet available within the FV-Biot framework. Therefore we stick to the second option to emulate the one-dimensional consolidation process.

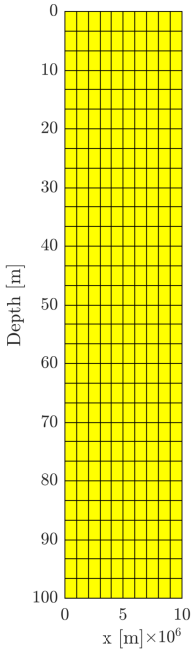


Figure 5.13 Computational grid for the Terzaghi's consolidation problem. The domain has 10 cells in the x-axis and 30 cells in the y-axis. The lengths are 100 m in the vertical direction and $10,000,000$ m in the horizontal direction.

The *pseudo*-one-dimensional grid is shown in Fig. 5.13. We use a simple two-dimensional Cartesian grid with 10 cells in the $x-$direction and 30 cells in the z-direction. The lengths of the domain are 100 m in the $z-$direction and $10,000,000$ m in the $x-$direction. Note that the large horizontal distance is needed to avoid any horizontal stress contribution to the vertical process of consolidation.

Whenever we are dealing with consolidation problems, there are certain physical parameters used specifically to describe the process. Furthermore, one of the remarkable advantages of Terzaghi's solution is that is represented using purely non-dimensional parameters. This allows to use non-realistic physical parameters, since, by combining them we will have a non-dimensional quantity. Let us first present the physical parameters.

For the elasticity problems presented in the last section, we employed the first and second Lamé parameters, λ_s and μ_s together with the rock density ρ_s. However, in soil mechanics, the *shear modulus* G and *bulk modulus* K_s are often preferred since they describe the only two possible deformations, compression and, shear. Let us revisit the relationship between the Lamé parameters

$$G_s = \mu_s \qquad K_s = \frac{2}{3}\mu_s + \lambda_s.$$

In many problems of consolidation, the stiffness of the soil can be characterized by the *confined compressibility* m_c. This parameter is often measured in the laboratory by employing a confined compression test or an oedometer test. For a vertical deformation with lateral confinement, it can be shown that

$$m_c = \frac{1}{K_s + \frac{4}{3}G_s}.$$

Finally, let us introduce *the coefficient of consolidation* c_c which is a measure of the tendency of the soil to get consolidated. Naturally, c_c depends on the rock as well as on the fluid parameters

$$c_c = \frac{K_{sat}}{\rho_w g \left(\alpha^2 m_c + S_\varepsilon\right)}.$$

The boundary conditions for the consolidation problem are shown in Table 5.8. The East and West boundaries are set as no flux and zero deformation. The Bottom boundary is also sealed to the flux and constrained to the deformation. At the Top boundary, a constant vertical load of 10 Pa is imposed whereas the water pressure is assumed to be zero. Initially the soil is undeformed, and the water pressure instantaneously equals the vertical load.

Table 5.7 Physical properties

Rock	λ_s	1	Pa
	μ_s	1	Pa
	G_s	1	Pa
	K_s	1.6667	Pa
	k	1	m^2
Fluid	ρ_w	1	$kg \cdot m^{-3}$
	μ_w	1	$Pa \cdot s$
	g	1	$m \cdot s^{-2}$
Porous medium	K_{sat}	1	$m \cdot s^{-1}$
	S_ε	0	Pa^{-1}
	α	1	$[-]$
Consolidation	m_c	0.3333	Pa^{-1}
	c_c	3	$m^2 \cdot s^{-1}$

Table 5.8 Boundary and initial conditions.

East	$u_x = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
West	$u_x = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Bottom	$u_x = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Top	T_x	0	Pa
	T_z	10	Pa
	p_w	0	Pa
Initial	u_x	0	m
	u_z	0	m
	p_w	10	Pa

The time parameters are shown in Table 5.8. The consolidation process takes place within the first $2,000$ s of applying the vertical load. We use a simple linear time stepping procedure since we are only dealing with the Biot's equations which are linear. This will not be the case if we couple the Richards' equation with the elasticity equations, which is subject of the next chapter. In the meantime, it is sufficiently safe to use a linear time discretization.

Table 5.9 Time and iteration parameters

Initial time	0	s
Final time	$2,000$	s
Time step	25	s
Maximum number of iterations	30	–
Absolute tolerance	1×10^{-10}	–

Before showing the results, let us first introduce the non-dimensional variables as presented in (Verruijt 2013). The non-dimensional water pressure $\mathfrak{p}_w$, the non-dimensional vertical length $\mathfrak{z}$, and the non-

dimensional time τ are defined as

$$\mathfrak{p}_w = \frac{p_w}{p_{w_0}}, \qquad \mathfrak{z} = \frac{z}{L_z}, \qquad \tau = \frac{c_c t}{{L_z}^2}.$$

Now we are in position to introduce the analytical solution derived by Terzaghi. This, in fact, can be done purely in terms of the three non-dimensional variables presented above. The differential equation for the one-dimensional problem can be stated as

$$\begin{aligned} \frac{\partial p_w}{\partial t} &= c_c \frac{\partial^2 p_w}{\partial z^2}, \\ p_w &= p_{w_0} \quad \text{for} \quad t = 0, \\ p_w &= p_{w_0} \quad \text{for} \quad z = 0, \\ \frac{\partial p_w}{\partial z} &= 0 \qquad \text{for} \quad z = L_z. \end{aligned}$$

The solution to this problem can be solved by separation of variables or Laplace transform. We referred to Churchill (1972) or Carslaw & Jaeger (1948) for an excellent introduction to these methods. We prefer not to develop the details regarding the analytical solution and state the final result

$$\mathfrak{p}_w = \frac{4}{\pi} \sum_{j=1}^{\infty} \frac{(-1)^{j-1}}{2j-1} \cos\left[(2j-1)\,\frac{\pi}{2}\mathfrak{z}\right] \exp\left[-(2j-1)^2\,\frac{\pi^2}{4}\tau\right]. \tag{5.1}$$

In Fig. 5.14 we show the non-dimensional pressure profiles for four different non-dimensional times. Initially, the water pressure equals the vertical load imposed at the top of the column. When the consolidation process takes place, water is released from the soil while simultaneously gets deformed due to the compressive force. Therefore, we see a reduction in the pore water pressure as time evolves. This system will reach its final state at $\tau = 1$ when all the water pressure becomes zero along the soil column.

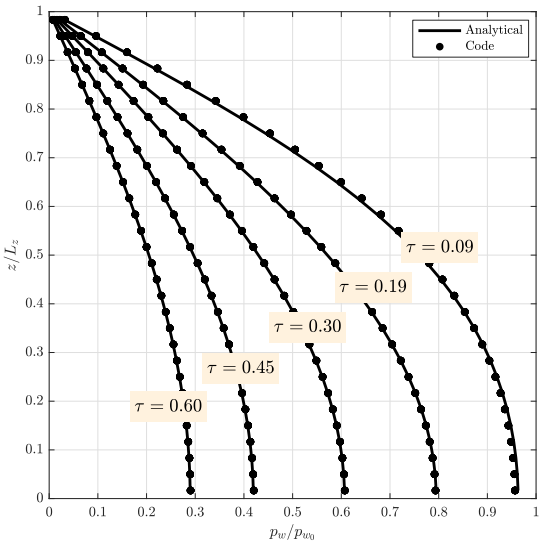


Figure 5.14 Validation of one-dimensional consolidation problem. Profile of dimensionless water pressure $\mathfrak{p}_w$ for several dimensionless times τ.

Counting with the analytical solution allow us to perform a numerical convergence test, similar to the one presented in Case 1. We measure the errors in the infinity norm with respect to the analytical solution for several grid refinement levels, i.e., using 10, 20, 40, 80 and 160 cells in the z-direction. The dimensionless water pressure profiles that we show corresponds to $\tau = 0.3$. In the left plot of Fig. 5.15 we show how the approximated solutions converge to the analytical solution as the number of cells in the $z-$direction increases.

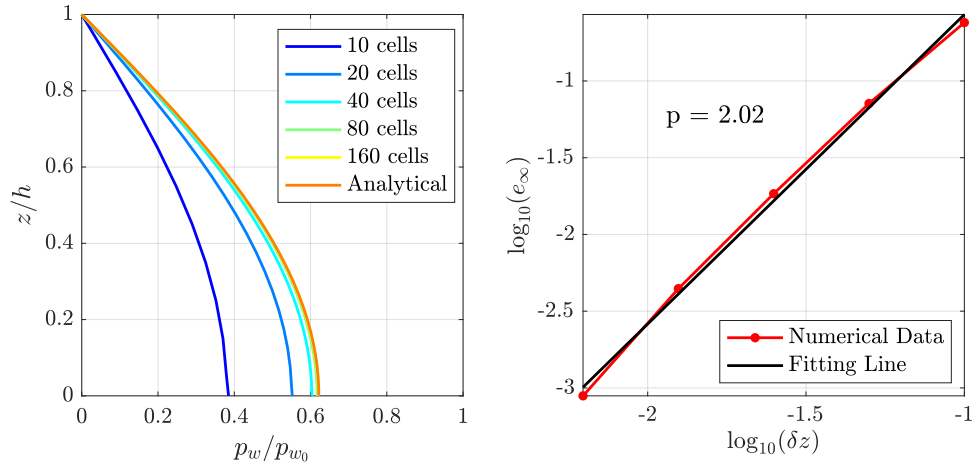


Figure 5.15 Numerical convergence test.

By fitting a line and determining the slope of the log-log plot of the error *vs.* the step size we can determine the order of convergence of the numerical approximation. As we can see in the right plot of Fig. (5.15) we have a quadratic convergence in space. This goes in agreement with theoretical results encountered in (Nordbotten 2016).

5.3.2 Case 6: Three-dimensional consolidation in a heterogeneous porous media

In this case, we consider a three-dimensional consolidation process including heterogeneity in the porous medium. The soil under consideration is made up of two types of rocks, Boise sandstone, and Tennessee marble. The computational grid and the location of the marble formation are shown in Fig. 5.16. We include two layers of marble of 0.25 m of thickness at the center of the domain. Note also that these formations have a flaw in their center, this will critically impact the consolidation process since they will act as a relieve mechanism for the pressure.

The physical parameters for the two types of rocks under consideration can be seen in Table 5.10. Note the significant difference in permeability and porosity, as well as Biot coefficient between the two rocks.

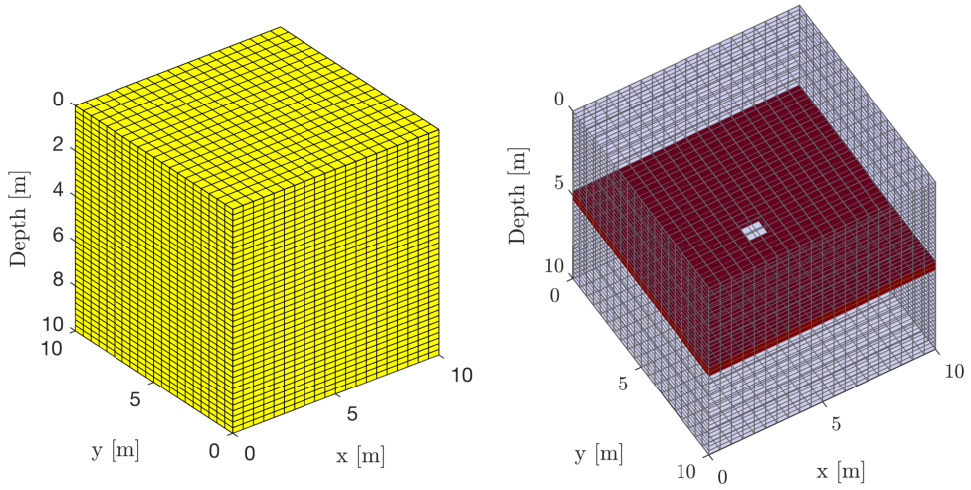


Figure 5.16 Left: Three dimensional computational grid (left) with 20 cells in the x and y-directions and 40 cells in the z -direction. Right: Location of the marble layer (in red) in the middle of the domain with a fault in the center.

Table 5.10 Physical properties

Boise sandstone	λ_s	3.5067×10^{10}	Pa
	μ_s	4.2×10^{9}	Pa
	ρ_s	$2,420$	$kg \cdot m^{-3}$
	C_s	2.3810×10^{-11}	Pa^{-1}
	n	0.26	$-$
	k	7.8954×10^{-13}	m^2
	C_m	2.1739×10^{-10}	Pa^{-1}
	α	0.895	$-$
	S_ε	1.3319×10^{-10}	Pa^{-1}
Tennessee marble	λ_s	2.8174×10^{10}	Pa
	μ_s	2.4×10^{10}	Pa
	ρ_s	$2,770$	$kg \cdot m^{-3}$
	C_s	2.0×10^{-11}	Pa^{-1}
	n	0.02	$-$
	k	9.8692×10^{-20}	m^2
	C_m	2.5×10^{-10}	Pa^{-1}
	α	0.200	$-$
	S_ε	1.2691×10^{-11}	Pa^{-1}

Table 5.11 Boundary and initial conditions.

East	$u_x = u_y = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
West	$u_x = u_y = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
South	$u_x = u_y = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
North	$u_x = u_y = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Top	$T_x = T_y$	0	Pa
	T_z	10×10^6	Pa
	p_w	0	Pa
Bottom	$u_x = u_y = u_z$	0	m
	$\underline{q} \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Initial	$u_x = u_y = u_z$	0	m
	p_w	10×10^6	Pa

The boundary and initial conditions can be seen in Table 5.11. We consider that a constant vertical load in the z-coordinate is imposed at the Top boundary where the water pressure is set to zero. All the other sides are set as no flux and zero displacement. The initial condition considers an undeformed soil and the water pressure is assumed to equal the vertical load. The consolidation process takes place for four hours divided into 20 time steps of 720 s.

The results are shown in Fig. 5.17. In the left column, we show the pressure distribution and in the right column the vertical displacement for four different times. Note that we are only plotting the cells

for which the pressure is less than 8.5 MPa.

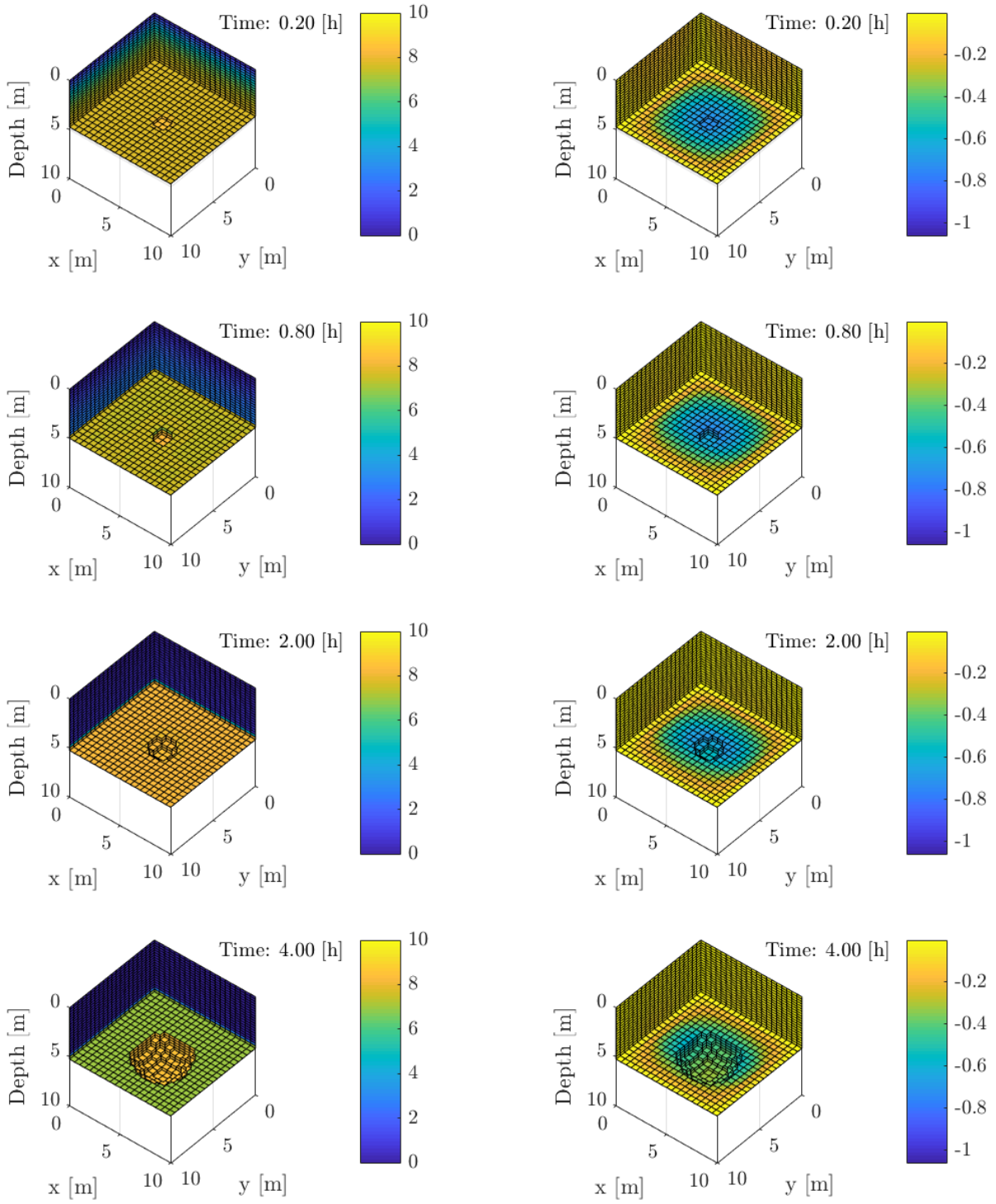


Figure 5.17 Water pressure in MPa (first column) and vertical deformation in mm (second column) for a consolidation process in a three-dimensional heterogeneous domain.

For 0.20 h we can see how the pressure at the first sandstone layers decreases rapidly. However, the marble formations act as impervious barriers and retard the consolidation process. For $0,80$ h we can see that almost all the sandstone layers have reached a pressure close to zero. We start to see some water infiltration through the fault located at the center of the domain. After 2.00 h have been passed the first marble layer decreased its pressure to 4 MPa and the second layer to 10 MPa. At the end of

the consolidation process the top sandstone layers are wholly consolidated, the pressure in the marble formation decreased considerably and the bottom sandstone is starting to get consolidated.

By visualizing the vertical deformation distributions we can see how the major displacements take place nearby the marble formations. This is to be expected since any fault within the domain will cause a stress concentration. Note also that after the water infiltrates through the fault ($t > 2.00\ h$), the magnitude of the vertical deformation starts to decrease once again since we are moving away from the stress concentration zones.

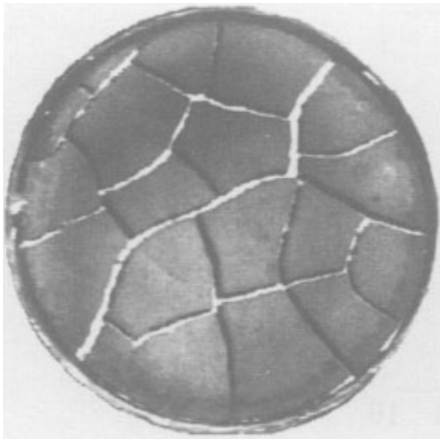
Chapter 6

Numerical application

In this numerical experiment, we simulate the drying process of an initially wet clayey soil in a Petri-dish. Our primary motivation to study desiccation processes in clayey soils is the initiation and consequent propagation of cracks. Desiccation cracking involves a gradual reduction in the saturation induced by evaporation from a material surface. The invasion of air accompanies the decrease in saturation into the pores. A reduction in the water pressure and an increment in the effective stress and soil shrinkage, also take place simultaneously. If the desiccation process takes place under constrained conditions (as in a Petri-dish), the increment of tensile stress can induce the formation of cracks when the stress reaches a critical value (Vo et al. 2017).

Perhaps, the most common geological example of desiccation cracking are the mudcracks. Naturally forming mudcracks start as wet, then muddy sediment dries up and contracts (Jackson & Bates 1997). Typically, a strain is developed when the top layer contracts and the soil at the bottom remain the same. If this strain is large enough, fractures start to appear at the surface of the soil to relieve the strain. Cracks will typically spread and join up, creating an interconnected network, see Fig. 6.1a.

(a)



(b)

Figure 6.1 a) Mudcracks in sewage sludge in Kos, Greece (https://upload.wikimedia.org/wikipedia/commons/2/2c/Desiccation-cracks_hg.jpg). Borrowed from Hannes Grobe under Creative Commons CC-BY-SA-2.5 license. **b)** Cracks induced by desiccation of a coffee-water mixture. Borrowed from *An Experimental Study of Cracking Induced by Desiccation by A. Groisman and E. Kaplan, p. 417.*

Extensive experimental studies have been carried out to understand the mechanisms of fracture initiation induced by desiccation in clayey soils (Albrecht & Benson 2001, Peron, Hueckel, Laloui & Hu 2009, Miller

et al. 1998, Tang et al. 2011, Weinberger 1999, Costa et al. 2013, Trabelsi et al. 2012). However, regardless the possible external factors that could influence the initiation of cracks (salinity, type of substrates, the thickness of substrates, etc.), the key criteria to determine a crack formation is the increment in the tensile stress within the system. In this context, we are particularly interested in the zones of tensile stress concentration that represent the areas where cracks are more likely to initiate. This numerical application was inspired by the work of Groisman & Kaplan (1994), where the authors performed several experiments on cracking of a coffee-water mixture in a Petri-Dish, see Fig. 6.1b.

To create a three-dimensional grid that resembles the actual geometry of a Petri-dish, we use a three-step procedure. First, we use the PDE-Toolbox from Matlab R2017b to generate a triangular grid in a circle (see Fig. 6.2 left) by applying a Dealuney triangulation (Matlab 2018). Then, the information from the triangulation (points and triangle locations) is passed to the `trianglegrid()` function from `MRST` to compute the geometry of the grid. Once the triangular mesh has been generated, we applied a Voronoi tessellation to convert the triangular mesh into a hexagonal mesh (see Fig. 6.2 right) by using the `pebi()` routine from MRST. The `pebi()` routine computes a dual of the triangular grid with edges that are perpendicular bisectors of the edges in the triangular grid. Finally, we extrude the hexagonal grid using the function `makeLayeredGrid()`, where a vector containing the thickness of the extruded layers must be passed. The final grid can be seen in Fig. 6.3.

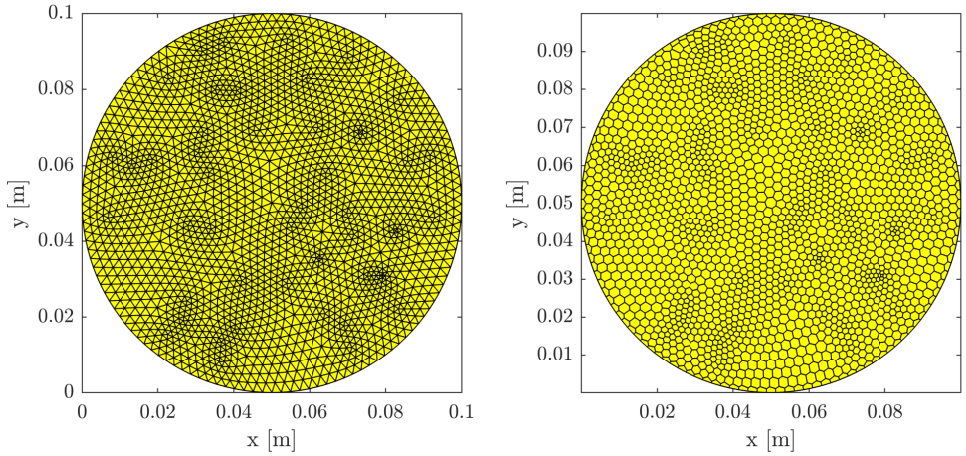


Figure 6.2 Triangular Delauney grid generated with PDE-Toolbox (left) and hexagonal Voronoi grid generated with MRST (right).

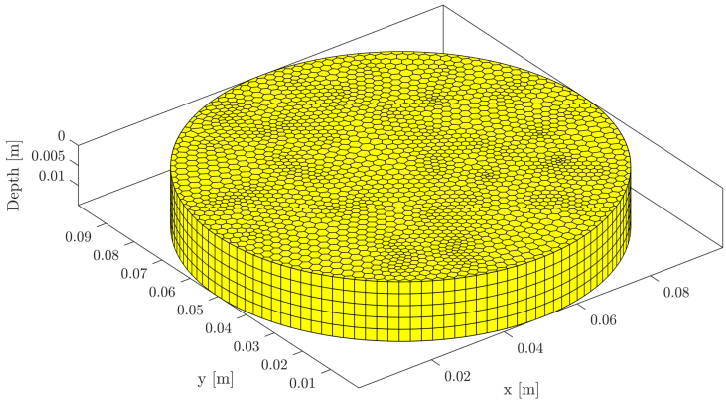


Figure 6.3 Extruded hexagonal three-dimensional grid resembling the actual geometry of a Petri-dish.

The physical properties of a typical clayey soil can be found in Table 6.1. The poro-elastic parameters have been taken from (Mondol et al. 2008) where a broad compendium of elastic properties for several

types of clays can be found. The van-Genuchten parameters have been taken from (Simunek et al. 2005).

Table 6.1 Physical properties.

Rock parameters	λ_s	1.0×10^{10}	Pa
	μ_s	2.5×10^{9}	Pa
	ρ_s	$1,760$	$kg \cdot m^{-3}$
	C_s	1.4×10^{-9}	Pa^{-1}
	n	0.39	$-$
	k	5.58×10^{-14}	m^2
	C_m	2.17×10^{-10}	Pa^{-1}
	α	0.9335	$-$
	S_ϵ	1.14×10^{-10}	Pa^{-1}
van Genuchten parameters	α_v	0.059	cm^{-1}
	θ_s	0.39	$-$
	θ_r	0.1	$-$
	m_v	0.3243	$-$
	n_v	1.48	$-$
Fluid parameters	ρ_w	$1,014$	$kg \cdot m^{-3}$
	μ_w	1	$Pa \cdot s$
	C_w	4.5455×10^{-10}	Pa^{-1}
	g	9.806	$m \cdot s^{-2}$

To simulate drying process, we assume instantaneous evaporation of the water at the top surface of the Petri-dish. We could have imposed a constant water evaporation rate, i.e. setting the top boundary flux as constant throughout the simulation. However, this approach will typically overestimate the amount of water that leaves the system. A better alternative is to adopt atmospheric boundary conditions with surface runoff.

External conditions control the potential evaporation rate across the upper boundary. Nevertheless, the actual flux depends also on the current saturation conditions. In other words, the top boundary condition may change from *flux controlled* to *pressure controlled*. Typically, the criteria to use one or another is obtained by limiting the water flux by the following conditions (Feddes et al. 1974)

$$\Gamma_{\text{top}} = \begin{cases} q & (q \leq E_{\max}) \;\&\; (\psi_{\text{top}} \geq \psi_{\text{crit}}) \\ E_{\max} & (q > E_{\max}) \;\&\; (\psi_{\text{top}} \geq \psi_{\text{crit}}) \\ \psi_{\text{crit}} & \text{otherwise} \end{cases}$$

where Γ_{top} is the adopted boundary condition, q is a prescribed water flux across the top boundary, $E_{\max}$ is the maximum (potential) evaporation rate, q_{top} is the water flux at the top boundary, ψ_{top} is the pressure head at the top boundary and ψ_{crit} is the critical pressure head which depends on the atmospheric conditions.

The most popular empirical models to determine $E_{\max}$ include the Thornthwaite equation (1948), the Penman equation (1948) and the Penman-Monteith equation (1965). These models typically require the knowledge of parameters such as the average daily temperature, the number of days in the month that are included in the calculation, the average day length and heat indices to approximate the potential

evaporation rate. We referred to Kumar et al. (1987) for an excellent introduction to these empirical equations. The application of these models is beyond the scope of this numerical example, and for simplicity, we are going to assume a given $E_{\max} = q_{top}$.

The critical pressure ψ_{crit} is the minimum allowed pressure head at the soil surface. As long as the pressure head at the top boundary is higher than ψ_{crit}, the top boundary condition will be flux controlled. The critical pressure can be obtained by the following equation (Van Dam & Feddes 2000)

$$R_h = \exp\left(\frac{\psi_{crit} M g}{RT}\right), \tag{6.1}$$

where R_h is the relative humidity, g is the gravitational acceleration, M is the molecular weight of water, R is the universal gas constant and T is the atmospheric absolute temperature.

Simunek et al. (2005) recommend selecting ψ_{crit} such that the corresponding water content is at least 0.005 higher than the residual water content. This is important, especially for coarse-textured soils which have a very steep retention curve where small changes in water contents could lead to significant changes in pressure heads, making the numerical solution unstable. If we define the *critical water content* as $\theta_{crit} := \theta_r + 0.005$, use the van Genuchten curve $\theta = f(\psi_w)$ and solve for ψ we can define a lower bound for ψ_{crit} which is

$$\psi_{crit} = -\frac{1}{\alpha_v}\left[\left(\frac{\theta_{crit} - \theta_r}{\theta_s - \theta_r}\right)^{-\frac{1}{m_v}} - 1\right]^{\frac{1}{n_v}} \tag{6.2}$$

Combining these two criteria, we should select ψ_{crit} based on the following conditions

$$\psi_{crit} = \begin{cases} \left(\frac{RT}{Mg}\right)\log R_h & \left(\frac{RT}{Mg}\right)\log R_h \geq -\frac{1}{\alpha_v}\left[\left(\frac{\theta_{crit}-\theta_r}{\theta_s-\theta_r}\right)^{-\frac{1}{m_v}} - 1\right]^{\frac{1}{n_v}} \\ -\frac{1}{\alpha_v}\left[\left(\frac{\theta_{crit}-\theta_r}{\theta_s-\theta_r}\right)^{-\frac{1}{m_v}} - 1\right]^{\frac{1}{n_v}} & \text{otherwise} \end{cases}$$

The boundary and initial conditions for this numerical example are shown in Table 6.2. The potential evaporation flux is set to $-6.4812 \times 10^{-9}\ m^3 \cdot s^{-1}$ which corresponds to a water velocity of $0.30\ cm \cdot h^{-1}$. If the system switches to pressure controlled, a pressure of $-7.9526 \times 10^6\ Pa$ which corresponds to a pressure head of $-80,000\ cm$ is imposed at the top boundary. We assume that at the top of the domain, the soil is completely traction free. The lateral surface as well as the bottom of the Petri-dish are assumed to be impermeable to the flux and constrained to deformations. The initial conditions correspond to a highly wet soil, with $p_w = -100\ Pa$ which corresponds to a saturation of 0.9963.

Table 6.2 Boundary and initial conditions.

Top[a]	$T_x = T_y = T_z$	0	Pa
	$\underline{q}_w \cdot \underline{n}$	-6.48×10^{-9}	$m^3 \cdot s^{-1}$
Top[b]	$T_x = T_y = T_z$	0	Pa
	p_w	-7.95×10^{6}	Pa
Lateral	$u_x = u_y = u_z$	0	m
	$\underline{q}_w \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Bottom	$u_x = u_y = u_z$	0	m
	$\underline{q}_w \cdot \underline{n}$	0	$m^3 \cdot s^{-1}$
Initial	$u_x = u_y = u_z$	0	m
	p_w	-1×10^{2}	Pa

[a] Flux controlled
[b] Pressure controlled

Table 6.3 Time and iteration parameters

Initial time	0	s
Final time	$10,000$	s
Initial time step	0.001	s
Minimum time step	0.001	s
Maximum time step	$1,000$	s
Max. number of iterations	10	–
Absolute tolerance	10^{-6}	–
Lower optimal iteration range	3	–
Upper optimal iteration range	7	–
Lower optimal iteration range	3	–
Lower time step mult. factor	1.3	–
Upper time step mult. factor	0.9	–

The time and iteration parameters can be seen in Table 6.3. According to our experience, when the system switches from flux to pressure controlled, there is often a jump in the number of iterations needed to reach the given tolerance. We found out that a practical way to overcome this issue is to *reset* the time step to the minimum time step. In other words, we give up computational speed in exchange for stability.

In Fig. 6.4 we show the pressure head (left column) and saturation (right column) distributions within the Petri-dish for five different times. For readability, we prefer not to show the coordinate axes. As expected, both the pressure head and the saturation decrease as a function of time. Note that we also included the percentage of the *possible* amount of evaporated water, i.e. not considering the water trapped as residual saturation. Initially, the Petri-dish starts with a saturation of 0.9963 and progressively decreases. However, the rate at which the system dries up depends not only on the boundary conditions but also on the water retention curves. Since the domain of interest is small, we do not see a defined saturation distribution but a complete reduction of the saturation at each time. Nevertheless, by inspecting the pressure profiles we can see that the zones of low pressure are located primarily around the top boundary. This should be expected, since the only path for the water to leave the system is the top boundary.

Perhaps, rather than analyzing the three-dimensional pressure and saturation distributions it would be more interesting to focus on the changes in pressure head and flux with respect to time at the top boundary (see Fig. 6.5). Since the top boundary is initially flux controlled, the water flux through this side of the domain remains constant and equal to the maximum evaporation rate $E_{\max}$. Meanwhile, the top pressure head decreases at a reasonable slow rate until we reach $4,000$ s, where we can see an abrupt decline in the pressure until reaching the critical pressure head ψ_{crit} at $4,133$ s . This is the exact point where the system changes from flux controlled to pressure controlled. From this point, we should expect the pressure at the top boundary to remain constant until the simulation ends. Note that when the transition between the two types of boundary conditions takes place, the water flux progressively decreases approaching zero. As a matter of fact, the system would reach an equilibrium point when there is no longer a gradient in hydraulic head in the $z-$ direction. It should be pointed out that regardless how long the final simulation time is, we will not be able to evaporate all the water, since the critical

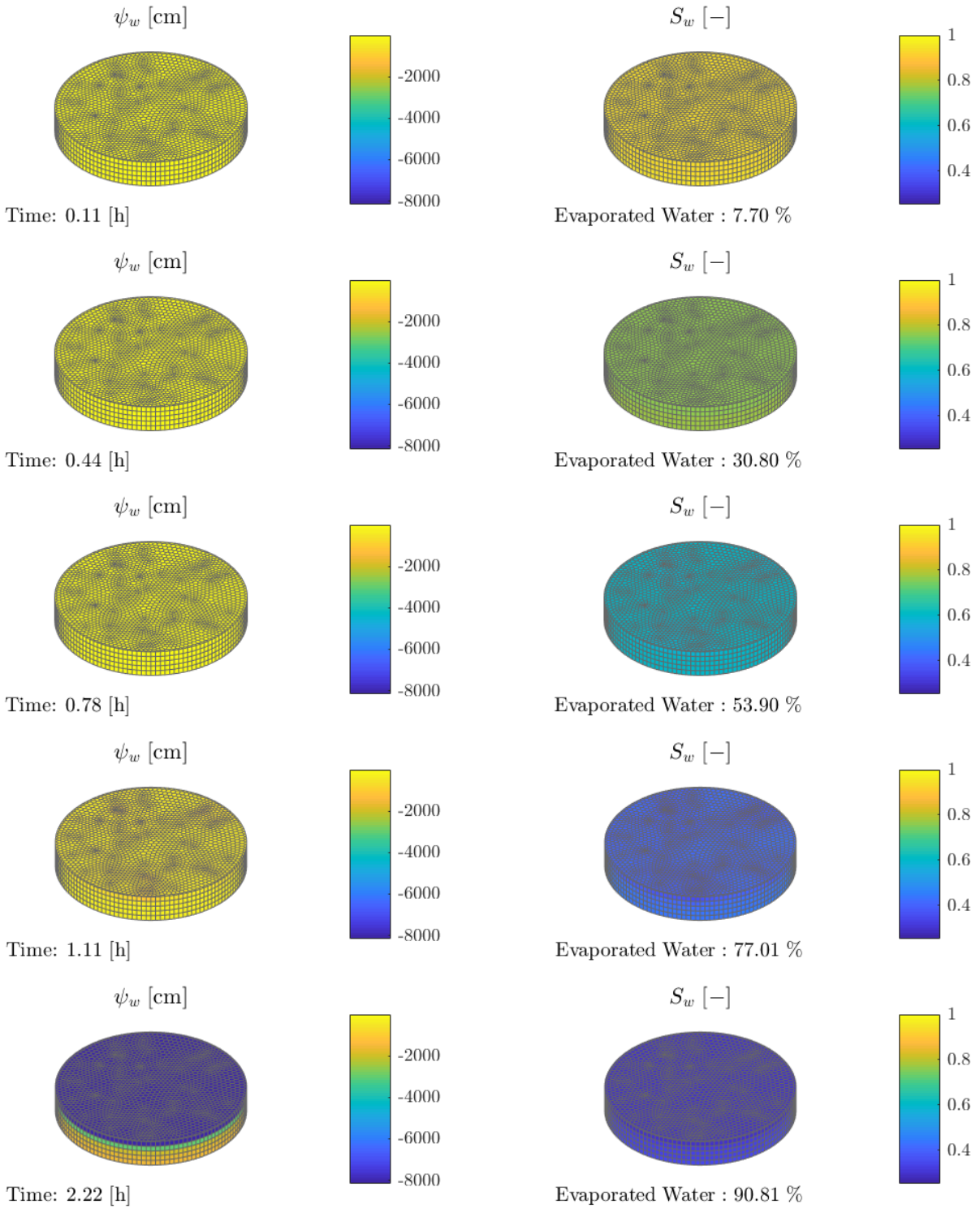


Figure 6.4 Pressure head and saturation for several times.

pressure head ψ_{crit} has been calculated from the critical water content θ_{crit} which, in turn, was set to be 0.005 higher than the residual water content θ_r.

We can now, focus our attention on the mechanical changes within the Petri-dish. In Fig. 6.6 we show the vertical and horizontal displacement fields for three different times. The contours that we show corresponds to the first layer located at a depth of 1.5 mm. We can note three marked characteristics for

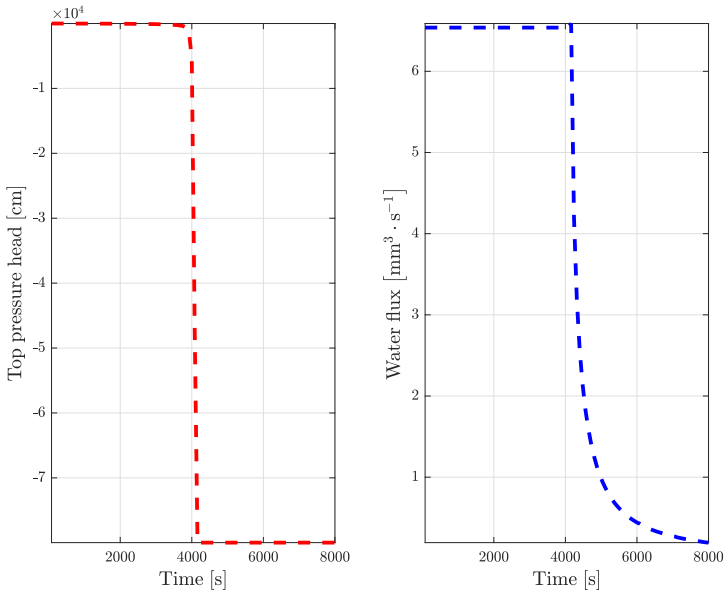


Figure 6.5 Top pressure head and water flux as a function of time.

both displacement fields. First, we can see a radial deformation pattern with the lowest displacements located at the center of the Petri-dish. Second, the magnitude of the displacement increases with time and is the same for u_x and u_y. This suggests that the symmetry is preserved in both directions as a function of time. Finally, the zones of higher displacements are located in a midway between the center and the lateral wall of the Petri-dish. Again, this should be expected since we have imposed zero displacement at the lateral boundaries of the domain.

We also show the displacement fields in the $z-$coordinate at each layer of the Petri-dish for the final simulation time (see Fig. 6.7). As we can see, the magnitude of the displacement decreases with depth. Two reasons explain this behavior. First, as the evaporation takes place nearby the top boundary, it is coherent to have the highest displacements around this zone. And second, since the bottom boundary is set as zero displacement, it is also consistent to see the smallest displacements nearby the bottom of the Petri-dish to satisfy the boundary conditions. Another essential point to remark is that in the first layer at $z = 1.5\ mm$ the deformations are more pronounced midway between the center and the lateral surface of the domain. This goes in agreement with the displacement fields in the x and $y-$directions.

The displacement fields shown in Figures 6.6 and 6.7 are consistent with the expected mechanical response of soils undergoing a desiccation process, where the reduction of water content (product of the evaporation) induces a shrinkage of the sample (Goehring et al. 2015).

Even though the principal effective stress remains in compression everywhere in the soil mass, the components of the full effective stress tensor may still have a tensile component (Shin & Santamarina 2011). Since the symmetric effective stress tensor have six components (3 normal, and 3 shears) at each face of the computational grid, the visualization process is a non-trivial task. Currently, methods like pseudocoloring, tensor glyphs and hyperstreamlines are used for this purpose (Boring 1998). However, if we are only interested in the tensile components of the effective stress, we can use the von Mises stress that provides a scalar value representing the equivalent tensile stress σ_v. In fact, the von Mises stress can be seen as the equivalent uni-axial tensile stress of a multi-axial state of stress (Mises 1913). In its more

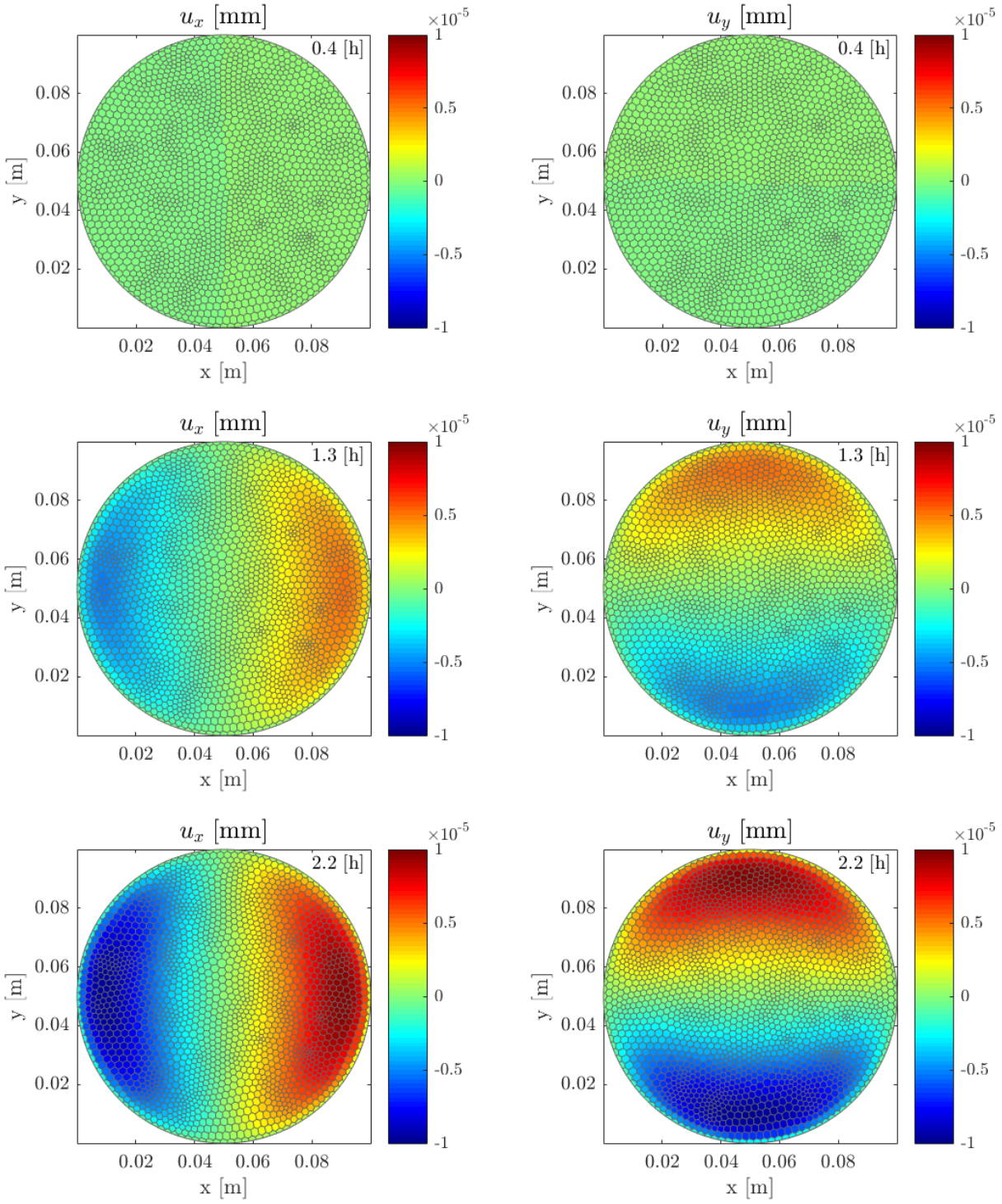


Figure 6.6 Horizontal and vertical displacement fields at Depth = 1.5 mm for three different times.

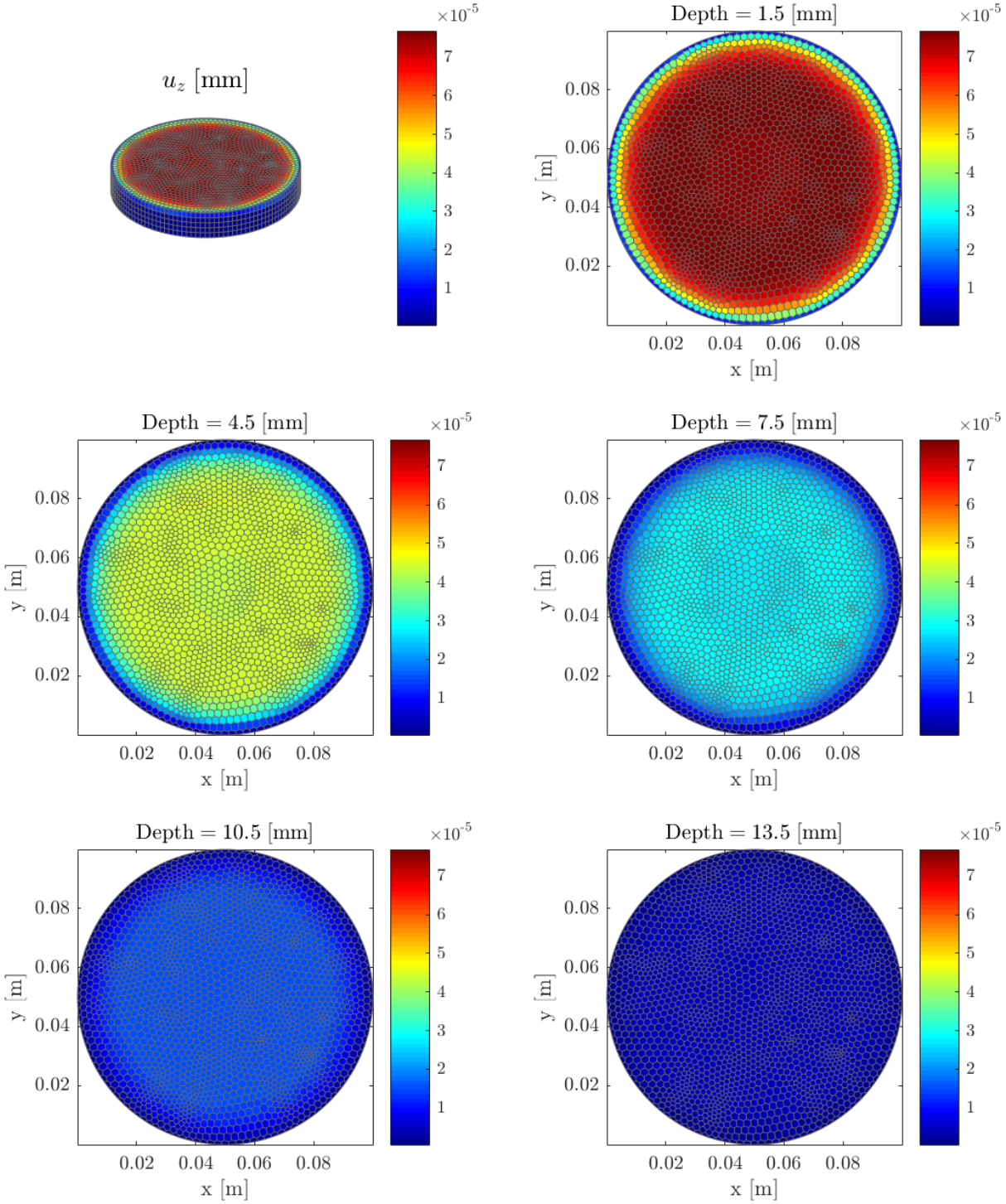


Figure 6.7 Displacement in the $z-$direction for the final simulation time at several depths of the Petri-dish.

general form, the von Mises stress is defined as

$$\sigma_v = \sqrt{\frac{1}{2}\left[\left(\sigma_{e_{xx}} - \sigma_{e_{yy}}\right)^2 + \left(\sigma_{e_{yy}} - \sigma_{e33}\right)^2 + \left(\sigma_{e_{zz}} - \sigma_{e_{xx}}\right)^2 + 6\left(\sigma_{e_{xy}}^2 + \sigma_{e_{yz}}^2 + \sigma_{e_{zx}}^2\right)\right]}. \tag{6.3}$$

For visualization purposes, let us introduce the normalized von Mises stress $\tilde{\sigma}_v$

$$[\tilde{\sigma}_v]_{\mathtt{f}} = \frac{[\sigma_v]_{\mathtt{f}} - \min\{[\sigma_v]_{\mathtt{f}}\}}{\max\{[\sigma_v]_{\mathtt{f}}\} - \min\{[\sigma_v]_{\mathtt{f}}\}} \qquad \forall \qquad \mathtt{f} \in (1, \mathtt{N_f}), \tag{6.4}$$

where $0 \leq [\tilde{\sigma}_v]_{\mathtt{f}} \leq 1$. Naturally, we are only interested in the highest values of the von Misses stress, and, of course, their location within the domain. Let us say, the set $\{[\tilde{\sigma}_v]_{\mathtt{f}} \,|\, [\tilde{\sigma}_v]_{\mathtt{f}} > 0.9\}$.

Until this point, our discussion was restricted to the spatial dependency of the tensile stress. However, the tensile stresses, and therefore the von Mises stresses, are time dependent. We will typically expect tensile stress to increment with time as soil shrinkage becomes more pronounced (Peron, Laloui, Hueckel & Hu 2009). This tendency is shown in Fig. 6.8 (left), where the maximum value of the von Mises stress within the domain as a function of time is plotted. Finally, we show the zones of concentrated normalized von Mises stress for the final simulation time in Fig. 6.8 (right), where the tensile stresses reached their maximum values. These zones correspond to the upper lateral walls of the Petri-dish. Thus, we can infer that cracks will initiate at the walls and propagate towards the interior of the domain. These results go in agreement with the experiments of (Groisman & Kaplan 1994). They have reported that primary cracks develop at the walls, mainly due to the high friction between the container and the porous material. The exact time of crack formation will primarily depend upon the tensile strength of the material under consideration. However, the exact location is quite difficult to determine. This will strongly depend upon the inhomogeneities of the material and possible nucleation points.

The natural extension of this numerical application will be to simulate the crack initiation. Nevertheless, this requires the consideration of several modeling, and implementational aspects that are beyond the scope of this thesis. These aspects are mentioned as further investigations in the next chapter.

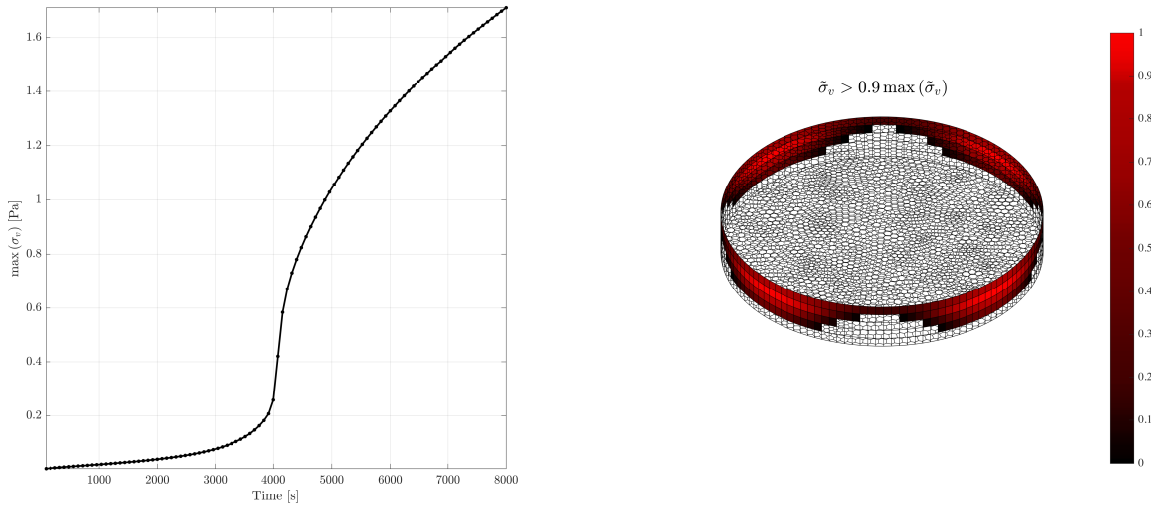


Figure 6.8 Left: Maximum von Mises stress as a function of time. Right: Zones of concentrated normalized von Mises stress located at the walls of the Petri-dish.

Chapter 7

Conclusion

In this thesis, we have presented a numerical simulator based on a cell-centered finite volume discretization capable of solving the unsaturated flow in deformable porous media. To our knowledge, this is the first simulator, at least in the framework of finite volumes capable of simulating these types of problems. Special cases have been thoroughly validated, i.e., unsaturated flow in non-deformable porous media (Richards' equation), linear elasticity (Navier-Lamé equations) and saturated flow in deformable porous media (Biot equations) with either analytical solutions or well-regarded commercial software.

Due to the non-linear constitutive relationships used to describe the water retention curves, the resulting discretized set of equations is also non-linear. To solve them, iterative-based solvers must be employed. In this context, the Newton method had shown remarkable results where virtually no effort was invested to calculate and compute the Jacobian matrix thanks to automatic differentiation.

As a numerical application, we have simulated the desiccation process of clayey soil in a Petri-dish. To emulate the reduction of the water content, we assumed instantaneous evaporation under atmospheric conditions. The displacements fields have shown the typical shrinkage that clayey soils undergo due to desiccation. A careful post-processing analysis of the resulting effective stress field allowed us to determine that the upper lateral walls of the Petri-dish are the potential zones of crack initiation.

The presented numerical tool represents a valuable resource that could be used to expand the current understating of the mechanisms of fracture initiation due to desiccation. Nevertheless, there are still several limitations to study fractured systems that we believe are worthy of further investigations. First, there are modeling questions, such as finding the correct way to describe the evaporation processes, not only at the top of the domain but within the fractures. Second, there are mesh-related or geometrical issues, like finding the best type of elements to represent the fracture geometry. Finally, a robust dynamical-adapted computationally-efficient algorithm to determine the angles and direction of fracture initiation and propagation according to some energy criteria such as the Griffith criteria, must be developed.

Appendices

List of Symbols

Acronyms

BE	Backward Euler
CCFVM	Cell Centered Finite Volume Method
FEM	Finite Element Method
FVM	Finite Volume Method
MPFA	Multi Point Flux Approximation
MPSA	Multi Point Stress Approximation
MRST	Matlab Reservoir Simulation Toolbox
REV	Representative Elementary Volume
TPFA	Two Point Flux Approximation
UFIDPM	Unsaturated Flow In Deformable Porous Media

Operators/Functions

$\mathbf{boundF}(\cdot)$	Discrete operator that computes boundary terms of the flow problem
$\mathbf{boundS}(\cdot)$	Discrete operator that computes boundary terms of the elasticity problem
$\mathbf{compat}(\cdot)$	Discrete operator that provides stability
$\mathbf{divF}(\cdot)$	Discrete operator that computes divergence of fluxes
$\mathbf{divS}(\cdot)$	Discrete operator that computes the divergence of the traction vector
$\mathbf{divU}(\cdot)$	Discrete operator that computes the divergence of the displacement
$\mathbf{F}(\cdot)$	Discrete operator that computes fluxes
$\mathbf{gradP}(\cdot)$	Discrete operator that computes the gradient of the pressure
$\mathbf{S}(\cdot)$	Discrete operator that computes traction vectors
$\mathbf{Tr}(\cdot)$	Trace operator
$\nabla \cdot (\bullet)$	Divergence
$\nabla (\bullet)$	Gradient
$\partial (\bullet)$	Partial derivative
$d (\bullet)$	Total derivative

Greek Symbols

α	Biot's coefficient	–
α_v	van Genuchten fitting parameter	–
λ	First Lamé parameter	$M \cdot L^{-1} \cdot T^{-2}$
λ_γ	Mobility of the phase γ	–
μ	Second Lamé parameter	$M \cdot L^{-1} \cdot T^{-2}$
μ_w	Water viscosity	$M \cdot L^{-1} \cdot T^{-1}$
ν	Poisson's ratio	–
Φ_w	Water potential	$L^2 \cdot T^{-2}$
ψ_w	Water pressure head	L
ψ_{crit}	Critical water pressure head	L
ρ_s	Density of the solid phase	$M \cdot L^{-3}$
ρ_w	Water density	$M \cdot L^{-3}$
σ_v	von Mises stress	$M \cdot L^{-1} \cdot T^{-2}$
θ_w^r	Residual water content	–
θ_w^s	Water content at saturated conditions	–
θ_w	Water content	–
θ_{crit}	Critical water content	–
$\tilde{\sigma}_v$	Normalized von Mises stress	$M \cdot L^{-1} \cdot T^{-2}$
$\underline{\underline{\gamma}}$	Engineering shear strain tensor	–
$\underline{\underline{\sigma}}$	Stress tensor	$M \cdot L^{-1} \cdot T^{-2}$
$\underline{\underline{\sigma}}_e$	Effective stress tensor	$M \cdot L^{-1} \cdot T^{-2}$
$\underline{\underline{\sigma}}_t$	Total stress tensor	$M \cdot L^{-1} \cdot T^{-2}$
$\underline{\underline{\varepsilon}}$	Strain tensor	–
ζ	Elevation head	L

Roman Symbols

$\dot{m}_w$	External rate of water mass per volume	$M \cdot L^{-3} \cdot T^{-1}$
$\mathfrak{C}^{-1}$	Compliance matrix	$M^{-1} \cdot L \cdot T^2$
$\mathfrak{C}$	Stiffness matrix	$M \cdot L^{-1} \cdot T^{-2}$
$\mathfrak{p}_w$	Dimensionless water pressure	–
$\mathfrak{z}$	Dimensionless vertical length	–

$\underline{\underline{k}}$	Intrinsic permeability	L^2
$\underline{\underline{K}}^w$	Water unsaturated hydraulic conductivity	$L \cdot T^{-1}$
$\underline{\underline{K}}^w_{sat}$	Water hydraulic conductivity	$L \cdot T^{-1}$
$\underline{g}$	Acceleration of gravity	$L \cdot T^{-2}$
$\underline{n}$	Normal unit vector	–
$\underline{q}_w$	Water Darcy's velocity	$L \cdot T^{-1}$
$\underline{T}$	Traction vector	$M \cdot L^{-1} \cdot T^{-2}$
$\underline{u}$	Displacement field	L
$\underline{v}_s$	Solid velocity	$L \cdot T^{-1}$
$\underline{v}_w$	Water velocity	$L \cdot T^{-1}$
$\underline{v}_{ws}$	Relative velocity of water w.r.t. solids	$L \cdot T^{-1}$
C	Specific moisture capacity	L^{-1}
c_c	Coefficient of consolidation	$L^2 \cdot L \cdot T^{-1}$
C_m	Compressibility of the porous medium	$M^{-1} \cdot L \cdot T^2$
C_s	Compressibility of solid grains	$M^{-1} \cdot L \cdot T^2$
C_w	Water compressibility	$M^{-1} \cdot L \cdot T^2$
D	Soil moisture diffusivity	$L^2 \cdot T^{-1}$
E	Young's modulus	$M \cdot L^{-1} \cdot T^{-2}$
$E_{\max}$	Maximum evaporation rate	$L^3 \cdot T^{-1}$
G	Shear modulus	$M \cdot L^{-1} \cdot T^{-2}$
g	Modulus of gravity acceleration	$L \cdot T^{-2}$
H^{-1}	Poroelastic expansion coefficient	$M^{-1} \cdot L \cdot T^2$
H_r	Relative humidity	–
h_w	Water hydraulic head	L
K	Bulk modulus	$M \cdot L^{-1} \cdot T^{-2}$
k_{rw}	Water relative permeability	–
M	Molecular weight of water	$M \cdot \mathrm{mol}^{-1}$
m_c	Confined compressibility	$M^{-1} \cdot L \cdot T^2$
m_v	van Genuchten fitting parameter	–
n	Porosity	–
n_v	van Genuchten fitting parameter	–

p_a	Air pressure	$M \cdot L^{-1} \cdot T^{-2}$
p_a	Pressure of the solid phase	$M \cdot L^{-1} \cdot T^{-2}$
p_w	Water pressure	$M \cdot L^{-1} \cdot T^{-2}$
Q_w	Water flux	$L^3 \cdot T^{-1}$
R	Universal gas constant	$M \cdot L^2 \cdot T^{-2} \cdot \mathrm{mol}^{-1} \cdot K^{-1}$
S_ε	Storativity	$M^{-1} \cdot L \cdot T^2$
S_a	Saturation of air	–
S_w	Saturation of water	–
T	Atmospheric absolute temperature	K
T_r	Transmissibility	L^3
V	Volume of the REV	L^3
V_a	Volume of air	L^3
V_f	Volume of fluids	L^3
V_s	Volume of solids	L^3
V_w	Volume of water	L^3

Bibliography

Aavatsmark, I. (2002), 'An introduction to multipoint flux approximations for quadrilateral grids', *Computational Geosciences* **6**(3-4), 405–432.

Aavatsmark, I. (2007), Multipoint flux approximation methods for quadrilateral grids, *in* '9th International forum on reservoir simulation', pp. 9–13.

Albrecht, B. A. & Benson, C. H. (2001), 'Effect of desiccation on compacted natural clays', *Journal of Geotechnical and Geoenvironmental Engineering* **127**(1), 67–75.

Bachmat, Y. & Bear, J. (1987), On the concept and size of a representative elementary volume (rev), *in* 'Advances in transport phenomena in porous media', Springer, pp. 3–20.

Barry, D., Parlange, J.-Y., Sander, G. & Sivaplan, M. (1993), 'A class of exact solutions for richards' equation', *Journal of Hydrology* **142**(1-4), 29–46.

Barry, D. & Sander, G. (1991), 'Exact solutions for water infiltration with an arbitrary surface flux or nonlinear solute adsorption', *Water resources research* **27**, 2667–2680.

Barry, D., Sander, G. & Phillips, I. (1991), Modelling solute transport, chemical adsorption and cation exchange, *in* 'Proceedings of the International Hydrology and Water Resources Symposium', Vol. 3, The Institution of Engineers, Australia, pp. 913–918.

Bear, J. (2013), *Dynamics of fluids in porous media*, Courier Corporation.

Biot, M. A. (1941), 'General theory of three-dimensional consolidation', *Journal of applied physics* **12**(2), 155–164.

Biot, M. & Willis, D. (1957), 'The elastic coeff cients of the theory of consolidation', *J. appl. Mech* **24**, 594–601.

Bird, R. B. (2002), 'Transport phenomena', *Applied Mechanics Reviews* **55**(1), R1–R4.

Boring, E. (1998), Visualization of tensor fields, Master's thesis, University of California, Santa Cruz.

Brezzi, F. & Fortin, M. (2012), *Mixed and hybrid finite element methods*, Vol. 15, Springer Science & Business Media.

Carslaw, H. & Jaeger, J. (1948), 'Operational methods in applied mathematics'.

Celia, M. A., Bouloutas, E. T. & Zarba, R. L. (1990), 'A general mass-conservative numerical solution for the unsaturated flow equation', *Water resources research* **26**(7), 1483–1496.

Chen, Z., Huan, G. & Ma, Y. (2006), *Computational methods for multiphase flows in porous media*, SIAM.

Churchill, R. V. (1972), 'Operational mathematics'.

Costa, S., Kodikara, J. & Shannon, B. (2013), 'Salient factors controlling desiccation cracking of clay in laboratory experiments', *Géotechnique* **63**(1), 18.

Cowin, S. & Mehrabadi, M. (1992), 'The structure of the linear anisotropic elastic symmetries', *Journal of the Mechanics and Physics of Solids* **40**(7), 1459–1471.

Darcy, H. (1856), *Les fontaines publiques de la ville de Dijon: exposition et application...*, Victor Dalmont.

Detournay, E. & Cheng, A. H.-D. (1995), Fundamentals of poroelasticity, *in* 'Analysis and design methods', Elsevier, pp. 113–171.

Dietrich, P., Helmig, R., Sauter, M., Hötzl, H., Köngeter, J. & Teutsch, G. (2005), *Flow and transport in fractured porous media*, Springer Science & Business Media.

Feddes, R. A., Bresler, E. & Neuman, S. (1974), 'Field test of a modified numerical model for water uptake by root systems', *Water Resources Research* **10**(6), 1199–1206.

Fredlund, D. G. & Rahardjo, H. (1993), *Soil mechanics for unsaturated soils*, John Wiley & Sons.

Freeze, R. A. & Cherry, J. A. (1979), 'Groundwater, 604 pp'.

FV-Biot (2018), https://github.com/keileg/fvbiot. Developed by E. Keilegavlen in the Department of Mathematics at the University of Bergen.

Giorgio, I., Andreaus, U., Scerrato, D. & Dell'Isola, F. (2016), 'A visco-poroelastic model of functional adaptation in bones reconstructed with bio-resorbable materials', *Biomechanics and modeling in mechanobiology* **15**(5), 1325–1343.

Goehring, L., Nakahara, A., Dutta, T., Tarafdar, S. & Kitsunezaki, S. (2015), *Desiccation cracks and their patterns: Formation and Modelling in Science and Nature*, John Wiley & Sons.

Groisman, A. & Kaplan, E. (1994), 'An experimental study of cracking induced by desiccation', *EPL (Europhysics Letters)* **25**(6), 415.

Hodneland, E., Hanson, E., Munthe-Kaas, A. Z., Lundervold, A. & Nordbotten, J. M. (2016), 'Physical models for simulation and reconstruction of human tissue deformation fields in dynamic mri', *IEEE Transactions on Biomedical Engineering* **63**(10), 2200–2210.

Hubbert, M. K. (1940), 'The theory of ground-water motion', *The Journal of Geology* **48**(8, Part 1), 785–944.

Jackson, J. & Bates, R. (1997), 'Glossary of geology, 769 pp', *Am. Geol. Inst., Alexandria, Va* .

Keilegavlen, E. & Aavatsmark, I. (2011), 'Monotonicity for mpfa methods on triangular grids', *Computational Geosciences* **15**(1), 3–16.

Keilegavlen, E. & Nordbotten, J. M. (2017), 'Finite volume methods for elasticity with weak symmetry', *International Journal for Numerical Methods in Engineering* .

Krogstad, S., Lie, K.-A., Møyner, O., Nilsen, H. M., Raynaud, X., Skaflestad, B. et al. (2015), Mrst-ad–an open-source framework for rapid prototyping and evaluation of reservoir simulation problems, *in* 'SPE reservoir simulation symposium', Society of Petroleum Engineers.

Kumar, K. K., Kumar, K. R. & Rakhecha, P. (1987), 'Comparison of penman and thornthwaite methods of estimating potential evapotranspiration for indian conditions', *Theoretical and applied climatology* **38**(3), 140–146.

Lewis, R. W. & Schrefler, B. A. (1998), *The finite element method in the static and dynamic deformation and consolidation of porous media*, John Wiley.

Lie, K.-A. (2014), 'An introduction to reservoir simulation using matlab: user guide for the matlab reservoir simulation toolbox (mrst). sintef ict'.

List, F. & Radu, F. A. (2016), 'A study on iterative methods for solving richards' equation', *Computational Geosciences* **20**(2), 341–353.

Lubliner, J. & Papadopoulos, P. (2016), *Introduction to solid mechanics*, Springer.

Matlab, P. (2018), 'Toolbox user's guide'.

MATLAB Reservoir Simulation Toolbox (2018), https://www.sintef.no/projectweb/mrst/. Developed by the Computational Geosciences group in the Department of Mathematics and Cybernetics at SINTEF Digital.

Mehrabadi, M. M. & Cowin, S. C. (1990), 'Eigentensors of linear anisotropic elastic materials', *The Quarterly Journal of Mechanics and Applied Mathematics* **43**(1), 15–41.

Merxhani, A. (2016), 'An introduction to linear poroelasticity', *arXiv preprint arXiv:1607.04274* .

Miller, C. J., Mi, H. & Yesiller, N. (1998), 'Experimental analysis of desiccation crack propagation in clay liners', *JAWRA Journal of the American Water Resources Association* **34**(3), 677–686.

Mises, R. v. (1913), 'Mechanik der festen körper im plastisch-deformablen zustand', *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse* **1913**(4), 582–592.

Mondol, N. H., Jahren, J., Bjørlykke, K. & Brevik, I. (2008), 'Elastic properties of clay minerals', *The Leading Edge* **27**(6), 758–770.

Nagtegaal, J. C., Parks, D. M. & Rice, J. (1974), 'On numerically accurate finite element solutions in the fully plastic range', *Computer methods in applied mechanics and engineering* **4**(2), 153–177.

Nordbotten, J. M. (2014), 'Cell-centered finite volume discretizations for deformable porous media', *International Journal for Numerical Methods in Engineering* **100**(6), 399–418.

Nordbotten, J. M. (2016), 'Stable cell-centered finite volume discretization for biot equations', *SIAM Journal on Numerical Analysis* **54**(2), 942–968.

Peron, H., Hueckel, T., Laloui, L. & Hu, L. (2009), 'Fundamentals of desiccation cracking of fine-grained soils: experimental characterisation and mechanisms identification', *Canadian Geotechnical Journal* **46**(10), 1177–1201.

Peron, H., Laloui, L., Hueckel, T. & Hu, L. B. (2009), 'Desiccation cracking of soils', *European journal of environmental and civil engineering* **13**(7-8), 869–888.

Pinder, G. F. & Celia, M. A. (2006), *Subsurface hydrology*, John Wiley & Sons.

Pinder, G. F. & Gray, W. G. (2008), *Essentials of multiphase flow in porous media*, John Wiley & Sons.

Richards, L. A. (1931), 'Capillary conduction of liquids through porous mediums', *Physics* **1**(5), 318–333.

Rogers, C., Stallybrass, M. & Clements, D. (1983), 'On two phase filtration under gravity and with boundary infiltration: Application of a bäcklund transformation', *Nonlinear Analysis: Theory, Methods & Applications* **7**(7), 785–799.

Ross, P. & Parlange, J.-Y. (1994), 'Comparing exact and numerical solutions of richards' equation for one-dimensional infiltration and drainage.', *Soil science* **157**(6), 341–344.

Sander, G., Parlange, J.-Y., Kühnel, V., Hogarth, W., Lockington, D. & O'kane, J. (1988), 'Exact nonlinear solution for constant flux infiltration', *Journal of Hydrology* **97**(3-4), 341–346.

Shin, H. & Santamarina, J. (2011), 'Desiccation cracks in saturated fine-grained soils: particle-level phenomena and effective-stress analysis', *Géotechnique* **61**(11), 961.

Simunek, J., Van Genuchten, M. T. & Sejna, M. (2005), 'The hydrus-1d software package for simulating the one-dimensional movement of water, heat, and multiple solutes in variably-saturated media', *University of California-Riverside Research Reports* **3**, 1–240.

Tang, C.-S., Shi, B., Liu, C., Suo, W.-B. & Gao, L. (2011), 'Experimental characterization of shrinkage and desiccation cracking in thin clay layer', *Applied Clay Science* **52**(1-2), 69–77.

Thomsen, L. (1986), 'Weak elastic anisotropy', *Geophysics* **51**(10), 1954–1966.

Timoshenko, S. & Goodier, J. (1951), 'Theory of elasticity.', *New York* **412**, 108.

Trabelsi, H., Jamei, M., Zenzri, H. & Olivella, S. (2012), 'Crack patterns in clayey soils: experiments and modeling', *International Journal for Numerical and Analytical methods in geomechanics* **36**(11), 1410–1433.

Van Dam, J. & Feddes, R. (2000), 'Numerical simulation of infiltration, evaporation and shallow groundwater levels with the richards equation', *Journal of Hydrology* **233**(1-4), 72–85.

Van Genuchten, M. T. (1980), 'A closed-form equation for predicting the hydraulic conductivity of unsaturated soils', *Soil science society of America journal* **44**(5), 892–898.

Verruijt, A. (2013), 'Theory and problems of poroelasticity', *Delft University of Technology* .

Verruijt, A. (2018), *An Introduction to Soil Mechanics*, Springer.

Vo, T. D., Pouya, A., Hemmati, S. & Tang, A. M. (2017), 'Numerical modelling of desiccation cracking of clayey soil using a cohesive fracture method', *Computers and Geotechnics* **85**, 15–27.

von Terzaghi, K. (1923), 'Die berechnung der durchassigkeitsziffer des tones aus dem verlauf der hydrodynamischen spannungs. erscheinungen', *Sitzungsber. Akad. Wiss. Math. Naturwiss. Kl. Abt. 2A* **132**, 105–124.

von Terzaghi, K. (1943), 'Theoretical soil mechanics'.

von Terzaghi, K. et al. (1925), 'Erdbaumechanik auf bodenphysikalischer grundlage'.

Šimůnek, J. & Bradford, S. A. (2008), 'Vadose zone modeling: Introduction and importance', *Vadose Zone Journal* **7**(2), 581–586.

Weinberger, R. (1999), 'Initiation and growth of cracks during desiccation of stratified muddy sediments', *Journal of Structural Geology* **21**(4), 379–386.