Doctoral theses at NTNU, 2019:19

Hilde Aas Nøst

Iteratively coupled implicit dynamic MPM-FVM

Doctoral thesis

Hilde Aas Nøst

ISBN 978-82-326-3648-8 (printed ver.)
ISBN 978-82-326-3649-5 (electronic ver.)
ISSN 1503-8181

NTNU
Norwegian University of
Science and Technology

NTNU

Norwegian University of Science and Technology
Thesis for the Degree of
Philosophiae Doctor
Faculty of Engineering
Department of Civil and Environmental
Engineering

Doctoral theses at NTNU, 2019:19

NTNU
Norwegian University of
Science and Technology

Hilde Aas Nøst

Iteratively coupled implicit dynamic MPM-FVM

Thesis for the Degree of Philosophiae Doctor

Trondheim, January 2019

Norwegian University of Science and Technology
Faculty of Engineering
Department of Civil and Environmental Engineering

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Engineering
Department of Civil and Environmental Engineering

ISBN 978-82-326-3648-8 (printed ver.)
ISBN 978-82-326-3649-5 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2019:19

Printed by NTNU Grafisk senter

PREFACE

This PhD study was done at the Geotechnical Division at the Department of Civil and Environmental Engineering, Faculty of Engineering at the Norwegian University of Science and Technology (NTNU). The work is presented as a monograph.

The study was financed by the Faculty of Engineering at NTNU and initiated by Prof. Thomas Benz in relation to a MPM initiative, MP3, an ITN application for EU-FP7.

The main supervisors have been Prof. Thomas Benz (2011-2013) and Prof. Gustav Grimstad (2013-2018), with Prof. Steinar Nordal as the co-supervisor.

The evaluation committee consisted of Lars Andresen from Norwegian Geotechnical Institute (NGI) as the first opponent, Prof. Phil J. Vardon from Technische Universiteit Delft as the second opponent, and Prof. Em. Lars Grande from NTNU as the third opponent and administrator of the committee.

Acknowledgements

I would like to gratefully acknowledge my main supervisor Professor Gustav Grimstad for his support and guidance during the course of my PhD. I appreciate your friendly reminder that a PhD is "just" an education when I doubted that I would ever finish. I appreciate our many discussions on the topic. I would also like to thank my co-supervisor Professor Steinar Nordal for trusting my ability to do a PhD, and for his support and experience. I am very grateful to my previous supervisor Professor Thomas Benz for introducing me to the topic of MPM during the final year of my Master studies. Thank you, Dr. Seyed Ali Ghoreishian Amiri, for fruitful discussions, good ideas, and for proofreading the thesis.

I wish to acknowledge the help provided by Marit Skjåk-Bræk for all kinds of administrative tasks.

My fellow PhD candidates of the geotechnical group were invaluable in their support. Thank you, Annika and Priscilla, for being role models for me when I started my PhD. Also thank you, Priscilla, for helping me find motivation when it started to dwindle. Thank you, Xenia, for making frustrations go away. Thank you, Emilie, for your enthusiasm and letting me join you on minor dog-sledding adventures. Takk, Linn, for alle gode kaffesamtaler. Thank you, all past and present PhD colleagues. I really appreciate the connection between us.

Takk til familie og venner som har holdt ut med at jeg "bare skal jobbe litt". Takk for at dere har vist interesse i arbeidet mitt, selv om jeg nok ikke alltid har lykkes i å formidle hva jeg *egentlig* holder på med.

Takk, kjære Sverre, for at du har holdt ut alle årene jeg har jobbet med doktorgraden. Takk for at du har holdt fortet hjemme. Jeg kunne ikke ha klart det uten deg.

Takk, kjære Ingeborg og Sigrid. Sammen med Sverre minner dere meg på hva som er det viktige i livet mitt.

Abstract

Several geotechnical engineering problems involve the interaction between deformation of the soil skeleton and the development of pore water pressures. Very often the deformations are considered to be small, which opens for a classical Lagrangian finite element analysis. However, large deformations are present in several geotechnical problems. Large deformations with methods such as updated Lagrangian finite element method might lead to severe mesh distortion. The material point method (MPM) is a method that avoids severe mesh distortion and can model large deformation problems with coupled flow. In literature, all approaches for coupling flow and mechanics with MPM use explicit time integration.

This thesis aims to develop a two-way coupled flow MPM with implicit time integration applicable to large deformation problems of flow through saturated porous media. Implicit time integration allows larger time steps and fewer computation steps for long-running processes. Minor aims are extending single-phase MPM software with the proposed method and an explicit approach from literature, and assessing the performance of the explicit and implicit approaches.

The explicit formulation uses the phase velocities as primary variables. The proposed implicit formulation uses the pore pressures and the displacements as primary variables. Additionally, the proposed formulation uses an iteratively coupled approach that splits the system into a mechanical part and a flow part which can be solved by separate solvers. This work couples MPM with the finite volume method (FVM) by taking advantage of the similarities between the finite element method (FEM) and MPM.

Both the explicit and implicit methods solves the one-dimensional consolidation problem. The quasi-static implicit approach has no oscillations at the beginning of the simulation. The dynamic implicit approach and the explicit approach experiences oscillations, with the explicit much larger than the implicit. Implicit time steps are 10^2 to 10^3 times larger than the explicit time steps. The implicit method simulates large deflection of a

poroelastic beam. The bending moment generates suction and pressure in the upper and lower parts of the beam, respectively. The beam deflects further while the pore pressures dissipate.

The proposed method can model both large deformations and coupled flow. The implicit time integration makes the method suitable for solving slow to medium rate coupled flow problems. Additionally, due to the modular nature of the proposed method, it is suited for extending single-phase implicit MPM codes to include coupled effects. Further work includes using advanced material models and extending the proposed method to model unsaturated soils.

Contents

CHAPTER 1

INTRODUCTION

Several geotechnical engineering problems are large deformation problems where the presence of water in the soil affects the soil shear strength. Landslides are often caused by a rise in pore pressures after for instance heavy rainfall, while earthquakes might lead to soil liquefaction and subsequent slides. In addition, water is the driving force in fluvial erosion problems, such as riverbank erosion and scour around underwater structures. The soil pore water content and pressures are affected by for instance the water level in the river [1]. In turn, the soil pore water pressure affects the soil strength. The erosion problems are large deformation problems because of the large mass transport and subsequent possible slope failure. Soil penetrating procedures such as the cone penetration test (CPT) and pile driving lead to immediate and local pore pressure generation around the penetrating shaft, which dissipates in rates from practically undrained to practically drained [2]. Flow through porous media under large deformations have applications in fields such as biomechanics [3] and food engineering [4].

Computational models may give an approximate idea of the ground behaviour during the above-mentioned processes and other large deformation problems that includes flow through porous media. Additionally, computational models may aid in interpreting CPT results, understanding the pile driving process, and predicting landslide runouts or effects of erosion and scour on structures. A proper computational model for these problems must handle both large deformations and coupled flow. MPM is a relatively

new numerical method which has been considered a promising tool in large deformation problems [5–7], including the large deformation problems with coupled flow in geotechnical engineering [8]. Therefore, MPM is chosen as the focus of this thesis.

Existing approaches developed for coupled MPM include velocity formulations for saturated [9] and unsaturated [10] soils, 2-point formulations [11–13] where the water and solid phases have each their set of particles, and coupling of MPM with the the finite difference method (FDM) method [14]. However, all of the previous approaches use explicit time integration. Explicit time integration is a good choice for high-velocity problems and in problems where all stress waves are needed. Similarly, implicit time integration is a good choice for slower rate problems [15, 16]. There exist implicit time integration approaches with single-phase MPM [16–19]. Although the quasi-static implicit MPM of Beuth [20] tracked the excess pore pressures, the method was undrained and therefore did not model the dissipation process. The consolidation process has not been done together with implicit time integration in MPM yet. Therefore, this thesis aims to provide an approach using implicit time integration for the modelling of coupled flow problems in geotechnical engineering that involves large deformations.

An approach that iteratively couples the displacements and the pore pressures has been in use in the field of reservoir engineering for some time [21–24]. Such coupling lets the displacements and pore pressures be solved by their own separate solver while keeping the other constant until global convergence is reached. The accuracy is the same as for the traditional fully coupled approach [22, 25, 26]. FEM for solving the displacements has been coupled with FVM for solving the flow [27, 28]. Coupling MPM with FVM is feasible too, because FEM and MPM both solve the linear momentum equation at the nodes. This thesis will adapt the iteratively coupled implicit approach for MPM and FVM.

1.1 Objectives and scope

The main objective is to further extend and adapt MPM for use in geotechnical engineering problems. The main research question is:

> How can implicit time integration be performed with coupled flow and large deformations in the material point method (MPM)?

Partial questions are:

- How can single-phase MPM codes be extended to include coupled effects?
- What are some of the benefits of implicit time integration on coupled effects in MPM?

Topics such as exploring several options for implicit time integration for coupled problems in MPM, constitutive models, advanced flux integration schemes, and advanced shape functions, are outside the scope of this thesis. However, the implicit method derived herein is applicable with more advanced options for the above-mentioned issues.

1.2 Thesis outline

The main body of this thesis can be split into three parts: background, main contribution, and validation.

The first part consists of Chapters 2 to 5. Chapter 2 presents a literature review on the topic of implicit time integration with coupled flow and displacements for MPM. The mathematical framework of single-phase flow through saturated porous media is presented in Chapter 3. Then, Chapter 4 presents the single-phase MPM. The velocity formulation for coupled MPM [9] is derived in Chapter 5, adapted for use with the Uintah computational framework [29].

The second part consists of the main contribution of this thesis, found in Chapters 6 and 7. An iteratively coupled implicit dynamic MPM-FVM is derived in Chapter 6. Chapter 7 describes the implementation into the Uintah computational framework [29].

The final part validates the methods presented and derived in the thesis. The velocity formulation in Chapter 5 and the iteratively coupled implicit MPM-FVM in Chapter 6 are compared and verified by the computation of the Terzaghi one-dimensional consolidation problem in Chapter 8. The iteratively coupled implicit MPM-FVM simulates the bending of a poroelastic beam. Results are compared and discussed.

The thesis ends with the conclusion in Chapter 9 and suggestions for further work in Chapter 10.

CHAPTER 2

LITERATURE REVIEW

This chapter presents the state-of-the-art of the subtopics of this thesis, after a general introduction to MPM. The main finding is the absence of implicit time integration with coupled MPM. The chapter concludes with proposing an implicit iteratively coupled approach adapted to MPM.

Chapter outline

2.1 General introduction to the material point method

FEM is often used to solve geotechnical problems such as bearing capacity and slope stability, in addition to coupled geomechanics problems like settlement calculations and transient groundwater flow[1]. The basis of FEM is a discretised grid or mesh of the problem, and this mesh deforms as the loading increases. Small deformations do not cause severe numerical problems due to the deformations. For large deformations, however, the mesh may deform in such a way that some elements become heavily distorted, thus leading to numerical inaccuracies and a potential breakdown of the calculation.

A solution to this issue could be remeshing at certain stages of the calculation, which is done in the coupled Eulerian-Lagrangian (CEL) and arbitrary Lagrangian-Eulerian (ALE) methods. These methods attempt to decouple the material displacements and the mesh displacements. Therefore the mesh may undergo less deformation than the material, and the problem of mesh distortion may be avoided. The uncoupling of material and mesh is done by generating a new mesh after the material deformation is computed if the material is distorted enough [31]. However, such remeshing might be time-consuming and computationally expensive, and numerical inaccuracies may be introduced when mapping properties from the old mesh to the new mesh.

Other types of numerical methods are developed for solving large deformations that avoids the mesh distortion issue by representing the material as particles instead of a mesh. One of the first meshless methods to be developed was the smoothed particle hydrodynamics (SPH) method [32], introduced in 1977. It was aimed at modelling astrophysical phenomena. Because the universe as we know it has no limits, the need of boundary conditions other than a free surface was not needed. Other methods similar to SPH include the moving least squares approximation, the diffuse element method, to name a few. In addition to the above-mentioned particle methods, there exist numerous other meshfree methods, such as the element-free Galerkin, the discrete element method and the particle-in-cell (PIC) method.

The PIC method is an older method presented by Evans & Harlow [33] that introduces a fixed background mesh, combining the benefits of a

[1] Consult a textbook on finite element analysis with geotechnical engineering, for instance Potts [30]

computational grid with the advantages of particle-based methods. PIC provided the basis for the fluid-implicit-particle method (FLIP) by Brackbill & Ruppel [34]. Later, Sulsky & Brackbill [35] extended FLIP for elastic bodies, but history-dependent materials could not be used with this method as the constitutive equations were applied to the grid nodes and not the particles. Another method based on FLIP were later introduced, but with the weak Galerkin formulation known from FEM allowing it to be applied in solid mechanics for history-dependent materials [36, 37]. This method is now better known as the material point method (MPM).

Like FEM, MPM is a numerical technique for finding approximate solutions to partial differential equations (PDE) as formulated in many engineering disciplines such as civil, chemical, mechanical, electrical, or aeronautical engineering. In classical FEM, the material domain is discretised as a Lagrangian mesh which deforms with the material. Unlike the classical FEM, MPM is not bound to a mesh or grid. The Eulerian reference mesh is fixed while the Lagrangian particles are free to move through the mesh. This combination of space discretisations makes MPM suitable for large deformation problems, as the distortion of the mesh is avoided while the material itself is allowed to deform. While MPM still suffers from the possible introduction of numerical inaccuracies in the mapping phase as in ALE, the problem of remeshing is solved by fixing the background grid[2], as opposed to ALE which has to compute the ideal remeshing. In a comparative study of meshless methods for use in hyper-velocity impact problems, Ma & Zhang [6] deemed the performance of MPM to be better than FEM and SPH.

In MPM the material points carry the history of loading, such as the deformation gradient, the volume and the effective stresses. The constitutive model is applied at each material point, which is analogous to the Gauss integration points of FEM. The conservation equations, however, are solved at the nodes of the background mesh, as with FEM. MPM and its predecessor PIC are considered meshfree particle methods [38], despite the presence of a background mesh.

In Figure 2.1 the stages of an FEM computation is illustrated; the blue dot in the middle being the Gauss point. In Figure 2.2 the corresponding stages of MPM is illustrated, with the smaller blue dots being the material points or particles. The first two stages of the two numerical methods are similar. The similarities between the particles and the Gauss point is seen when comparing Figures 2.1 and 2.2. However, the last stage is different.

[2]The mesh may also be redefined arbitrarily every time step if one wishes.

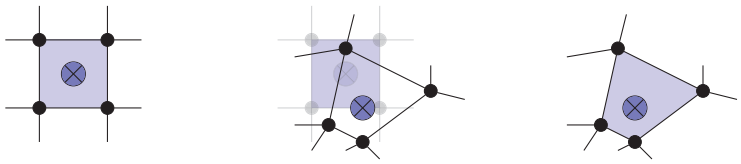


Figure 2.1: The phases of FEM from left to right: The initial configuration, the Lagrangian phase, the updated configuration.

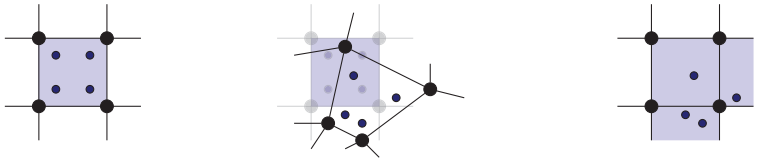


Figure 2.2: The three phases of MPM from left to right: The initial configuration, the Lagrangian phase, the convective phase.

Additionally, the Gauss point in FEM is fixed to the cell it was created in, but the particles in MPM are free to move from the original cell to new cells. In FEM, the deformed cell remains deformed for the rest of the computation. In MPM, the deformations are computed at the nodes of the mesh and then mapped onto the particles so they move. During the convective phase, the particles remain in their updated positions, but the mesh is reset. This way, mesh distortion is avoided even for large deformations.

MPM was later extended to include modelling of granular materials [39] and in modelling silo discharge of granular material [40]. Today, several fields are now developing MPM for their usage, examples include fire and explosion modelling [41], drifting of arctic sea ice [42], modelling food processes [43], biomechanics [44] and even modelling the fluid-membrane interaction in in-vitro fertilization of mice [45]. MPM has also been used in the field of computer graphics for modelling of granular media, such as sand [46] and snow [47]. The latter method [47] was used to simulate snow behaviour in the Disney animated movie Frozen.

MPM is also being developed within the field of geotechnical engineering and soil mechanics. Some of the recent advances include improcing frictional contact [48, 49], modelling CPT with various degree of drainage [2, 9], and modelling of internal erosion [50].

2.2 Implicit time integration with the material point method

Explicit time integration is considered the best approach for high frequency or velocity problems where the time step has to be small enough to catch the response [51]. In those cases, the chosen time step is smaller than the critical time step anyway. For almost any other type of problem where the elastic wave response is not needed, implicit time integration is preferred despite the computational cost per time step. Explicit time integration is limited by the critical time step defined by the Courant-Friedrichs-Lewy (CFL) condition in order to converge, and is therefore *conditionally stable*.

Implicit time integration, however, is not limited by the CFL condition. Implicit time integration is theoretically *unconditionally stable*, which guarantees convergence, although accuracy may suffer when the time step is too large. The accuracy of the solution may be affected by too large time steps, but in most cases, there will be convergence to a solution. Allowing for larger time steps, a simulation with implicit time integration uses fewer time steps in total than the explicit counterpart. However, the computational cost of one time step with implicit time integration is larger than for explicit time integration. In the case of explicit time integration, no extra iterations are needed as the equations are computed off known variables. In the case of implicit time integration, a system of equations on the form of $\mathbf{Ax} = \mathbf{b}$ must be solved for the unknown $\mathbf{x}$. If the system is linear, the coefficient matrix $\mathbf{A}$ must either be inverted or the system solved by other methods. Solving the system becomes even more expensive if it is nonlinear, with coefficient matrix $\mathbf{A}$ and solution vector $\mathbf{b}$ depending on $\mathbf{x}$. In general, explicit time integration is faster per time step, but need more time steps than implicit time integration, which is more expensive per time step, but requires fewer time steps overall.

In the context of MPM, explicit time integration has an added disadvantage. When particles move through the mesh, there is a jump in the acceleration of the particle [37, 52, 53] and unbalance in the nodal internal forces [18] caused by the individual history variables of each material point [54]. Steffen *et al.* [53] showed analytically that the spatial errors dominate when the time steps are small, as with explicit time integration. Guilkey & Weiss [18] reported that explicit time integration for quasi-static problems increased the numerical artefact related to grid crossing, while implicit time integration did not.

Implicit time integration in MPM can be grouped into two approaches:

with and without explicitly forming the system tangent stiffness matrix. Starting with Guilkey & Weiss [18] and Guilkey & Weiss [55], the tangent stiffness matrix approach has been adopted in both dynamic [16, 56] and quasi-static [57, 58] formulations.

Cummins & Brackbill [59] proposed a matrix-free Newton-Krylov approach with a restriction on the time step based on the strain to locate the yield stress. Therefore the gain in time step size was small compared to explicit time integration, although the accuracy of the implicit solution was better than the explicit [59]. Several other matrix-free formulations have since been developed [17, 19, 60–62].

Using a larger strain restriction than the restriction given by [59], lead to larger applicable time steps [60]. Sulsky & Kaul [19] succeeded in using time steps that were in the order of 10^4 times the explicit time step. There is however an upper limit to time step size for implicit MPM to avoid inverting the MPM background grid during one time step [18, 19, 59, 60].

In addition, there are several formulations for implicit MPM formulations with higher-order shape functions, like the generalised interpolation material point (GIMP) [58, 60] and dual domain material point (DDMP) [62] methods. GIMP has been combined with both matrix-free [60] and tangent stiffness [58] approaches.

2.3 Coupled geomechanics in the material point method

Coupled geomechanics in MPM shares several traits with coupled geomechanics in FEM. One of these is which primary variables to use: $\mathbf{u}$-p formulation or $\mathbf{v}$-$\mathbf{w}$ formulation. There exist several approaches for solving the two-phase problem of geomechanics [63]. The $\mathbf{u}$-p formulation assumes that the relative acceleration of fluid to solid phase is much smaller than the acceleration of the solid phase itself. It can be simplified further, for instance by assuming that the fluid acceleration is negligible or that all accelerations are negligible (quasi-static assumption). The $\mathbf{u}$-p formulation has one kinematic variable, either displacement $\mathbf{u}$ or velocity $\mathbf{v}$ of the solid, in addition to the pore pressure p. This leads to an easily solvable parabolic problem which can be extended to partially saturated soils by adding another pressure parameter denoting the pore air pressure. The $\mathbf{u}$-p formulation can be used to model for instance earthquakes and other slower phenomena as consolidation, but cannot properly model high-frequency

phenomena [64].

The **v**-**w** formulation formulation,[3] on the other hand, can model high-frequency problems. Here, **v** denotes the solid velocity and **w** is the water velocity. In velocity formulations, the acceleration of the two phases are not equal and both are present. The **u**-p formulation cannot model mixing and separation of phases because the water velocity is relative to the solid velocity [12]. As can be seen in Table 2.1, most recent advances in the field of coupled MPM has been with the **v**-**w** formulation.

2.3.1 Spatial discretisation

There are two main approaches of spatial discretisation to model coupled flow with the MPM: One is using one set of material points to model all phases, where each phase take up a relative portion of the material point volume and mass [9, 10, 65, 68, 69]. The other main approach is to use two sets of material points, one for each phase considering saturated soils [11–13, 67, 70]. The two-point MPM proposed by Bandara & Soga [13] was derived for saturated soils, but performed well for partially saturated soils too.

In their review, Soga *et al.* [8] pointed out that two-point formulations do conserve both the mass of water and the mass of the solid skeleton, while the one point formulations could not guarantee the conservation of mass of water. Another advantage of the two-point formulation is that the water particles both can model the pore water within the soil skeleton and free water. That way, geotechnical problems such as seepage problems through a dam embankment can be modelled naturally. See for instance Abe *et al.* [12] and Bandara & Soga [13] for simulations of levee failure using two-point MPM formulations.

Coupling MPM with other numerical methods. In addition to pure MPM, approaches to coupled flow, either with one or two sets of material points, MPM have been coupled with other numerical methods. Higo *et al.* [14] and Higo *et al.* [66] modelled the solid phase with MPM and the pore pressures were evaluated at the background mesh cell centres by FDM. Later, the same authors extended their method to work with GIMP [72]. They applied the MPM-FDM method on seepage problems and for unsaturated soils but did not mention how to apply boundary conditions in the case of large deformations.

[3] Equivalent to the u-U formulation proposed by Zienkiewicz & Shiomi [64].

Table 2.1: Summary of the different coupled MPM formulations that exist in literature sorted by year ascending. All of the references in the table use explicit time integration.

Reference	Year	No. of layers	Formulation	Other
Zhang *et al.* [11]	2008	2	—	
Zhang *et al.* [65]	2009	1	**u**-p	
Higo *et al.* [14] and Higo *et al.* [66]	2010	1	**u**-p	MPM-FDM
Mackenzie-Helnwein *et al.* [67]	2010	2	**v**-**w**	
Zabala & Alonso [68]	2011	1	**u**-p	
Jassim *et al.* [9]	2013	1	**v**-**w**	
Zheng *et al.* [69]	2013	1	**u**-p	CPDI & GIMP
Abe *et al.* [12]	2014	2	**u**-p	
Bandara & Soga [13]	2015	2	**v**-**w**	
Yerro *et al.* [10]	2015	1	**v**-**w**	3-phase
Liu *et al.* [70]	2017	2	**v**-**w**	GIMP
Yerro *et al.* [50]	2017	1	**v**-**w**	Internal erosion
Yang *et al.* [71]	2017	1	**v**-**w**	MPM-DEM

Another coupling approach is to couple MPM with the discrete element method (DEM), proposed by Yang *et al.* [71]. The fluid motion and the interphase forces were modelled with MPM, and the solid particle and solid boundary interaction as DEM. The main drawback of DEM is the discretisation size. It would be too expensive to model all soil grains discretely. Consequently, the results in [71] may not be practically useful for problems with a large number of grains. However, it did show promising results for modelling solid particles interspersed in water, as is the case with debris flow.

Alternative MPM formulations. The standard MPM interpolation functions are linear, which is the cause of several numerical problems [73]. GIMP [54] was suggested to minimize cell-crossing errors by smoothing out the particle mass, similar to SPH. In addition, other techniques are developed, such as the shape tracking method of convected particle domain interpolation (CPDI) [74] and higher-order interpolation functions such as B-splines [75, 76].

These alternative interpolation techniques have also been applied to coupled MPM formulations. Abe *et al.* [12], Zheng *et al.* [69], and Liu *et al.* [70] applied GIMP to coupled MPM formulations to reduce cell crossing errors. However, Abe *et al.* [12] pointed out that with fine meshes, GIMP results tended to original MPM results.

Integration of pore pressures. Pore pressures are integrated either at the cell centre or at particle location. The two-point formulations are derived for evaluating the pore pressure at the water particle locations [12, 13, 70]. Although, in order to stabilise the problem, pore pressures are averaged over the cell and integrated at cell centre [12, 13, 68], analogous to the kinematic locking mitigation procedures in [20, 77, 78]. This is equivalent of reduced order pressure interpolation technique, as the Q_1-P_0 element in FEM [12, 13]. The coupled MPM-FDM also integrates pore pressures at cell centres by default [14, 66, 72]. Using the cell average of the pore pressure and evaluating it in the cell centre corresponds to pore pressures being piecewise constant values.

The one-point formulations also evaluate pore pressures at the particle locations. Jassim *et al.* [9], Yerro *et al.* [10], Zhang *et al.* [65], and Liu *et al.* [70] (**v**-**w** formulation) computed pore pressures at particle location using nodal velocities with gradients computed at the particle location. Zabala & Alonso [68] and Zheng *et al.* [69] (**u**-p formulation) computed pore pressures

at nodes and interpolated to particles, using gradients evaluated at the cell centre.

2.3.2 Time integration.

FEM and MPM are also similar with regards to time integration methods. As stated in Section 2.2, explicit time integration must have time steps within some critical time step bounds. The upper bound is defined by the CFL condition. However, for coupled geomechanics, Vermeer & Verruijt [79] defined a lower bound. Zabala & Alonso [68] argued that the CFL condition for saturated soils gives unacceptable small time steps for simulating construction and consolidation of saturated materials. The elastic modulus of the material is used when computing the wave speed. Since the elastic modulus for an undrained saturated material is large, the accepted time step will then be small. Zabala & Alonso [68] surpassed the problem by scaling the material densities. This approach was also used by Jassim *et al.* [9].

All coupled MPM approaches discussed here use explicit time integration. Kularathna & Soga [80] did use a semi-implicit approach for their MPM formulation for incompressible flow, but this thesis is concerned with compressible flow.

2.4 Coupling strategies

As discussed in Section 2.2, explicit time integration schemes are conditionally stable. Additionally, for coupled geomechanical systems the range of applicable time step sizes is narrow, see Section 2.3.2. If the time step is outside of the valid range, the computation will either become too inaccurate or fail to converge. Implicit time integration, on the other hand, is unconditionally stable but leads to computational expensive time steps. In addition, a system of equations describing coupled phenomena has a larger number of unknowns than the corresponding single-phase system, which increases the complexity and computational cost of the problem.

The derivation and implementation of a fully coupled implicit method is thus an additional challenge. The system is nonlinear as the displacements affect the pore pressures and the pore pressures affect the displacements.

There exist several approaches to solving problems of flow through porous media with implicit time integration. In a comparison of coupling methods, Dean *et al.* [27] mentions three approaches commonly used for

solving problems of flow in porous media. Those are the *fully coupled*, *iteratively coupled* and the *explicit coupled* methods, where the two latter can be grouped into *partially coupled* approaches.

2.4.1 Fully coupled approach.

Existing implicit FEM geotechnical codes such as Plaxis use a fully coupled approach when solving consolidation problems. In this case, both the hydraulic and the geomechanical problems are solved at once. This scheme is unconditionally stable.

However, in order to use a fully coupled approach, the fully coupled equations must be implemented. Due to the coupling terms, this may be challenging in already existing codes. Furthermore, the resulting system of equations may be time-consuming and expensive to solve, as the coefficient matrix is dense.

2.4.2 Partially coupled approaches.

This is a family of approaches of which both the explicit and the iteratively coupled methods belong. Since the hydraulic and geomechanical equations are solved separately, each of the subproblems can, therefore, be solved with separate simulators suitable to the task [81]. Because of this coupling, extending existing software is made simpler.

Explicit approach. The explicit or *staggered* approach solve the subproblems only once per time step [82] and is conditionally stable, but additional stabilization measures can be implemented [83].

The partial coupling also allows for using different time steps for the two subproblems [23]. This *loose coupling* approach use smaller time steps for the flow problem, and updates the mechanical problem after a number of flow steps have been solved. The magnitude of pore volume change decides the number of mechanical updates: small changes in pore volume give large geomechanical time steps. Consequently, this approach is economic in use of computational time, especially because of the cost of solving for displacements with implicit time integration. However, since this type of coupling is explicit, this scheme is only conditionally stable [27].

Iteratively coupled approach. Prevost [84] suggested solving transient coupled field problems by partitioning the fields and iterating. Then the intermediate solutions from each subproblem are fed into the other

subproblem repeatedly until convergence. Splitting the displacements and the pore pressures allow for a combination of solvers to be used [22]. This concept is applied to couple FVM and FEM simulators [27, 28, 85–88], with FVM solving the flow equations and FEM the mechanical eqations of the problem of flow in porous media.

The *iteratively coupled* approach gives a converged solution that is identical to the solution obtained by the fully coupled procedure. The approach may be unconditionally stable depending on how the partial coupling is performed [24, 26]. Kim [24] describes, verifies and tests four different iteratively coupled schemes based on the physical problem: drained split, undrained split, fixed-strain split, and fixed-stress split.

The drained and undrained splits solve the mechanical system first. The difference between the two splits is that the drained split freezes the pore pressure field ($\delta p = 0$) and the undrained split freezes the change in fluid mass ($\delta m = 0$) while solving the mechanical system. Then the flow system is solved using the resulting displacement field, which is now kept constant.

The fixed-strain and fixed-stress split solve the flow system first. The fixed-strain split freezes the rate of volumetric strain ($\dot{\varepsilon}_{\mathrm{v}} = 0$), whereas the fixed-stress split freezes the rate of volumetric stress ($\dot{\sigma}_{\mathrm{v}} = 0$). Then the mechanical system is solved using the pore pressure obtained in the flow system and with no change in pore pressure ($\dot{p} = 0$).

Of these, the undrained and the fixed-stress splits were shown to be unconditionally stable, with the fixed-stress split having a minor advantage over the undrained split when dealing with nearly incompressible fluids [24, 26, 87, 89]. The undrained bulk modulus K_{u} is used when solving the mechanical problem using the undrained split, which results in a stiff system that needs a robust linear solver [24]. White *et al.* [90] showed that the fixed-stress operator split is equal to applying an upper triangular block preconditioner to the fully implicit block system.

However, the unconditional stability of the undrained and fixed-stress splits is only shown for the quasi-static case [26].

2.5 Literature review conclusion

As mentioned in Chapter 1, several geotechnical applications are types of problems suitable for implicit time integration. Additionally, implicit time integration is unconditionally stable and allows for larger time steps, where explicit time integration is conditionally stable and limited by the

critical time step. MPM cause several errors, of which the spatial errors are prominent. The spatial errors are reduced when using larger time steps.

There are two main approaches to implicit time integration with MPM. One is matrix-free, the other explicitly forms the tangential stiffness matrix, but the resulting system of equations may also be solved by a matrix-free approach.

The development of coupled geomechanics with MPM has mainly used explicit time integration. MPM has been coupled with FDM previously.

Iterative coupling for solving coupled geomechanics problems with the fixed-stress split is unconditionally stable in the quasi-static case. The displacements and the pore pressures can be solved by different solvers, which makes it easier to extend existing codes. In MPM, the displacements are solved at the nodes, which is similar to FEM. Consequently, the application of this technique to MPM is viable.

The software used in this thesis uses the tangential stiffness matrix approach for implicit time integration. Therefore that approach is adopted here for ease of implementation. Furthermore, the fixed-stress split is applied when iteratively coupling the single-phase implicit MPM with a newly implemented FVM.

As far as the laws of mathematics refer to reality,
they are not certain;
and as far as they are certain,
they do not refer to reality.

Albert Einstein

CHAPTER 3

THEORETICAL BACKGROUND AND GOVERNING EQUATIONS

This chapter presents the framework and governing equations for the modelling of coupled problems in saturated porous media. First, topics in continuum mechanics are presented. Then the mathematical framework for modelling saturated porous media is derived. Two formulations are considered, a displacement-pressure formulation and a velocity formulation. Their corresponding strong governing equations and accompanying boundary conditions are derived.

Chapter outline

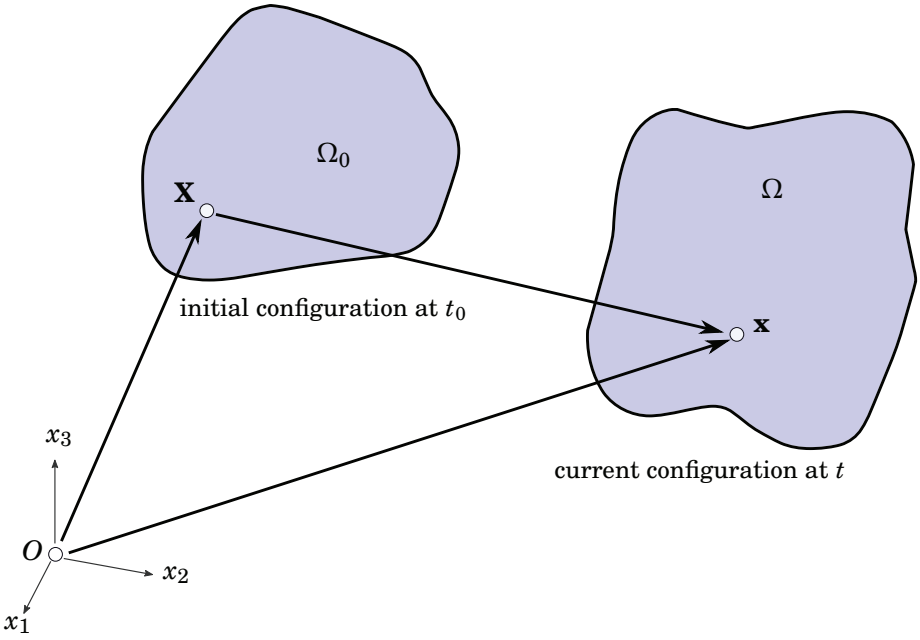


Figure 3.1: Configuration of a body Ω_0 at time t_0 with a material point of coordinates $\mathbf{X}$, and the same body, now deformed Ω, at time t where the material point has changed position to $\mathbf{x}$.

3.1 Continuum mechanics preliminaries

The mathematical framework is based on continuum mechanics principles. This section will introduce the relevant topics that are used when developing the framework.

We need a *frame of reference* to track deformations, translations, and rotations of a body Ω. There are two main types of such references: *Eulerian* and *Lagrangian* frames of reference. Both describe the configuration and are mathematically and physically equivalent, but the point of view is different.

As an example, let us consider a moving car. The velocity of the car varies along the road as a function of space and time. At the same time, consider the police measuring car velocities by radar. They stand at the fixed position $\mathbf{x}$. They will also measure various velocities of the cars passing by as a function of space and time, the velocities are associated only with this particular spatial point. Now, our car comes along and passes the

position $\mathbf{x}$ at time t. The velocity measured by the police (Eulerian) is the same as the velocity measured by the moving car (Lagrangian),[1] despite having different frames of reference.

Consider a function that depends on spatial and time coordinates. In a Eulerian description, a function ϕ that describes some property, depends on the *current* coordinates $(\mathbf{x}, t)$. In a Lagrangian description, another function Φ that describes the same property as the Eulerian ϕ, depends on the *initial* coordinates denoted by $(\mathbf{X}, t)$, as seen in Figure 3.1. This initial reference configuration can either be the configuration at the start of the simulation, $t_0 = 0$, which is called *total* Lagrangian. An alternative would be to use the previous configuration as the initial reference, $t_0 = t_{n-1}$ if t_n denotes the current time step. This approach is called *updated Lagrangian*.

The location of a material point $\mathbf{X}$ within the domain Ω_0 is given by the Lagrangian function

$$\mathbf{x} = \mathbf{x}(\mathbf{X}, t), \tag{3.1}$$

which have a unique inverse Eulerian function

$$\mathbf{X} = \mathbf{X}(\mathbf{x}, t) \tag{3.2}$$

that describes the initial position $\mathbf{X}$ of the material point that happen to be located at $\mathbf{x}$ at time t. Differentiating the Lagrangian function $\Phi(\mathbf{X}, t)$ with respect to time gives

$$\frac{\partial \Phi}{\partial t}(\mathbf{X}, t), \tag{3.3}$$

since $\mathbf{X}$ is independent of time t. The same differential of the Eulerian function $\phi(\mathbf{x}, t)$ with respect to time, however, becomes more complex. The total derivative is

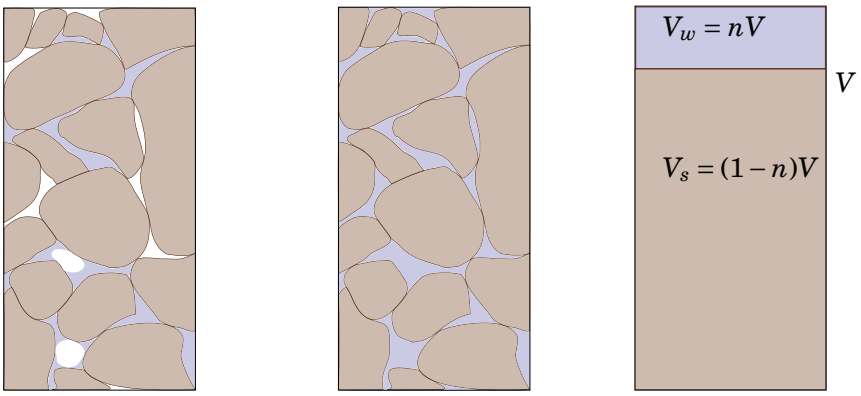
$$\frac{\mathrm{d}\phi}{\mathrm{d}t}(\mathbf{x}, t) = \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial t} \tag{3.4}$$

with the chain rule applied. We note that $\partial \mathbf{x}/\partial t = \mathbf{v}$ and rewrite to

$$\frac{\mathrm{d}\phi}{\mathrm{d}t}(\mathbf{x}, t) = \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi, \tag{3.5}$$

where ∇ operates on functions defined in the current configuration. This derivative is often called the *material time derivative* in Eulerian coordinates. We see that the two time differentials in Equations (3.3) and (3.5)

[1] Assuming a perfect world with perfect measurements.



(a) Unsaturated soil **(b)** Saturated soil **(c)** Saturated soil as a continuum

Figure 3.2: Representations of an elemental soil volume consisting of solid grains (brown) and voids filled with water (blue) and air (white).

are different, with the Eulerian being more complex due to the *convective* term $\mathbf{v} \cdot \nabla \phi$. The convective term accounts for the relative motion between the material and the grid.

The choice of reference configuration depends on the problem to solve. For instance, in solid mechanics, the material state and history of loading are usually needed, and thus, a Lagrangian description where each material point is traced is preferable. Additionally, while fluids experience large deformations under anisotropic loading, solids may not. A Lagrangian description may lead to computational breakdown due to the numerical trouble resulting from severe material distortion. Inversion of material elements, for example, may result in a negative determinant of the deformation gradient, the *Jacobian*, $J < 0$. A negative Jacobian J is physically inadmissible since it means that the updated volume of the material element is negative. A Eulerian description is easier when modelling large deformations because mesh distortion is avoided, however, the convective term is computationally expensive and can introduce numerical errors due to its non-symmetry [91].

For small strains the linearised strain can be used, where higher order strain terms are cancelled. For large strains and deformations, there exist several alternative strain measures to choose from. All of them can be found from the deformation gradient $\mathbf{F}$, which is given as the differential of

the current position with respect to the reference position:

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}. \tag{3.6}$$

3.2 Mixture theory

Soil is a porous material that consists of three phases: a solid, a liquid and a gas phase, see Figure 3.2. Usually, the liquid and gas phases consist of water and air, respectively. This feature makes soil a heterogeneous material where its components exhibit different behaviours. For instance, the soil grains are able to resist shear forces, which water and air are assumed to be unable to. Additionally, air has negligible weight. Therefore the ratios between the three phases contribute to the soil behaviour.

The soil skeleton consists of soil grains and voids filled with fluid. The soil is *unsaturated* if the fluid is a mix of water and air (Figure 3.2(a)). The soil is *saturated* if the voids are completely filled with liquid water (Figure 3.2(b)). Some porous materials may even have several different liquids and gases in their voids, as encountered in the field of reservoir engineering. The soil skeleton is deformable, while both the soil grains and the water are assumed to be almost or completely incompressible.

Pore water pressure *generation* or build-up occurs when the soil skeleton is compressed during loading and the pore pressures increase because the voids decrease. When the pore pressures increase, the intergranular stresses between the soil grains are lowered leading to a decrease of the soil shear strength. Pore water pressure *dissipation* or *consolidation* is a transient state in which the water travels through the pore network. The water flows from points of higher potential (energy) or excess pore pressure to points of lower excess pore pressure until steady-state is reached. With time the water is squeezed out of the soil voids, which in turns leads to lower pore pressures, smaller voids, and increased intergranular stresses. The dissipation process heavily influences geotechnical engineering problems such as settlement calculations.

Saturated soils can be modelled as a single-phase material when subject to either *drained* or *undrained* loading conditions. Drained loading conditions assume that there is no pore pressure build-up at all and therefore full dissipation. Coarse materials or long-term effects are often assumed to be drained. In the first case, the water quickly dissipates by flowing almost freely through the soil skeleton, while in the latter case almost all pore pressure dissipation is finished. Undrained loading conditions occur when

water cannot escape from the soil volume. There is pore pressure build-up, but no dissipation and hence no volume change. Fine-grained clays tend to have undrained behaviour due to the tiny pores and grains, and undrained conditions may also be considered for short-term loadings. The two loading conditions have in common that there is no pore pressure flow relative to the soil skeleton. In the drained case, there are no pore pressures at all. In the undrained case, the water moves with the soil. Therefore, drained and undrained loading conditions may model the soil as a single-phase material.

For partially drained loading conditions, there is relative movement between the water and the soil grains. Such conditions are often two-way coupled problems where displacements and pore pressures influence each other, and consequently, they are complex to model. To derive the basic set of equations governing the generation and consequent dissipation of pore pressures, several simplifying assumptions are made.

First, the soil skeleton can deform, changing the bulk volume. Usually, the water is modelled as slightly compressible with a very high bulk modulus, although water may also be considered incompressible. The soil grains are usually modelled as being incompressible, which means that there is no volume change of the solid part of the bulk volume. With water being almost incompressible and soil grains being incompressible, the main change in bulk volume would have to be due to a change in the size of pores. Consequently, the bulk volume deforms during consolidation because of the escape of water from the pores and slightly because of the small compressibility of water.

To summarise, the following assumptions are made in this thesis:

Assumption 1. *The soil is saturated.*

Assumption 2. *No mass exchange between phases.*

Assumption 3. *The soil particles are incompressible:* $K_s = \infty$

Assumption 4. *There is no spatial variation of phase densities*

Assumption 5. *Isotropic permeability*

Assumption 6. *The relative acceleration of fluid to solid phase is much smaller than the acceleration of the solid phase so that* $a_w \approx a_s$

Properties are given subscripts s and w for solid and water phase respectively, while properties without subscript describe the saturated mixture.

3.2.1 Volume fractions and density

Mixture theory assumes that every point in a mixture is occupied simultaneously by all its components [92]. Considering a saturated soil element as depicted Figure 3.2(b) with total volume V, the total volume of all soil grains is V_s and for water V_w. Because the soil is saturated, we can express the total volume as

$$V = V_s + V_w, \tag{3.7}$$

which can be used to form a continuum approximation of the soil as is done in Figure 3.2(c). There are two common soil properties to denote the proportion of voids to solids:

$$e = \frac{V_w}{V_s} \qquad \text{and} \qquad n = \frac{V_w}{V}, \tag{3.8}$$

where e is called the *void ratio* and n[2] the *porosity*. They can be derived from one another from

$$e = \frac{n}{1-n} \qquad \text{and} \qquad n = \frac{e}{1+e}. \tag{3.9}$$

The volume proportion of the solids to the total volume is $n_s = (1-n)$ for saturated soils since the sum of porosities add up to 1.

Having defined the porosity n, the total density of the mixture of soil grains and water is expressed by

$$\rho = n_s \rho_s + n \rho_w \tag{3.10}$$

$$= (1-n)\rho_s + n\rho_w. \tag{3.11}$$

The total density is derived from the assumption that the total mass of the mixture is the sum of the mass of water and the mass of solids: $m = m_s + m_w$. The averaged densitites are sometimes denoted by superscript, so that $\rho^s = (1-n)\rho_s$ and $\rho^w = n\rho_w$. In general terms, the average phase density ρ^α, where α is the phase, is written as

$$\rho^\alpha = n_\alpha \rho_\alpha. \tag{3.12}$$

3.2.2 The principle of effective stress

With the same total stress, the larger the pore pressures are, the less strength the total soil skeleton has. Larger pore pressures mean that the

[2] In porous media physics it is common to denote porosity by φ, but this symbol usually denotes the friction angle in geotechnical engineering.

pore water carries larger portions of the load, reducing contact between soil grains. In turn, the shear strength of the soil changes with the pore pressures since the shear strength is influenced by the contact between soil grains, like friction and locking. This effect leads to the derivation of *effective stress*. The foundation of the principle of effective stress was introduced by Terzaghi in 1923 [translated in 93][3].

For saturated soils, the total stress can be decomposed into the stress carried by the solid grains $\boldsymbol{\sigma}_s$ and the stress carried by the water $\boldsymbol{\sigma}_w$, so that

$$\boldsymbol{\sigma} = (1-n)\boldsymbol{\sigma}_s + n\boldsymbol{\sigma}_w, \tag{3.13}$$

where $\boldsymbol{\sigma}_w = p\mathbf{I}$, p is the pore pressure, and $\mathbf{I}$ the second order identity tensor. However, the effective stress concept is used instead, where the total stress is decomposed into intergranular stresses or *effective stresses* and pore water pressure p:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}' + p\mathbf{I}, \tag{3.14}$$

where $\boldsymbol{\sigma}$ is the total stress tensor and $\boldsymbol{\sigma}'$ is the effective stress tensor. Then we can derive that the portion of the stress carried by the solid grains is

$$(1-n)\boldsymbol{\sigma}_s = \boldsymbol{\sigma}' + (1-n)p\mathbf{I}. \tag{3.15}$$

The advantage of using the effective stress approach is that a soil constitutive model can govern the effective stresses and Darcy's law the pore pressures. Borja & Alarcón [95] remarked that the effective stress approach is valid for large deformations too.

In this thesis, both the stresses and the pore pressure are considered negative in compression.

3.2.3 Stiffness moduli

The drained bulk modulus of the soil skeleton is denoted K_{dr}. For 3D problems $K_{dr} = K$. For 1D problems

$$K_{dr}^{1D} = E_{oed} = 3\frac{1-\nu}{1+\nu}K, \tag{3.16}$$

where ν is the Poisson ratio and E_{oed} is the *oedometer modulus*[4] which reflects the stiffness when the soil is allowed to deform in only one direction.

[3] Several authors have since developed the principle of effective stress. See de Boer & Ehlers [94] for a historical review.

The *Biot modulus* M is the stiffness modulus for the pores. The pore fluid compressibility is the inverse of the Biot modulus and is the sum of the compressibility of the phases, so we have that

$$\frac{1}{M} = \frac{n}{K_{\mathrm{w}}} + \frac{1-n}{K_{\mathrm{s}}}, \tag{3.17}$$

and with the assumption that the soil grain stiffness $K_{\mathrm{s}} \approx \infty$, this gives

$$\frac{1}{M} = \frac{n}{K_{\mathrm{w}}}, \tag{3.18}$$

where $K_{\mathrm{w}} \sim 2.2\,\mathrm{GPa}$ is the bulk modulus of water.

3.3 Governing equations

Two physical properties need to be conserved to couple soil displacements and water pressure: the conservation of mass and the conservation of linear momentum. Three momentum balances can be posed: one for each phase plus one for the mixture. Only two of those are needed for the computational framework, though, as two of the three balances can be linearly combined into the third.

The time differentials of displacements $\mathbf{u}$ are denoted as

$$\mathbf{v}(\mathbf{x}, t) = \dot{\mathbf{u}} = \frac{\partial \mathbf{u}}{\partial t} \tag{3.19}$$

$$\mathbf{a}(\mathbf{x}, t) = \ddot{\mathbf{u}} = \frac{\partial^2 \mathbf{u}}{\partial t^2}. \tag{3.20}$$

3.3.1 Conservation of mass

In the case of no mass entering or leaving the domain Ω, the mass of the continuum should remain constant. Mathematically, the total mass balance is written as

$$\int_{\Omega} \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) \right] \mathrm{d}\Omega = 0, \tag{3.21}$$

and consequently,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = \frac{\partial \rho}{\partial t} + \mathbf{v} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{v} = 0, \tag{3.22}$$

[4] Also known as the *constrained modulus*.

which is the conservation of mass in Eulerian reference frame. We can obtain the conservation of mass in the Lagrangian reference frame by reintroducing the material time derivative from Equation (3.5), which gives

$$\frac{\mathrm{d}\rho}{\mathrm{d}t} + \rho\nabla\cdot(\mathbf{v}) = 0. \tag{3.23}$$

Similarly, assuming no mass exchange between the phases of a mixture continuum (Assumption 2), we derive the conservation of mass for each phase. First, we have the conservation of mass of the solid phase in the Lagrangian frame:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left[(1-n)\rho_\mathrm{s}\right] + (1-n)\rho_\mathrm{s}\nabla\cdot\mathbf{v} = 0, \tag{3.24}$$

and using Assumption 3 we end up with

$$-\frac{\mathrm{d}n}{\mathrm{d}t} + (1-n)\nabla\cdot\mathbf{v} = 0. \tag{3.25}$$

For the conservation of mass of the water phase, we get

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(n\rho_\mathrm{w}\right) + n\rho_\mathrm{w}\nabla\cdot\mathbf{w} = 0, \tag{3.26}$$

where $\mathbf{w}$ is the phase velocity of water.

The storage equation for u-p formulation. The conservation of mass for the saturated soil as one continuum can be derived from the conservation of the phase masses in Equations (3.25) and (3.26). In order to combine the two phases, we need to write the conservation of mass of the water phase with respect to the solid skeleton. The superscripts s and w will denote which phase the material time derivative is done with respect to.

If there is a function f in water phase reference frame, the material time derivative with respect to the solid skeleton will be

$$\frac{\mathrm{d}^\mathrm{w}f}{\mathrm{d}t} = \frac{\mathrm{d}^\mathrm{s}f}{\mathrm{d}t} + (\nabla f)\cdot\tilde{\mathbf{w}}, \tag{3.27}$$

where $\tilde{\mathbf{w}} = \mathbf{w} - \mathbf{v}$ is the relative velocity between the phases: the *interstitial velocity*. It represents the velocity of the fluid through the pores relative to the skeleton while taking into account the portion of the body that is water, denoted by porosity n.

Rewriting Equation (3.26) with respect to the solid skeleton then gives:

$$\frac{\mathrm{d}^{\mathrm{s}} n\rho_{\mathrm{w}}}{\mathrm{d}t} + \left(\boldsymbol{\nabla} n\rho_{\mathrm{w}}\right)\cdot\tilde{\mathbf{w}} + n\rho_{\mathrm{w}}\boldsymbol{\nabla}\cdot\mathbf{v} + \tilde{\mathbf{w}} = 0 \tag{3.28}$$

$$n\frac{\mathrm{d}^{\mathrm{s}}\rho_{\mathrm{w}}}{\mathrm{d}t} + \rho_{\mathrm{w}}\frac{\mathrm{d}^{\mathrm{s}} n}{\mathrm{d}t} + \boldsymbol{\nabla}\cdot\left(\rho_{\mathrm{w}} n\tilde{\mathbf{w}}\right) + n\rho_{\mathrm{w}}\boldsymbol{\nabla}\cdot\mathbf{v} = 0. \tag{3.29}$$

Let us introduce the specific discharge[5] $\mathbf{q}$ defined as the velocity of the pore water relative to the pores averaged over the entire cross-sectional area and found from

$$\mathbf{q} = n(\mathbf{w}-\mathbf{v}) = n\tilde{\mathbf{w}}, \tag{3.30}$$

where n is the porosity, and $\mathbf{w}$ and $\mathbf{v}$ the pore water and solid grains velocities, respectively.

To find the time derivative of ρ_{w}, we apply the chain rule

$$\frac{\mathrm{d}\rho_{\mathrm{w}}}{\mathrm{d}t} = \frac{\partial\rho_{\mathrm{w}}}{\partial p}\frac{\mathrm{d}p}{\mathrm{d}t}, \tag{3.31}$$

and introduce the linearly compressible water constitutive model

$$\frac{\partial\rho_{\mathrm{w}}}{\partial p} = -\frac{\rho_{\mathrm{w}}}{K_{\mathrm{w}}}. \tag{3.32}$$

Inserting Equation (3.31) with Equation (3.32) and Equations (3.25) and (3.30) into Equation (3.29) gives

$$\rho_{\mathrm{w}}\frac{\mathrm{d}p}{\mathrm{d}t} = \frac{K_{\mathrm{w}}}{n}\left[\rho_{\mathrm{w}}\boldsymbol{\nabla}\cdot\mathbf{v} + \boldsymbol{\nabla}\cdot\left(\rho_{\mathrm{w}}\mathbf{q}\right)\right], \tag{3.33}$$

and with Assumption 4, we can cancel ρ_{w} to obtain

$$\frac{\mathrm{d}p}{\mathrm{d}t} = \frac{K_{\mathrm{w}}}{n}\left[\boldsymbol{\nabla}\cdot\mathbf{v} + \boldsymbol{\nabla}\cdot\mathbf{q}\right]. \tag{3.34}$$

The storage equation for v-w formulation. For writing the storage equation in Lagrangian coordinates, we obtain first the conservation of mass for the solid and water phases, respectively:

$$\frac{\partial n}{\partial t} = \boldsymbol{\nabla}\cdot[(1-n)\mathbf{v}] \tag{3.35}$$

$$\rho_{\mathrm{w}}\frac{\partial n}{\partial t} + n\frac{\partial\rho_{\mathrm{w}}}{\partial t} + \boldsymbol{\nabla}\cdot\left(n\rho_{\mathrm{w}}\mathbf{w}\right) = 0. \tag{3.36}$$

Using Assumption (4) with Equations (3.31) and (3.32) then gives:

$$\frac{\partial p}{\partial t} = \frac{K_{\mathrm{w}}}{n}[\boldsymbol{\nabla}\cdot[(1-n)\mathbf{v}] + n\boldsymbol{\nabla}\cdot\mathbf{w}]. \tag{3.37}$$

[5] Also called *superficial water velocity* or *Darcy velocity*.

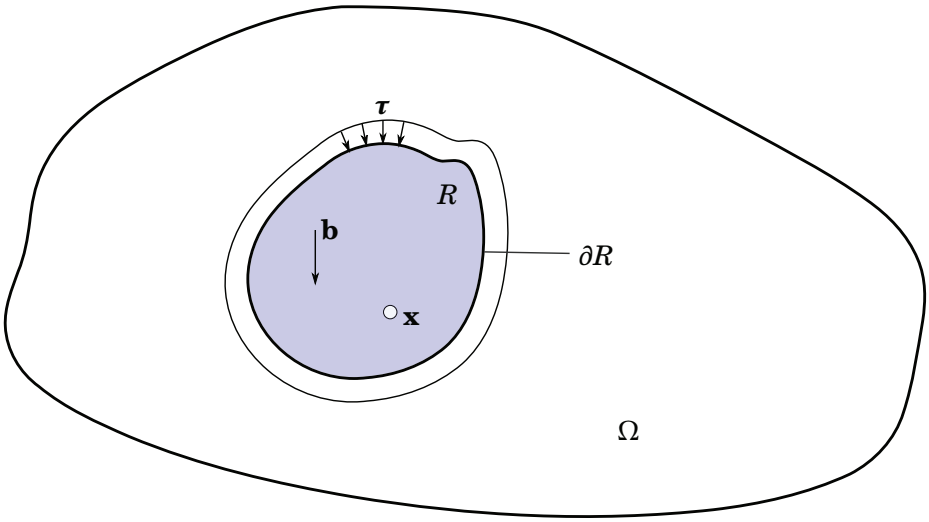


Figure 3.3: An arbitrary subregion R with boundary ∂R within the domain Ω. The subregion R has density ρ and is subject to boundary traction force $\boldsymbol{\tau}$, body force $\mathbf{b}$, and is moving with velocity $\mathbf{v} = \mathrm{d}\mathbf{x}/\mathrm{d}t$.

3.3.2 Conservation of momentum

The conservation of momentum is two-fold: conservation of *angular* momentum and *linear* momentum. The conservation of angular momentum is satisfied if the stress tensor is symmetric, that is, when

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\mathsf{T}}, \tag{3.38}$$

with $(\cdot)^{\mathsf{T}}$ denoting the transpose of the tensor. The conservation of angular momentum automatically holds if the stress tensor used is symmetric. The Cauchy stress tensor and the first Piola stress tensor are symmetric. The contribution of pore pressure $p\mathbf{I}$ to the total stress $\boldsymbol{\sigma}$ is also symmetric.

The conservation of linear momentum stems from Newton's second law of motion: $\sum \mathbf{F} = m\mathbf{a}$. We start out with the linear momentum L of some subregion R of our domain Ω, as shown in Figure 3.3, written as

$$L = \int_R \rho \mathbf{v} \, \mathrm{d}R, \tag{3.39}$$

where ρ is the material density and $\mathbf{v}$ is the material velocity. The time derivative of L should be equal to the forces acting on the subregion R, which would be the traction $\boldsymbol{\tau}$ on the boundary ∂R and the body force per

unit mass $\mathbf{b}$ acting in the subregion R. Written out, this gives

$$\frac{\mathrm{d}L}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\int_R \rho\mathbf{v}\,\mathrm{d}R = \int_{\partial R} \boldsymbol{\tau}\,\mathrm{d}S + \int_R \rho\mathbf{b}\,\mathrm{d}R. \tag{3.40}$$

With surface traction defined as $\boldsymbol{\tau} = \boldsymbol{\sigma}\cdot\mathbf{n}$ and applying the divergence theorem, the first term on the right hand side of Equation (3.40) can be written as

$$\int_{\partial R} \boldsymbol{\tau}\,\mathrm{d}S = \int_{\partial R} \boldsymbol{\sigma}\mathbf{n}\,\mathrm{d}S = \int_R \nabla\cdot\boldsymbol{\sigma}\,\mathrm{d}R. \tag{3.41}$$

After applying the Reynold's transport theorem and inserting Equation (3.41) to Equation (3.40), we get

$$\int_R \left[\rho\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} - \nabla\cdot\boldsymbol{\sigma} + \rho\mathbf{b}\right]\mathrm{d}R = 0, \tag{3.42}$$

which can be written as

$$\rho\mathbf{a} = \nabla\cdot\boldsymbol{\sigma} + \rho\mathbf{b}, \tag{3.43}$$

assuming that the integrand of Equation (3.42) is continuous.

For multiphase materials the conservation of momentum for phase α is

$$\frac{\mathrm{d}_\alpha}{\mathrm{d}t}\int_R \rho^\alpha\mathbf{v}^\alpha\,\mathrm{d}R + \sum_\beta \mathbf{P}^{\beta\alpha} = \int_R \rho^\alpha\mathbf{b}^\alpha\,\mathrm{d}R + \int_{\partial R} \boldsymbol{\tau}^\alpha\,\mathrm{d}S, \tag{3.44}$$

where ρ^α is the averaged phase density defined in Equation (3.12), β denotes the other phases, and $\mathbf{P}^{\beta\alpha}$ the momentum between phases β and α. Saturated soil is a two-phase medium where the momentum exchange between the water and solid phases is written as

$$\mathbf{P}^{\mathrm{sw}} = n\frac{\rho_\mathrm{w} g}{k}n(\mathbf{w}-\mathbf{v}) = -\mathbf{P}^{\mathrm{ws}}, \tag{3.45}$$

where n is the porosity, k the hydraulic conductivity, ρ_w the density of water, g the magnitude of the acceleration of gravity, and $\mathbf{w}$ and $\mathbf{v}$ the velocity of water and solid phase, respectively.

Using Equations (3.44) and (3.45) and performing the same steps as for Equation (3.40), the balance of momentum for the water phase ($\alpha = \mathrm{w}$) can be written as

$$\rho_\mathrm{w}\mathbf{a}_\mathrm{w} = \nabla p - \frac{\rho_\mathrm{w} g}{k}n(\mathbf{w}-\mathbf{v}) + \rho_\mathrm{w}\mathbf{b}, \tag{3.46}$$

where $\mathbf{a}_\mathrm{w} = \mathrm{d}\mathbf{w}/\mathrm{d}t$ and $\mathbf{b}$ is the body force. For the $\mathbf{u}$-p formulation, Assumption 6 is applied, which means that $\mathbf{a}_\mathrm{w}$ is replaced with the solid phase acceleration $\mathbf{a}_\mathrm{s}$. If inertial forces of water are neglected, $\mathbf{a}_\mathrm{w} = \mathbf{0}$ and the left-hand side of Equation (3.46) vanishes.

The conservation of momentum equation for the mixture is obtained by summing the conservation of momentum equations for all the phases in the material. Consequently, the interphase momentums $\mathbf{P}^{\alpha\beta}$ cancel eachother out, as seen from Equation (3.45). This leaves

$$n\rho_\mathrm{w}\mathbf{a}_\mathrm{w} + (1-n)\rho_\mathrm{s}\mathbf{a}_\mathrm{s} = n\boldsymbol{\nabla} p + n\rho_\mathrm{w}\mathbf{b} + (1-n)\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}_\mathrm{s} + (1-n)\rho_\mathrm{s}\mathbf{b}. \tag{3.47}$$

Inserting Equation (3.15) into the solid stress divergence term and grouping stress terms gives

$$(1-n)\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}_\mathrm{s} + n\boldsymbol{\nabla} p = \boldsymbol{\nabla}\cdot\left[\boldsymbol{\sigma}' + (1-n)p\mathbf{I}\right] + n\boldsymbol{\nabla} p = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}' + \boldsymbol{\nabla} p. \tag{3.48}$$

Inserting the above equation and the definition of mixture density of Equation (3.11) into Equation (3.47) gives the momentum balance of the mixture as

$$n\rho_\mathrm{w}\mathbf{a}_\mathrm{w} + (1-n)\rho_\mathrm{s}\mathbf{a}_\mathrm{s} = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}' + \boldsymbol{\nabla} p + \rho\mathbf{b} \tag{3.49}$$

for the $\mathbf{v}$-$\mathbf{w}$ formulation, where $\mathbf{a}_\mathrm{w} = \dot{\mathbf{w}}$ and $\mathbf{a}_\mathrm{s} = \dot{\mathbf{v}} = \ddot{\mathbf{u}}$.

For the $\mathbf{u}$-p formulation, Assumption 6 is applied, which reduces Equation (3.49) to

$$\rho\mathbf{a}_\mathrm{s} = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}' + \boldsymbol{\nabla} p + \rho\mathbf{b}, \tag{3.50}$$

because $\mathbf{a}_\mathrm{w} \approx \mathbf{a}_\mathrm{s} = \ddot{\mathbf{u}}$. Note that Equation (3.50) is equal to Equation (3.43) with the total stress $\boldsymbol{\sigma}$ replaced by Equation (3.14). For complete quasi-static assumptions, the left-hand side of Equation (3.50) vanishes.

3.3.3 Initial and boundary conditions

To complete the description of the boundary value problem, initial and boundary conditions are needed. *Initial conditions* provide the starting point of the solution at time $t = t_0$. Kinematic variables as displacement $\mathbf{u}_0$ and velocity $\mathbf{v}_0 = \dot{\mathbf{u}}_0$ are given in addition to properties such as initial stress $\boldsymbol{\sigma}_0$. The initial conditions for stresses are:

$$\boldsymbol{\sigma}'(\mathbf{x}, t_0) = \boldsymbol{\sigma}'_0 \qquad \text{and} \qquad p(\mathbf{x}, t_0) = p_0, \tag{3.51}$$

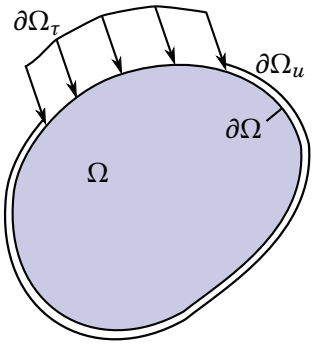


Figure 3.4: Boundary conditions of domain Ω. Together, the traction boundary condition $\partial\Omega_{\boldsymbol{\tau}}$ and the displacement boundary condition $\partial\Omega_{\mathbf{u}}$ cover the boundary $\partial\Omega$ of the domain Ω.

where $\boldsymbol{\sigma}'_0$ is the initial effective stress and p_0 the inital pore water pressure in the soil. The initial conditions for the kinematic variables are

$$\mathbf{v}(\mathbf{x}, t_0) = \mathbf{v}_0 = \dot{\mathbf{u}}_0 \qquad \text{and} \qquad \mathbf{w}(\mathbf{x}, t_0) = \mathbf{w}_0, \tag{3.52}$$

where the $\mathbf{u}$-p formulation does not use water velocity $\mathbf{w}$.

Boundary conditions provide the solution at the domain boundaries independent of time that is compatible with the initial conditions. The most common type of boundary conditions are *Dirichlet* and *Neumann* boundary conditions, which for the solid phase translates to displacements and traction, respectively. The surface boundary $\partial\Omega$ is divided into two disjoint sets of boundary conditions [51] so that

$$\partial\Omega_{\mathbf{u}} \cap \partial\Omega_{\boldsymbol{\tau}} = \emptyset \qquad \text{and} \qquad \partial\Omega_{\mathbf{u}} \cup \partial\Omega_{\boldsymbol{\tau}} = \partial\Omega, \tag{3.53}$$

that is, the traction boundary $\partial\Omega_{\boldsymbol{\tau}}$ and displacement boundary $\partial\Omega_{\mathbf{u}}$ do not overlap, but together they cover the entire surface boundary $\partial\Omega$ (Figure 3.4). With the addition of a water phase, the surface boundary $\partial\Omega$ can be divided into another group of disjoint sets that represent the water phase boundary. For the $\mathbf{u}$-p formulation, the boundaries can be written as

$$\partial\Omega_{\mathbf{q}} \cap \partial\Omega_p = \emptyset \qquad \text{and} \qquad \partial\Omega_{\mathbf{q}} \cup \partial\Omega_p = \partial\Omega, \tag{3.54}$$

where $\partial\Omega_{\mathbf{q}}$ is the surface boundary subset on which the boundary flux is applied, while $\partial\Omega_p$ is the prescribed water pressure boundary. For the $\mathbf{v}$-$\mathbf{w}$ formulation, the flux boundary $\partial\Omega_{\mathbf{q}}$ is replaced with the water velocity boundary $\partial\Omega_{\mathbf{w}}$. The water phase boundaries are independent of the solid phase boundaries.

The kinematic boundary conditions are defined on $\partial\Omega_{\mathbf{u}}$ and $\partial\Omega_{\mathbf{q}}$ for the solid and water phase, respectively, as

$$\mathbf{u}(\mathbf{x}, t) = \bar{\mathbf{u}}(t) \qquad \text{on } \partial\Omega_{\mathbf{u}} \tag{3.55}$$

$$\mathbf{q}(\mathbf{x}, t) = \bar{\mathbf{q}}(t) \qquad \text{on } \partial\Omega_{\mathbf{q}} \tag{3.56}$$

or for the **v**-**w** formulation

$$\mathbf{v}(\mathbf{x},t) = \bar{\mathbf{v}}(t) \qquad \text{on } \partial\Omega_{\mathbf{u}} \tag{3.57}$$

$$\mathbf{w}(\mathbf{x},t) = \bar{\mathbf{w}}(t) \qquad \text{on } \partial\Omega_{\mathbf{w}}. \tag{3.58}$$

The Cauchy surface traction $\bar{\boldsymbol{\tau}}$ is given by

$$\boldsymbol{\sigma}(\mathbf{x},t)\cdot\mathbf{n} = \bar{\boldsymbol{\tau}} \qquad \text{on } \partial\Omega_{\boldsymbol{\tau}}, \tag{3.59}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress, $\mathbf{n}$ is the boundary surface normal, and $\bar{\boldsymbol{\tau}} = \boldsymbol{\tau}(\mathbf{x})\cdot\mathscr{T}(t)$ with $\mathscr{T}(t)$ a function governing the change of loading with time. The prescribed pore water pressure is

$$p(\mathbf{x},t) = \bar{p} \qquad \text{on } \partial\Omega_{p}, \tag{3.60}$$

where $\bar{p} = p(\mathbf{x})\cdot\mathscr{T}(t)$.

3.3.4 Constitutive model

The constitutive laws that govern the relation between stresses and strains give the effective stress response of the soil. For large deformations it is beneficial to let the stress depend on the deformation gradient $\mathbf{F}$ [95], as

$$\boldsymbol{\sigma}' = \mathscr{G}(\mathbf{F}), \tag{3.61}$$

where $\mathscr{G}$ is some constitutive model.

The material response should not depend on the frame of reference, it should be *objective*. That is, no matter the reference point, the material experience the same level of stress. Both the Cauchy stress tensor (the true stress) $\boldsymbol{\sigma}$ and the deformation gradient $\mathbf{F}$ are objective. A material on the form of Equation (3.61) is *Cauchy-elastic* and independent of deformation path.

3.3.5 Darcy's law

Fluid flow through soil is most often laminar due to the dissipation by the viscous interaction between pore fluid and solid grains defined in Equation (3.45). Therefore the flow can be described by Darcy's law, which is a linear relationship between the specific discharge $\mathbf{q}$ defined in Equation (3.30) and the *hydraulic gradient*, i.e. change in pore pressure head. The generalised dynamic Darcy's law [96, 97] is

$$n(\mathbf{w}-\mathbf{v}) = -\frac{k}{\rho_{\mathrm{w}} g}\left[\boldsymbol{\nabla} p + \rho_{\mathrm{w}}\mathbf{a}_{\mathrm{w}} - \rho_{\mathrm{w}}\mathbf{b}\right], \tag{3.62}$$

where the *hydraulic conductivity*[6] [m/s] is given by

$$k = \kappa \frac{\rho_{\mathrm{w}} g}{\mu_{\mathrm{w}}}, \tag{3.63}$$

where $\kappa\,[\mathrm{m}^2]$ is the *absolute permeability* tensor of the porous medium, $\rho_{\mathrm{w}}\,[\mathrm{kg/m}^3]$ is the density of water, $g\,[\mathrm{m/s}^2]$ is the gravitational acceleration, and $\mu_{\mathrm{w}}\,[\mathrm{kg/(ms)}]$ is the dynamic viscosity of water. While κ is a property of the soil skeleton, the hydraulic conductivity is influenced by pore fluid properties.

Absolute permeability κ is expected to change with porosity n, as smaller voids would mean less water to flow through the soil skeleton. In addition, soil is anisotropic and highly nonhomogeneous spatially. Therefore, a more realistic representation of the permeability κ is a tensor that varies spatially and with porosity: $\boldsymbol{\kappa}(\mathbf{x}, n)$. In this thesis, however, permeability is regarded as constant both spatially and with respect to porosity, in addition to being isotropic. This reduces the permeability tensor to a constant scalar.

3.4 Chapter summary

This chapter presented and derived the mathematical framework for modelling saturated porous media. The strong form of the governing equations was derived for two formulations: **u**-p formulation and **v**-**w** formulation. Large deformations have been considered.

[6] Also called the *permeability coefficient*.

You know my methods.
Apply them.

Sir Arthur Conan Doyle
The Sign of Four (1890)

CHAPTER 4

MATERIAL POINT METHOD

This chapter aims to provide the foundation upon which the main contribution of this thesis is built. The mathematical framework of both an explicit MPM and an implicit MPM formulation is presented here. First, the governing equations are derived from the conservation of mass and momentum. Then, the governing equations are discretised in space and time for both explicit and implicit time integrations.

Chapter outline

4.1 The concept of MPM

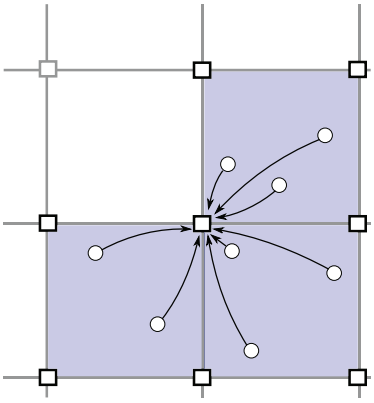
Large deformation problems with a Lagrangian description can be fixed by using smaller time steps or remeshing, but both solutions have drawbacks. Remeshing, for instance, can be computationally expensive and also troublesome with history-dependent materials due to the mapping of material state variables from the old to the new mesh [38]. The material point method avoids both the Lagrangian problem of mesh distortion and the Eulerian problem with the convective term. The Lagrangian material points or *particles* eliminate the convective term of the material time derivative, while the Eulerian background grid avoids mesh distortion.

MPM uses both a Lagrangian and a Eulerian representation. The continuum body is discretised by Lagrangian material points that can move through a Eulerian computational grid. The grid can be arbitrarily defined and redefined. Most often, the grid is fixed in time, although some moving-mesh formulations have been used [53, 77]. The momentum equations are solved at grid nodes, while the constitutive equation or material response is solved at material points after they have deformed together with the grid. Therefore, the convection term of the Eulerian material time derivative disappears from the grid solution. The grid is reset after the material response is computed. The history-dependent material parameters are not mapped back to the grid, effectively walking through the grid.

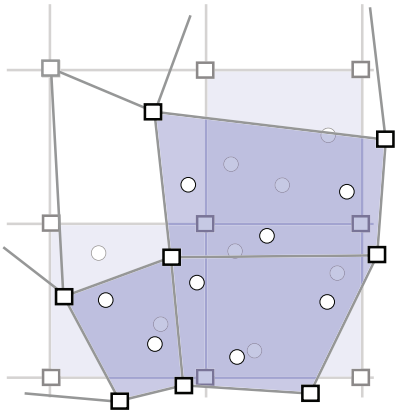
Figure 4.1 shows the 3 phases of the MPM algorithm as described by [37]. First, there is an initialisation or mapping phase where particle information is interpolated to the nodes of the background grid **(a)**. Then, the equation of motion is solved at the nodes in a Lagrangian phase where the background grid serves as an updated Lagrangian frame **(b)**. The particles move relative to the nodes, as the shape functions use the element local coordinates, eliminating the convective term of the Eulerian material time derivative. The convective phase is the final phase where the particles are updated with their new positions and velocities, while the grid is reset to its initial configuration or redefined to a new configuration **(c)**.

4.2 Weak form of the governing equations

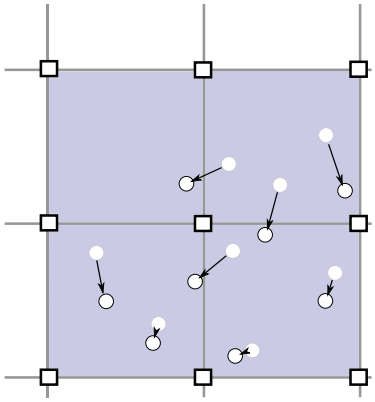
The mathematical framework describing MPM is based on the equations for conservation of mass and the conservation of momentum. These equations are then transformed into *weak* or *integral* form from the *strong* or *differential* form. The weak form loosens the continuity requirements of



(a) Initialisation or mapping phase. Particle momentums, accelerations and masses are interpolated to the nodes.



(b) Lagrangian phase. The background grid serves as an updated Lagrangian frame. The grid deforms and the particles move relative to the grid.



(c) Convective phase. The new particle locations and velocities are updated, and the grid data is deleted.

Figure 4.1: The 3 main phases of MPM. The first is the mapping phase **(a)**, the second is the Lagrangian phase **(b)**, and the third is the convective phase **(c)**. The squares are nodes, with the active nodes being black. The circles are particles. The shaded cells are grid cells that contain particles.

the solution and is, therefore, easier to solve. In addition, the derivatives of the field variables are replaced by derivatives of interpolation or shape functions, which are easier to obtain than the true derivatives of the field variables. This approach is similar to FEM.

Let us first define functional spaces for the solution for the displacements $\mathbf{u}$

$$\mathscr{U} = \left\{ \mathbf{u} : \Omega \to \mathbb{R}^3 \quad | \quad \mathbf{u} \in H^1, \quad \mathbf{u} = \bar{\mathbf{u}} \text{ on } \partial\Omega_{\mathbf{u}} \right\} \tag{4.1}$$

where H^1 is the Sobolev space of degree 1. This means that the displacement field $\mathbf{u}$ is discretised by linear (degree 1) shape functions. We are seeking the solution of a discrete approximation of this space, that is, finding $\mathbf{u}_h \in \mathscr{U}_h$ where the approximated space $\mathscr{U}_h$ is the corresponding subspace of the solution space $\mathscr{U}$.

The space of the test function $\boldsymbol{\eta}$ is

$$\mathscr{U}_0 = \left\{ \boldsymbol{\eta} : \Omega \to \mathbb{R}^3 \quad | \quad \boldsymbol{\eta} \in H^1, \quad \boldsymbol{\eta} = \mathbf{0} \text{ on } \partial\Omega_{\mathbf{u}} \right\} \tag{4.2}$$

The approximate test function $\boldsymbol{\eta}_h$ is defined on the corresponding subspace $\mathscr{U}_{0,h}$. If the test function $\boldsymbol{\eta}$ is a kinematically admissible virtual displacement of the system, then this is the *principle of virtual work*. This means that the test function $\boldsymbol{\eta}$ should be continuous and *kinematic admissible* on the domain Ω, and $\mathbf{0}$ on the boundary $\partial\Omega_{\mathbf{u}}$. The weak solution of the problem is obtained by multiplying the strong form of the conservation of linear momentum equation (3.43) with the corresponding test function $\boldsymbol{\eta}_h$ and integrating over the domain Ω, giving

$$\int_\Omega \boldsymbol{\eta}_h \cdot \rho \mathbf{a}_h \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{\eta}_h : (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_h) \, \mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \cdot \rho \mathbf{b}_h \, \mathrm{d}\Omega, \tag{4.3}$$

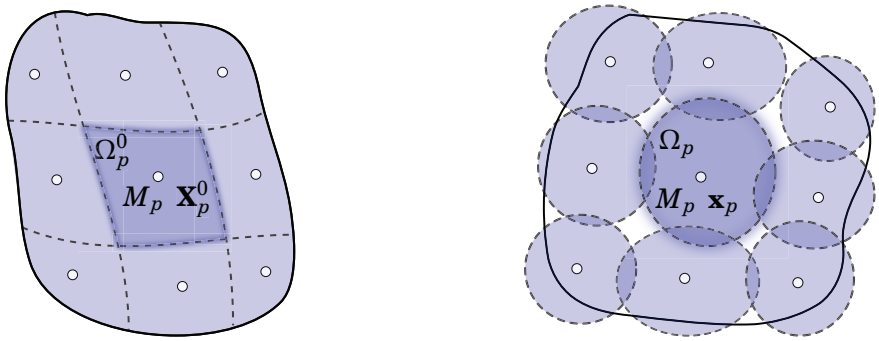
where subscript h denote the approximation of the field variable.

Using integration by parts on the first term on the right-hand side of Equation (4.3) yields

$$\int_\Omega \boldsymbol{\eta}_h : (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_h) \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{\nabla} \cdot \left(\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h \right) \mathrm{d}\Omega - \int_\Omega \boldsymbol{\nabla} \boldsymbol{\eta}_h : \boldsymbol{\sigma}_h \, \mathrm{d}\Omega. \tag{4.4}$$

The divergence theorem and Cauchy's formula $\boldsymbol{\tau} = \boldsymbol{\sigma}_h \cdot \mathbf{n}$ can then be applied to the first term on the right hand side of Equation (4.4), so that

$$\int_\Omega \boldsymbol{\nabla} \cdot \left(\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h \right) \mathrm{d}\Omega = \int_{\partial\Omega} \boldsymbol{\eta}_h \cdot \boldsymbol{\sigma}_h \cdot \mathbf{n} \, \mathrm{d}S = \int_{\partial\Omega_\tau} \boldsymbol{\eta}_h \cdot \bar{\boldsymbol{\tau}} \, \mathrm{d}S, \tag{4.5}$$



(a) Initial material point discretisation at t_0

(b) Current material point discretisation at t

Figure 4.2: Material point discretisation in the initial configuration **(a)** and current configuration **(b)**. Particle volumes are tracked, but particle shape is not. Particle mass remains constant, thus fulfilling conservation of mass (Equation (3.23)).

where the integral is now a surface integral with $\mathbf{n}$ the surface normal and $\bar{\boldsymbol{\tau}}$ the prescribed traction. Consequently, Equation (4.5) represents the traction boundary condition. The boundary of the domain is denoted by $\partial\Omega = \partial\Omega_{\mathbf{u}} \cup \partial\Omega_{\boldsymbol{\tau}}$ where $\partial\Omega_{\mathbf{u}}$ and $\partial\Omega_{\boldsymbol{\tau}}$ are non-overlapping parts of the boundary with prescribed displacement and traction, respectively.

Reinserting Equations (4.4) and (4.5) into Equation (4.3), we arrive at the weak form of the momentum equation, defined as the residual $\mathbf{r}_{\text{mom}}$:

$$\mathbf{r}_{\text{mom}} = -\int_\Omega \boldsymbol{\eta}_h \cdot \rho \mathbf{a}_h \,\mathrm{d}\Omega - \int_\Omega \boldsymbol{\sigma}_h : \nabla \boldsymbol{\eta}_h \,\mathrm{d}\Omega + \int_{\partial\Omega_\tau} \boldsymbol{\eta}_h \cdot \bar{\boldsymbol{\tau}} \,\mathrm{d}S + \int_\Omega \boldsymbol{\eta}_h \cdot \rho \mathbf{b}_h \,\mathrm{d}\Omega = \mathbf{0}. \tag{4.6}$$

4.3 Spatial discretisation

MPM uses two discretisations: the computational grid (Eulerian) and the material points (Lagrangian). The entire computational domain with or without material is divided into elements, while the material body is divided into smaller volumes called *material points*.

4.3.1 Particle discretisation

We divide the original domain Ω^0 into subdomains Ω_p^0, where $p = 1, 2, \ldots, N_\text{p}$ with N_p as the total number of subdomains, as illustrated in Figure 4.2(a).

Each of these subdomains is referenced by a material point located at the subdomain's centroid defined by the position vector $\mathbf{X}_p^0$. The material point carry the concentrated mass M_p which is defined by

$$M_p = \rho_p V_p, \tag{4.7}$$

where ρ_p is the material point density and V_p the volume of the subdomain represented by the particle. In the current configuration at some time t, the original subdomains will have deformed into Ω_p at a new position $\mathbf{x}_p$, see Figure 4.2(b). The shapes of the subdomains are now unknown[1], but they still carry the same mass as before. Consequently, the conservation of mass (Equation (3.23)) is automatically fulfilled.

The Dirac delta function is used as the basis function for the mass density so that

$$\rho(\mathbf{x}, t) = \sum_{p=1}^{N_\mathrm{p}} M_p \delta\left[\mathbf{x} - \mathbf{x}_p\right]. \tag{4.8}$$

This mass density function lumps the mass into one infinitesimal point. Alternatively, smoother density functions such as given by GIMP [54] may be used instead. A useful property of the Dirac delta function is that

$$\int_x f(x)\delta(x-a)\,\mathrm{d}x = f(a). \tag{4.9}$$

Reorganising terms of the weak form of the momentum equation (4.6), we have

$$\int_\Omega \left(\boldsymbol{\eta}_h \cdot \rho\mathbf{a} + \rho\boldsymbol{\sigma}^\mathrm{s} : \nabla\boldsymbol{\eta}_h\right)\mathrm{d}\Omega = \int_\Omega \boldsymbol{\eta}_h \cdot \rho\mathbf{b}\,\mathrm{d}\Omega + \int_{\partial\Omega} \boldsymbol{\eta}_h \cdot \bar{\boldsymbol{\tau}}\,\mathrm{d}S, \tag{4.10}$$

where $\rho = \rho(\mathbf{x}, t)$. Stresses $\boldsymbol{\sigma}$ is a property of the material points. Therefore, $\boldsymbol{\sigma}$ is replaced by a *specific stress* $\boldsymbol{\sigma}^\mathrm{s}$ defined as

$$\boldsymbol{\sigma} = \rho\boldsymbol{\sigma}^\mathrm{s}, \tag{4.11}$$

which is beneficial when discretising the weak form of Equation (3.43) [5, 36]. So after inserting Equation (4.8) into Equation (4.10) we obtain a discretised equation:

$$\sum_{p=1}^{N_\mathrm{p}} M_p\left[\boldsymbol{\eta}_p \cdot \mathbf{a}_p + \boldsymbol{\sigma}_p^\mathrm{s} : \nabla\boldsymbol{\eta}_p\right] = \sum_{p=1}^{N_\mathrm{p}} M_p \boldsymbol{\eta}_p \cdot \mathbf{b}_p + \int_{\partial\Omega} \boldsymbol{\eta} \cdot \bar{\boldsymbol{\tau}}\,\mathrm{d}S, \tag{4.12}$$

where notation has been simplified so that subscript p means that the variable is evaluated at $\mathbf{x}_p$.

[1] CPDI is an alternative formulation of MPM that tracks the shape of material points.

4.3.2 Grid discretisation

Let us now define a grid where the coordinates of the grid are denoted by $\mathbf{x}_i$ with $i = 1, 2, \ldots, N_\mathrm{n}$ and N_n the total number of grid nodes. The subspaces of the spaces defined in Equations (4.1) and (4.2) are finite-dimensional, and allows an approximate solution of the field as

$$\mathbf{u}_h = \sum_{i=1}^{N_\mathrm{n}} \mathbf{u}_i(t) N_i(\mathbf{x}), \tag{4.13}$$

which is equivalent to FEM interpolation. In this work, N_i is a linear shape function at node i defined as

$$N_i(\mathbf{x}) = \begin{cases} \prod_{k=1}^{d} \left(1 - \frac{\left(x_k - x_{i,k}\right)}{\Delta x_k}\right) & \text{if } |\mathbf{x} - \mathbf{x}_i| < \Delta\mathbf{x} \\ 0 & \text{otherwise,} \end{cases} \tag{4.14}$$

where d denotes the number of dimensions, $\mathbf{x}_i$ is the spatial position of node i, and $\Delta\mathbf{x}$ is the cell dimensions or grid spacing. While linear shape functions are regarded as the standard of MPM, there are higher-order alternatives [73].

As previously discussed, the convective term of the material time derivative of the nodal basis functions disappears since they are applied when the grid deforms with the particles at the same relative initial positions. Consequently, the velocity $\mathbf{v}_h$, the acceleration $\mathbf{a}_h$, and the test function $\boldsymbol{\eta}_h$ can all be represented by the same basis functions:

$$\mathbf{v}_h(\mathbf{x}, t) = \sum_{i=1}^{N_\mathrm{n}} \mathbf{v}_i(t) N_i(\mathbf{x}) \tag{4.15}$$

$$\mathbf{a}_h(\mathbf{x}, t) = \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_i(t) N_i(\mathbf{x}) \tag{4.16}$$

$$\boldsymbol{\eta}_h(\mathbf{x}, t) = \sum_{i=1}^{N_\mathrm{n}} \boldsymbol{\eta}_i(t) N_i(\mathbf{x}). \tag{4.17}$$

The shorthand notation $N_{ip} = N_i\left(\mathbf{x}_p\right)$ will be used hereafter to denote shape functions for node i evaluated at material point location $\mathbf{x}_p$. The subscript h will also be dropped because only the approximate solution will be discussed in the rest of this chapter.

The momentum equation is solved at nodes, so we can insert Equations (4.16) and (4.17) at $\mathbf{x} = \mathbf{x}_p$ into Equation (4.12), thus obtaining

$$\sum_{p=1}^{N_\mathrm{p}} M_p \left[\sum_{i=1}^{N_\mathrm{n}} \boldsymbol{\eta}_i N_{ip}\right]\left[\sum_{j=1}^{N_\mathrm{n}} \mathbf{a}_j N_{jp}\right] + \sum_{p=1}^{N_\mathrm{p}} M_p \left[\sum_{i=1}^{N_\mathrm{n}} \boldsymbol{\eta}_i \nabla N_{ip}\right]\boldsymbol{\sigma}^\mathrm{s} = \sum_{p=1}^{N_\mathrm{p}} M_p \left[\sum_{i=1}^{N_\mathrm{n}} \boldsymbol{\eta}_i N_{ip}\right]\mathbf{b}_p + \sum_{i=1}^{N_\mathrm{n}} \boldsymbol{\eta}_i N_{ip}\hat{\boldsymbol{\tau}}_i, \quad (4.18)$$

where

$$\hat{\boldsymbol{\tau}}_i = \int_{\partial\Omega} N_i(\mathbf{x})\bar{\boldsymbol{\tau}}(\mathbf{x})\,\mathrm{d}S. \quad (4.19)$$

The difference from FEM is that the location of the material point $\mathbf{x}_p$ is arbitrary within the element, and that the material point can move freely between elements.

Since $\boldsymbol{\eta}_i$ is arbitrary and non-zero, we can eliminate it. Then we obtain the following equation for each node $i = 1, 2, \ldots, N_\mathrm{n}$:

$$\underbrace{\sum_{p=1}^{N_\mathrm{p}} M_p N_{ip}\left(\sum_{j=1}^{N_\mathrm{n}} \mathbf{a}_j N_{jp}\right)}_{\sum_{j=1}^{N_\mathrm{n}} m_{ij}\mathbf{a}_j} + \underbrace{\sum_{p=1}^{N_\mathrm{p}} M_p \nabla N_{ip} : \boldsymbol{\sigma}^\mathrm{s}}_{\mathbf{F}_\mathrm{int}} = \underbrace{\sum_{p=1}^{N_\mathrm{p}} M_p N_{ip}\mathbf{b}_p + \hat{\boldsymbol{\tau}}_i}_{\mathbf{F}_\mathrm{ext}}. \quad (4.20)$$

In the term of nodal internal force $\mathbf{F}_\mathrm{int}$ we note that $M_p\boldsymbol{\sigma}^\mathrm{s} = M_p\rho_p\boldsymbol{\sigma} = V_p\boldsymbol{\sigma}$, so that

$$\mathbf{F}_\mathrm{int} = \sum_{p=1}^{N_\mathrm{p}} V_p \nabla N_{ip} : \boldsymbol{\sigma}, \quad (4.21)$$

where $\boldsymbol{\sigma}$ is evaluated at the material points. In Equation (4.20) a *consistent* mass matrix

$$m_{ij} = \sum_{p=1}^{N_\mathrm{p}} M_p N_i N_j \quad (4.22)$$

is introduced. It is, however, more common to use a *diagonally lumped* mass matrix with MPM [19, 37, 39], although some MPM formulations use a consistent mass matrix [17]. Additionally, in MPM the nodal mass matrix is assembled at the beginning of every time step, as the mass associated with the nodes will change as particles move through the mesh. In FEM the nodal mass matrix is assembled once. Therefore, the consistent mass

matrix in FEM may be inverted and stored for all subsequent time steps. In MPM the consistent mass matrix would have to be inverted or solved for at all time steps.

A diagonally lumped mass matrix is a mass matrix with diagonal terms only. One of the main benefits is that a diagonal matrix can always be inverted, which is done by simply inverting the individual entries. A diagonal matrix can be represented by a vector in computer codes, thus saving computational space. A drawback, on the other hand, is that some numerical dissipation of kinematic energy occurs [98].

There exist several approaches to mass matrix lumping [99]. Here the simple approach of row summation lumping is used [51]. That is, each diagonal entry in the lumped mass matrix is the corresponding row sum of the consistent mass matrix:

$$m_i = \sum_{j=1}^{N_\mathrm{n}} m_{ij} = \sum_{j=1}^{N_\mathrm{n}} \sum_{p=1}^{N_\mathrm{p}} M_p N_i N_j = \sum_{p=1}^{N_\mathrm{p}} M_p N_i. \tag{4.23}$$

Introducing the lumped mass matrix defined in Equation (4.23) into Equation (4.20) and grouping all force vectors to the right hand side, we have

$$m_i \mathbf{a} = \mathbf{F}_\mathrm{ext} - \mathbf{F}_\mathrm{int}, \tag{4.24}$$

which is the equation of conservation of linear momentum at node i. This equation is identical to the FEM equivalent, demonstrating the similarities of MPM and FEM. Consequently, many features of FEM can be applied to MPM. Examples of further improvement of MPM based on FEM are the implicit time integration proposed by Guilkey & Weiss [18] and the random material point method proposed by Wang [100].

Equation (4.24) is only semi-discretised, as it is only the spatial partial derivatives that have been discretised, not the temporal. Refer to Section 4.5 for explicit time integration and to Section 4.6 for implicit time integration of the governing equations.

4.4 Applying initial and boundary conditions

In MPM the initial conditions defined in Equations (3.51) and (3.52) are directly applied to the particles during the creation procedure, along with material properties. Due to the two discretisation levels of MPM, the boundary conditions are applied to both boundaries depending on boundary type, as shown in Figure 4.3.

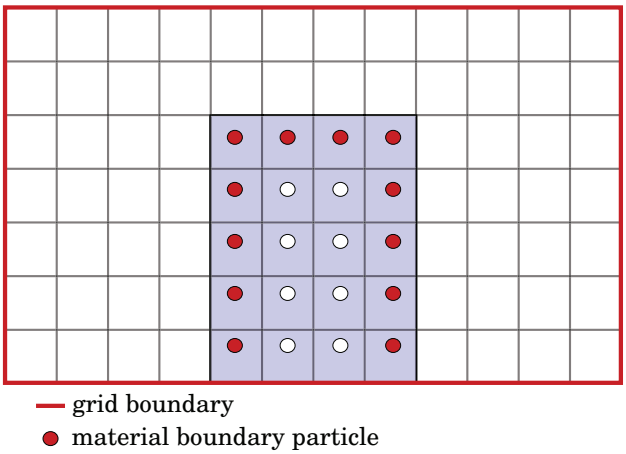


Figure 4.3: In MPM boundary conditions can be applied to either the grid (thick dark line) or to boundary particles (filled circles).

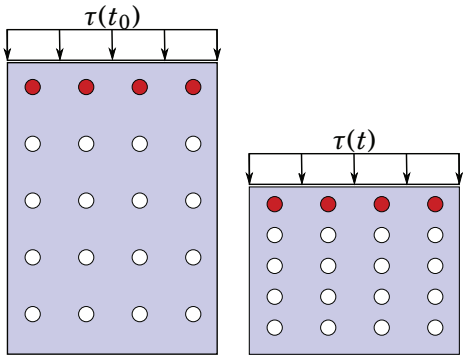


Figure 4.4: Traction applied to boundary particles (filled) is moving with the deforming body.

4.4.1 Boundary conditions applied to particles

The traction boundary is applied on boundary particles, as shown in Figure 4.4. This allows the traction boundary to move with the body, as opposed to traction applied to the fixed background grid. During particle creation at the very start of the simulation, the boundary particles are located and flagged. Then the surface area A_τ where the traction is applied and the number of boundary particles N_{bp} are computed. Then at each time step t the boundary particles are assigned a traction force as follows:

$$\mathbf{f}_p^{\text{trac}} = \frac{A_\tau}{N_{\text{bp}}} \boldsymbol{\tau}(t). \tag{4.25}$$

The Uintah computational framework [29] has the special case of pressure traction boundary conditions available, meaning stress is applied in the direction of the boundary surface normal. This means that the surface traction $\boldsymbol{\tau}(t)$ is found from

$$\boldsymbol{\tau}(t) = p(t) \cdot \mathbf{n}, \tag{4.26}$$

where p is the prescribed pressure where the sign indicates direction of loading, and $\mathbf{n}$ is the boundary surface normal. Zero traction boundaries need not be defined explicitly.

An alternative to applying traction forces on a layer of boundary particles is to create a *moving mesh*, as described in [77]. The idea is that the boundary surface always aligns with the computational grid, in a manner similar to FEM. The moving mesh boundary can also be applied to kinematic boundary conditions, for instance, the application of prescribed displacements or velocities. However, in the case of large deformations, the moving mesh procedure can lead to mesh distortion, and it is also difficult to mesh properly if the boundary surface is irregular, such as the surface of a failing slope.

4.4.2 Boundary conditions applied to grid

Zero kinematic boundary conditions or symmetric boundary conditions are applied directly on the boundary nodes. These boundary conditions are applied by setting the nodal displacement or velocity component that is normal to the grid boundary to 0 before the equation of motion is computed. In general, the procedures for applying boundary conditions directly at grid nodes are similar to FEM.

4.5 Explicit material point method

The original material point method used explicit time integration [18, 37], and it is by far the most common in literature [8]. Explicit time integration makes advancing the solution in time straightforward because no iterations are needed per time step. The current solution is used together with a time step to achieve the new solution, on the form of a first order Taylor series expansion around the known value:

$$f^{n+1} = f^n + \left.\frac{\mathrm{d}f}{\mathrm{d}t}\right|_n \Delta t, \tag{4.27}$$

where f is some function of time, the differential $\mathrm{d}f/\mathrm{d}t$ is evaluated at n, Δt is the stride or time step, and superscripts n and $n+1$ denotes solutions at time t_n and $t_{n+1} = t_n + \Delta t$.

Using a lumped mass matrix and explicit time integration, we first solve for the nodal acceleration $\mathbf{a}_i$ from the equation of conservation of linear momentum

$$m_i \mathbf{a}_i = \mathbf{F}_{\text{ext}} - \mathbf{F}_{\text{int}}, \tag{4.28}$$

which is done by dividing with the nodal mass m_i. Equation (4.28) can be solved at all nodes independently due to the lumped mass matrix.

Then the particle velocities are computed directly from the nodal accelerations through

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \Delta t \sum_{i=1}^{N_\text{n}} \mathbf{a}_i^{n+1} N_{ip}^n, \tag{4.29}$$

where N_{ip}^n is simplified notation for $N_i(\mathbf{x}_p^n)$. This is done to avoid first updating the nodal velocities and then interpolating to the particles [37]. After obtaining the particle velocities, the nodal velocities are updated from the momentum as

$$m_i^n \mathbf{v}_i^n = \mathbf{p}_i^n, \tag{4.30}$$

where the nodal momentum $\mathbf{p}_i^n$ is the sum of particle momentums computed from the particle velocities by

$$\mathbf{p}_i^n = \sum_{p=1}^{N_\text{p}} M_p N_{ip}^n \mathbf{v}_p^n. \tag{4.31}$$

The reason for using the nodal accelerations to update the particle velocities is to overcome the *small mass problem* identified by Sulsky *et al.* [37]. The small mass problem arises when particles are positioned very close to one boundary in an otherwise empty element, as illustrated in Figure 4.5. The opposite nodes will then have very small masses due to the interpolation weight, and this may result in unphysically large nodal accelerations [18, 37, 52–54]. This approach does not eliminate these accelerations, as division by mass is still done to solve for the acceleration in Equation (4.28). However, by updating the nodal velocities through the particle momentums, the number of nodal mass divisions is minimized. In addition, the accelerations are smoothed, which is another feature that dissipates energy [52].

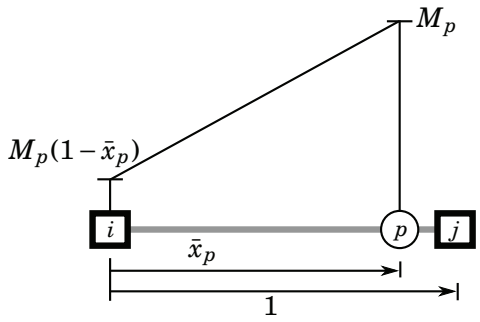


Figure 4.5: The small mass problem in 1D. When a particle p is very close to node j, the interpolated mass to the opposite node i will be very small as normalised position $\bar{x}_p$ of the particle approaches the normalised node spacing.

With the acceleration $\mathbf{a}_i^n$ from Equation (4.28) and velocity $\mathbf{v}_i^n$ from Equation (4.30), the updated nodal velocity is found from

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \mathbf{a}_i^n \Delta t, \tag{4.32}$$

which is a simple forward Euler step. The particle updated positions are computed from the nodal velocity, as

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{v}_i^{n+1} N_{ip}^n, \tag{4.33}$$

while the particle velocity is aleady updated from Equation (4.29).

During the Lagrangian step of solving the equations of motion, the grid is assumed to deform in the flow of material, keeping all particles in the same relative position as in the undeformed configuration. This means that the basis functions N_{ip}^n are valid throughout this phase before the grid is reset.

4.5.1 Critical time step

Explicit time integration is conditionally stable, which means that there will be no convergence and hence no solution if the time step is too large. This was pointed out by Courant *et al.* [101] and lead to the formulation of the CFL condition, which states that no information can travel past its immediate neighbours during one time step. This condition must be fulfilled to obtain stability, but it is not enough to guarantee stability.

An example of a violation of the condition would be a wave travelling through the material with a velocity $\mathbf{v}$ such that the travelled distance during one time step, $\Delta\mathbf{u} = \mathbf{v}\Delta t$, is greater than the grid spacing $\Delta\mathbf{x}$. In this case, the following equality most hold to obtain a solution:

$$\Delta\mathbf{u} = \mathbf{v}\Delta t \leq \Delta\mathbf{x}. \tag{4.34}$$

The largest speed in the material would give the lowest time step. Therefore the *dilational wave speed* of the material c is used to compute the critial time step, and is computed from

$$c = \sqrt{\frac{K + 4/3G}{\rho}}, \tag{4.35}$$

where ρ the material density, and K and G are the bulk and shear moduli, respectively. Since the equations of motion are solved at the nodes in MPM, the grid spacing determines Δt_c and not the distance between material points. The smallest grid spacing Δx_{min} across all dimensions is used to estimate Δt_c. With unstructured meshes, it may be difficult to find Δx_{min}, but this is not a problem with the MPM part of the Uintah computational framework [29] as it uses a fixed regular Cartesian mesh.

The critical time step can then be found from

$$\Delta t_c \leq \frac{\Delta x_{\text{min}}}{c}. \tag{4.36}$$

The critical time step Δt_c is updated at every time step because the particle density ρ changes with time. A coarser grid allows for larger time steps, but will also lead to a less accurate solution.

4.5.2 Outline of one time step

Refer to Figure 4.1 for an illustration of the phases described below. The explicit MPM code in the Uintah computational framework [29] performs one time step as follows:

1. Mapping phase

Particle quantities are mapped from particles to nodes.

1) Apply external loads on boundary particles
2) Map particle masses to nodes:

$$m_i^n = \sum_{p=1}^{N_p} M_p N_{ip}^n \tag{4.37}$$

3) Compute particle momentums:

$$\mathbf{p}_p^n = M_p \mathbf{v}_p^n \tag{4.38}$$

4) Compute nodal velocities from Equation (4.38):

$$\mathbf{v}_i^n = \frac{\sum_{p=1}^{N_\mathrm{p}} \mathbf{p}_p^n N_{ip}^n}{m_i^n} \tag{4.39}$$

5) Map particle external loads to node:

$$\mathbf{F}_{\mathrm{ext}}^n = \sum_{p=1}^{N_\mathrm{p}} \mathbf{F}_{p,\,\mathrm{ext}}^n N_{ip}^n \tag{4.40}$$

6) Compute nodal internal force from particle stress:

$$\mathbf{F}_{\mathrm{int}}^n = \sum_{p=1}^{N_\mathrm{p}} V_p \nabla N_{ip}^n : \boldsymbol{\sigma} \tag{4.41}$$

Note that subscript i is omitted for the notation of the internal and external nodal forces.

2. Lagrangian phase

Momentum equation (4.28) is solved for the nodal acceleration $\mathbf{a}_i^n$. Furthermore, the following is done:

1) Compute particle velocities from nodal acceleration

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_i^n N_{ip}^n \tag{4.42}$$

2) Compute nodal velocities

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \mathbf{a}_i^n \Delta t \tag{4.43}$$

3) Compute particle velocity gradients

$$\mathbf{L}_p^{n+1} = \nabla \mathbf{v}_p^* = \nabla \left(\sum_{i=1}^{N_\mathrm{n}} \mathbf{v}_i^{n+1} N_{ip}^n \right) \tag{4.44}$$

4) Update particle volume

5) Apply the constitutive model on particles to get $\boldsymbol{\sigma}_p^{n+1}$

3. Convective phase

Particles are moved and grid properties are deleted. The particle displacements and positions are updated from the nodal velocities:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{v}_i^{n+1} N_{ip}^n \tag{4.45}$$

4.6 Implicit material point method

This thesis extends the single-phase implementation of an implicit MPM found in the Uintah open-source computational framework [29]. Since the approach used in that framework is the tangential stiffness method proposed by Guilkey & Weiss [18], that is the approach that will be presented here.

We start from defining a residual $\mathbf{r}$ from Equation (4.24), noting that all kinematic properties are evaluated at timestep $n+1$, which gives

$$\mathbf{r} = \mathbf{M}\mathbf{a}^{n+1} - \mathbf{F}_\mathrm{ext}^{n+1} + \mathbf{F}_\mathrm{int}^{n+1} = \mathbf{0}. \tag{4.46}$$

This is the global system. In the rest of this section, superscript $n+1$ will be omitted when superscript k or $k+1$ is used.

The iterative Newton-Raphson method is applied to Equation (4.46) to obtain an approximation of the residual at iteration $k+1$, written as

$$\mathbf{r}^{k+1} \approx \mathbf{r}^k + \left.\frac{\partial \mathbf{r}}{\partial \mathbf{u}}\right|_k \mathrm{d}\mathbf{u}^k = \mathbf{0}. \tag{4.47}$$

The time differentials are integrated by use of the Newmark-β method [102] for the displacements $\mathbf{u}$:

$$\mathbf{a}^{n+1} = \frac{1}{\beta \Delta t^2}\left(\mathbf{u}^{n+1} - \mathbf{u}^n\right) - \frac{1}{\beta \Delta t}\mathbf{v}^n - \frac{1-2\beta}{\beta}\mathbf{a}^n \tag{4.48}$$

$$\mathbf{v}^{n+1} = \frac{\gamma}{\beta \Delta t}\left(\mathbf{u}^{n+1} - \mathbf{u}^n\right) + \left(1 - \frac{\gamma}{\beta}\right)\mathbf{v}^n + \Delta t\left(1 - \frac{\gamma}{2\beta}\right)\mathbf{a}^n \tag{4.49}$$

where $\gamma = 1/2$ and $\beta = 1/4$ corresponds to constant acceleration during the time interval Δt, while $\beta = 1/6$ corresponds to linear acceleration [102]. The equations for acceleration $\mathbf{a}^{n+1}$ and velocity $\mathbf{v}^{n+1}$ can be simplified by noticing that $\mathbf{u}^n = \mathbf{0}$ due to the grid resetting of MPM. In the derivations that follow, $\beta = 1/4$ and $\gamma = 1/2$.

In order to find the system stiffness matrix, that is, the differential in Equation (4.47), we differentiate the terms of Equation (4.46). The first term is

$$\frac{\partial \mathbf{a}^k}{\partial \mathbf{u}^k} = \frac{4}{\Delta t^2}. \tag{4.50}$$

Furthermore, we have that

$$\frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{u}} \equiv \mathbf{K}, \tag{4.51}$$

where $\mathbf{K} = \mathbf{K}_{\text{mat}} + \mathbf{K}_{\text{geo}}$ is the tangent stiffness matrix which in non-linear analysis consists of the material stiffness matrix

$$\mathbf{K}_{\text{mat}} = \sum_p \mathbf{B}^{\mathsf{T}} \mathbf{D} \mathbf{B} V_p, \tag{4.52}$$

and the non-linear geometric stiffness matrix

$$\mathbf{K}_{\text{geo}} = \sum_p \mathbf{B}_{\text{NL}}^{\mathsf{T}} \boldsymbol{\sigma} \mathbf{B}_{\text{NL}} V_p. \tag{4.53}$$

The strain-displacement matrix $\mathbf{B} = \nabla \mathbf{N}$ written out is

$$\begin{bmatrix} \mathrm{d}N_1/\mathrm{d}x & 0 & 0 & \cdots & \mathrm{d}N_8/\mathrm{d}x & 0 & 0 \\ 0 & \mathrm{d}N_1/\mathrm{d}y & 0 & \cdots & 0 & \mathrm{d}N_8/\mathrm{d}y & 0 \\ 0 & 0 & \mathrm{d}N_1/\mathrm{d}z & \cdots & 0 & 0 & \mathrm{d}N_8/\mathrm{d}z \\ \mathrm{d}N_1/\mathrm{d}y & \mathrm{d}N_1/\mathrm{d}x & 0 & \cdots & \mathrm{d}N_8/\mathrm{d}y & \mathrm{d}N_8/\mathrm{d}x & 0 \\ 0 & \mathrm{d}N_1/\mathrm{d}z & \mathrm{d}N_1/\mathrm{d}y & \cdots & 0 & \mathrm{d}N_8/\mathrm{d}z & \mathrm{d}N_8/\mathrm{d}y \\ \mathrm{d}N_1/\mathrm{d}z & 0 & \mathrm{d}N_1/\mathrm{d}x & \cdots & \mathrm{d}N_8/\mathrm{d}z & 0 & \mathrm{d}N_8/\mathrm{d}x \end{bmatrix}, \tag{4.54}$$

which is a $6 \times (3 \cdot 8)$ matrix. The numerical subscript denotes the node, and with a 3D Cartesian grid, there are 8 nodes per element.

The non-linear strain-displacement matrix $\mathbf{B}_{\text{NL}}$ is a $3 \times (3 \cdot 8)$ matrix:

$$\begin{bmatrix} \mathrm{d}N_1/\mathrm{d}x & 0 & 0 & \cdots & \mathrm{d}N_8/\mathrm{d}x & 0 & 0 \\ 0 & \mathrm{d}N_1/\mathrm{d}y & 0 & \cdots & 0 & \mathrm{d}N_8/\mathrm{d}y & 0 \\ 0 & 0 & \mathrm{d}N_1/\mathrm{d}z & \cdots & 0 & 0 & \mathrm{d}N_8/\mathrm{d}z \end{bmatrix}. \tag{4.55}$$

We assume $\mathbf{F}_{\text{ext}}^{n+1}$ to be independent of the displacement, and consequently its derivative with respect to the displacement is 0. Then, by inserting Equation (4.46) and the differentials defined in Equations (4.50) and (4.51) into Equation (4.47), we get

$$\mathbf{M}\mathbf{a}^k - \mathbf{F}_{\text{ext}}^{n+1} + \mathbf{F}_{\text{int}}^k + \left(\frac{4}{\Delta t^2} \mathbf{M} + \mathbf{K} \right) \mathrm{d}\mathbf{u}^{k+1} = \mathbf{0}. \tag{4.56}$$

Introducing $\bar{\mathbf{K}} \equiv \left(4/\Delta t^2\right)\mathbf{M} + \mathbf{K}$, inserting Equation (4.48), and grouping the terms with known quantities on the right hand side of the above equations, leads to the implicit tangent stiffness method for the material point method [18, 55]:

$$\bar{\mathbf{K}}\,\mathrm{d}\mathbf{u}^{k+1} = \mathbf{F}_{\text{ext}}^{n+1} - \mathbf{F}_{\text{int}}^{k} - \mathbf{M}\left(\frac{4}{\Delta t^2}\mathbf{u}^k - \frac{4}{\Delta t}\mathbf{v}^n - \mathbf{a}^n\right), \tag{4.57}$$

where $\mathbf{u}^n = \mathbf{0}$ because the grid is reset at the start of each time step. The above system of equations can be written more compactly as

$$\bar{\mathbf{K}}\,\mathrm{d}\mathbf{u}^{k+1} = -\mathbf{r}^k. \tag{4.58}$$

The iterations are run until the convergence criteria are met. Two criteria are defined, and both should be satisfied. There are a displacement based criterion and a criterion based on the product of the displacement increment and the residual:

$$\frac{\left\|\mathrm{d}\mathbf{u}^k\right\|}{\left\|\mathrm{d}\mathbf{u}^{\max}\right\|} < \mathrm{DTOL} \qquad \text{and} \qquad \frac{\left\|\mathrm{d}\mathbf{u}^{k+1}\mathbf{r}^{k+1}\right\|}{\left\|\mathrm{d}\mathbf{u}^{0}\mathbf{r}^{0}\right\|} < \mathrm{RTOL}, \tag{4.59}$$

where $\|\mathrm{d}\mathbf{u}^{\max}\|$ is the maximum value of the displacement increment norm, and $\left\|\mathrm{d}\mathbf{u}^0\mathbf{r}^0\right\|$ is the initial value of the corresponding norm. DTOL and RTOL are user-defined tolerances.

4.6.1 Outline of one time step

In the implicit MPM code distributed with the Uintah open-source computational framework [29], one time step is performed as follows:

1. Mapping phase

Particle quantities are mapped from particles to nodes. This is identical to the explicit MPM time step except that the computation of the internal force is moved to the iterative loop.

1) Apply external loads on boundary particles

2) Map particle masses to nodes:

$$m_i^n = \sum_{p=1}^{N_\text{p}} M_p N_{ip}^n \tag{4.60}$$

3) Compute particle momentums:

$$\mathbf{p}_p^n = M_p \mathbf{v}_p^n \tag{4.61}$$

4) Compute nodal velocities from Equation (4.61):

$$\mathbf{v}_i^n = \frac{\sum_{p=1}^{N_\mathrm{p}} \mathbf{p}_p^n N_{ip}^n}{m_i^n} \tag{4.62}$$

5) Map particle external loads to node:

$$\mathbf{F}_{\mathrm{ext}}^n = \sum_{p=1}^{N_\mathrm{p}} \mathbf{F}_{p,\,\mathrm{ext}}^n N_{ip}^n \tag{4.63}$$

Note that subscript i is omitted for the notation of the internal and external nodal forces.

2. Lagrangian phase

The system of equations (4.58) is solved iteratively.

1) Initialization. Compute $\boldsymbol{\sigma}_p^k(\mathbf{u}^k)$, V_p^k, $\bar{\mathbf{K}}$, $\mathbf{F}_{\mathrm{int}}(\boldsymbol{\sigma}^k)$, and $\mathbf{r}^k$.
2) Solve for displacement increment $\mathrm{d}\mathbf{u}^{k+1}$
3) Update grid kinematics by the Newmark β-method and the trapezoidal rule:

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathrm{d}\mathbf{u}^{k+1} \tag{4.64}$$

$$\mathbf{v}^{k+1} = \frac{2}{\Delta t}\mathbf{u}^{k+1} - \mathbf{v}^n \tag{4.65}$$

Note that $\Delta\mathbf{u}^{k+1} = \mathbf{u}^{k+1} - \mathbf{u}^n = \mathbf{u}^{k+1}$ since grid is reset at the beginning of each time step ($\mathbf{u}^n = \mathbf{0}$).

4) Check for convergence from

$$\frac{\left\|\mathrm{d}\mathbf{u}^{k+1}\right\|}{\left\|\mathrm{d}\mathbf{u}^{\max}\right\|} < \mathrm{DTOL} \qquad \text{and} \qquad \frac{\left\|\mathrm{d}\mathbf{u}^{k+1}\mathbf{r}^{k+1}\right\|}{\left\|\mathrm{d}\mathbf{u}^0\mathbf{r}^0\right\|} < \mathrm{RTOL} \tag{4.66}$$

If not both convergence criteria are met, points (1)-(4) are repeated until convergence criteria is met. Else, the converged state is saved and the grid acceleration computed from Equation (4.48).

3. Convection phase

Particles are moved and grid properties are deleted. The particle positions and velocities are updated from the nodal displacements and

accelerations:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \sum_{i=1}^{N_\mathrm{n}} \mathbf{u}_i^{n+1} N_{ip} \tag{4.67}$$

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \frac{\Delta t}{2}\left[\mathbf{a}_p^n + \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_i^{n+1} N_{ip}\right] \tag{4.68}$$

4.7 Chapter summary

In this chapter the single-phase the material point method (MPM) was presented and derived. After the MPM discretisation procedure, the system of equations is similar to the corresponding system of FEM. The system of equations is solved at the grid nodes in both MPM and FEM. Therefore, methods developed for FEM can be applied to MPM too.

Two time integration schemes have been presented: one explicit scheme using the simple forward Euler method, and one implicit scheme using Newmark-β method with coefficients that correspond to the trapezoidal rule. The explicit MPM must fulfil the Courant-Friedrichs-Lewy (CFL) condition in order to converge, while the implicit MPM has no such constraint on the time step other than the accuracy may be lower with larger time steps.

CHAPTER 5

COUPLED DYNAMIC VELOCITY-BASED MPM

This chapter presents the coupled **v**-**w** formulation with explicit time integration as described by Jassim *et al.* [9]. The approach acknowledges the difference between solid and water accelerations and keeps track of two separate velocity fields, one for each phase. It is a one-point formulation, and the points move according to the solid velocity. A single-phase explicit MPM code [29] is extended with this formulation in this study.

Chapter outline

5.1 Governing equations

The **v**-**w** formulation by Jassim *et al.* [9] is based on the conservation of linear momentum and mass as presented in Section 3.3. The relevant strong equations used to derive the weak formulation are repeated here for convenience.

First, we need the equation of the conservation of fluid linear momentum:

$$\rho_{\mathrm{w}}\mathbf{a}_{\mathrm{w}} = \nabla p - \frac{\rho_{\mathrm{w}} g}{k} n(\mathbf{w}-\mathbf{v}) + \rho_{\mathrm{w}}\mathbf{b}, \tag{5.1}$$

where ρ_{w} is the density of water, $\mathbf{a}_{\mathrm{w}}$ the acceleration of water, p pore pressure, k is the hydraulic conductivity, g is the magnitude of gravitational acceleration, n is the porosity, $\mathbf{w}$ and $\mathbf{v}$ are the water and solid velocities, respectively, and $\mathbf{b}$ represents body forces such as gravity.

Next, we need the equation of the conservation of the mixture linear momentum:

$$n\rho_{\mathrm{w}}\mathbf{a}_{\mathrm{w}} + (1-n)\rho_{\mathrm{s}}\mathbf{a}_{\mathrm{s}} = \nabla\cdot\boldsymbol{\sigma} + \rho\mathbf{b}, \tag{5.2}$$

where ρ_{s} is the density of the solid grains, $\rho = n\rho_{\mathrm{w}} + (1-n)\rho_{\mathrm{s}}$ is the saturated density of the soil, $\mathbf{a}_{\mathrm{s}}$ is the acceleration of the solid phase, and $\boldsymbol{\sigma} = \boldsymbol{\sigma}' + p\mathbf{I}$ is the total stress. Equations (5.1) and (5.2) will be turned into the weak form through the MPM discretisation procedure.

At the material point level, a constitutive model that describes the relationship between strain and effective stress is needed:

$$\boldsymbol{\sigma}' = \mathscr{G}(\mathbf{F}), \tag{5.3}$$

where $\mathscr{G}$ is some constitutive function that depends on the deformation gradient $\mathbf{F}$. When using explicit time integration, the time steps are often so small that a small strain assumption can be used. In that case, we could use a linear elastic material model with

$$\boldsymbol{\sigma}' = \mathbf{D} : \boldsymbol{\varepsilon}, \tag{5.4}$$

where $\mathbf{D}$ is the stiffness matrix, and the small strain tensor in terms of deformation gradient $\mathbf{F}$ is

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\mathbf{F} + \mathbf{F}^{\mathsf{T}}\right) - \mathbf{I}. \tag{5.5}$$

For the evolution of the pore pressures, we need the storage equation derived from the conservation of mixture mass, written as

$$\frac{\partial p}{\partial t} = \frac{K_\mathrm{w}}{n}[(1-n)\boldsymbol{\nabla}\cdot\mathbf{v} + n\boldsymbol{\nabla}\cdot\mathbf{w}], \tag{5.6}$$

where K_w is the bulk modulus of water, and n is assumed to be spatially constant [77].

To complete the boundary value problem, the initial conditions are:

$$\boldsymbol{\sigma}'(\mathbf{x},t_0) = \boldsymbol{\sigma}'_0 \qquad \text{and} \qquad p(\mathbf{x},t_0) = p_0 \tag{5.7}$$

$$\mathbf{v}(\mathbf{x},t_0) = \mathbf{v}_0 \qquad \text{and} \qquad \mathbf{w}(\mathbf{x},t_0) = \mathbf{w}_0. \tag{5.8}$$

The Dirichlet boundary conditions are

$$\mathbf{v}(\mathbf{x},t) = \bar{\mathbf{v}}(t) \qquad \text{on } \partial\Omega_\mathbf{u} \tag{5.9}$$

$$\mathbf{w}(\mathbf{x},t) = \bar{\mathbf{w}}(t) \qquad \text{on } \partial\Omega_\mathbf{w}, \tag{5.10}$$

and the Neumann boundary conditions are

$$\boldsymbol{\sigma}(\mathbf{x},t)\cdot\mathbf{n} = \bar{\boldsymbol{\tau}}(t) \qquad \text{on } \partial\Omega_{\boldsymbol{\tau}} \tag{5.11}$$

$$p(\mathbf{x},t)\cdot\mathbf{n} = \bar{\mathbf{p}}(t) \qquad \text{on } \partial\Omega_p. \tag{5.12}$$

5.1.1 Weak form of the governing equations

The governing equations in strong form (5.1) and (5.2), need to be transformed into weak form. The procedure is similar to the single-phase MPM presented in Chapter 4, and covered in more detail in Section 4.2. First, we define a space of test functions so that

$$\mathcal{V}_0 = \{\boldsymbol{\eta}_h : \Omega \to \mathbb{R}^3 \quad | \quad \boldsymbol{\eta}_h \in H^1, \boldsymbol{\eta}_h = \mathbf{0} \,\text{on}\, \Gamma_\mathrm{v}\}, \tag{5.13}$$

where H^1 is the Sobolev space of degree 1. Both primary variables $\mathbf{v}$ and $\mathbf{w}$ belong to this space, which means that the velocity fields of both phases are discretised by the same linear (degree 1) shape functions. The weak solution is obtained by multiplying Equations (5.1) and (5.2) with the test functions $\boldsymbol{\eta}_h$ and integrating over the domain Ω. This gives

$$\int_\Omega \boldsymbol{\eta}_h \cdot \rho_\mathrm{w}\mathbf{a}_{\mathrm{w},\,h}\,\mathrm{d}\Omega = \int_\Omega \boldsymbol{\eta}_h\cdot\boldsymbol{\nabla}p_h\,\mathrm{d}\Omega - \int_\Omega \boldsymbol{\eta}_h\cdot\frac{\rho_\mathrm{w}g}{k}n(\mathbf{w}_h-\mathbf{v}_h)\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h\cdot\rho_\mathrm{w}\mathbf{b}\,\mathrm{d}\Omega \tag{5.14}$$

$$\int_\Omega \boldsymbol{\eta}_h \cdot n\rho_\mathrm{w}\mathbf{a}_{\mathrm{w},\,h}\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \cdot (1-n)\rho_\mathrm{s}\mathbf{a}_{\mathrm{s},\,h}\,\mathrm{d}\Omega = \int_\Omega \boldsymbol{\eta}_h \cdot (\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}_h)\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \cdot \rho\mathbf{b}\,\mathrm{d}\Omega. \quad (5.15)$$

Applying the divergence theorem on the pore pressure gradient term gives

$$\int_\Omega \boldsymbol{\eta}_h \cdot \boldsymbol{\nabla} p_h\,\mathrm{d}\Omega = \int_{\partial\Omega} \boldsymbol{\eta}_h \bar{p}\,\mathrm{d}S - \int_\Omega p_h\big(\boldsymbol{\nabla}\cdot\boldsymbol{\eta}_h\big)\,\mathrm{d}\Omega, \quad (5.16)$$

where $\bar{p} = p_h\mathbf{n}$ is the prescribed pore pressure at the boundary of $\partial\Omega_p$ with outward surface normal $\mathbf{n}$. Similarly, the divergence theorem is applied to the total stress term as in Equation (4.3), which gives

$$\int_\Omega \boldsymbol{\eta}_h \cdot (\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}_h)\,\mathrm{d}\Omega = \int_{\partial\Omega} \boldsymbol{\eta}_h \cdot \bar{\boldsymbol{\tau}}\,\mathrm{d}S - \int_\Omega \boldsymbol{\nabla}\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h\,\mathrm{d}\Omega, \quad (5.17)$$

where the prescribed traction at the boundary is $\bar{\boldsymbol{\tau}} = \boldsymbol{\sigma}_h : \mathbf{n}$.

Reinserting Equations (5.16) and (5.17) into Equations (5.14) and (5.15) and reorganising terms gives the weak form for the conservation of momentum for the water as

$$\int_\Omega \boldsymbol{\eta}_h \cdot \rho_\mathrm{w}\mathbf{a}_{\mathrm{w},\,h}\,\mathrm{d}\Omega = \int_{\partial\Omega} \boldsymbol{\eta}_h \bar{p}\,\mathrm{d}S - \int_\Omega p_h\big(\boldsymbol{\nabla}\cdot\boldsymbol{\eta}_h\big)\,\mathrm{d}\Omega - \int_\Omega \boldsymbol{\eta}_h \cdot \frac{\rho_\mathrm{w} g}{k} n(\mathbf{w}_h - \mathbf{v}_h)\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \cdot \rho_\mathrm{w}\mathbf{b}\,\mathrm{d}\Omega, \quad (5.18)$$

and for the conservation of momentum for the mixture as

$$\int_\Omega \boldsymbol{\eta}_h \cdot (1-n)\rho_\mathrm{s}\mathbf{a}_{\mathrm{s},\,h}\,\mathrm{d}\Omega = -\int_\Omega \boldsymbol{\eta}_h \cdot n\rho_\mathrm{w}\mathbf{a}_{\mathrm{w},\,h}\,\mathrm{d}\Omega + \int_{\partial\Omega} \boldsymbol{\eta}_h \cdot \bar{\boldsymbol{\tau}}\,\mathrm{d}S - \int_\Omega \boldsymbol{\nabla}\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h\,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \cdot \rho\mathbf{b}\,\mathrm{d}\Omega. \quad (5.19)$$

5.1.2 Application of initial and boundary conditions

As discussed in Section 4.4 for the single-phase MPM, the boundary conditions can be applied either to the grid or directly on the particles. The water phase boundary conditions are applied in the same way. In other words, the prescribed pore water pressure boundary is applied to the boundary particles, while zero kinematic boundary conditions, which equals impermeable boundaries with zero flux, are applied to the grid. Prescribed water velocity is not included in this implementation and is therefore 0 by default, but can be applied to particles during creation procedure.

5.2 Spatial discretisation

The **v**-**w** formulation presented here is a one-point formulation. Similar to the single-phase MPM described in Chapter 4, the domain Ω^0 is divided into one set of subdomains Ω_p^0 with N_p subdomains. Each point represents the saturated mixture, and the concentrated particle mass M_p is therefore

$$M_p = \rho V_p, \tag{5.20}$$

where $\rho = n\rho_\mathrm{w} + (1-n)\rho_\mathrm{s}$ is the saturated density of the particle. Particle masses for the water and solid phases is defined as

$$M_\mathrm{w} = \rho_\mathrm{w} V_p \qquad \text{and} \qquad M_\mathrm{s} = \rho_\mathrm{s} V_p, \tag{5.21}$$

respectively. Note that these masses are not the actual phase masses of the particle, as they have not been scaled to their respective volume fraction.

Doing the same discretisation procedure as in Section 4.3 on the conservation equations of momentum (5.18) and (5.19), we get

$$\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w}\left[\boldsymbol{\eta}_h \cdot \mathbf{a}_{\mathrm{w},\,h}\right] = \int_{\partial\Omega} \boldsymbol{\eta}_h \bar{p}\mathbf{I}\,\mathrm{d}S - \int_\Omega p_h\left(\boldsymbol{\nabla}\cdot\boldsymbol{\eta}_h\right)\mathrm{d}\Omega + \sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w}\boldsymbol{\eta}_h\cdot\mathbf{b} - \sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w}\boldsymbol{\eta}_h\cdot\frac{g}{k}n(\mathbf{w}_h - \mathbf{v}_h) \tag{5.22}$$

for the water phase, and

$$\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{s}(1-n)\left[\boldsymbol{\eta}_h \cdot \mathbf{a}_{\mathrm{s},\,h}\right] = -\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w} n\left[\boldsymbol{\eta}_h \cdot \mathbf{a}_{\mathrm{w},\,h}\right] + \sum_{p=1}^{N_\mathrm{p}} M_p\boldsymbol{\eta}_h\cdot\mathbf{b} + \int_{\partial\Omega}\boldsymbol{\eta}_h\cdot\bar{\boldsymbol{\tau}}\,\mathrm{d}S - \int_\Omega \boldsymbol{\nabla}\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h\,\mathrm{d}\Omega \tag{5.23}$$

for the mixture. Then the approximate field variables $\boldsymbol{\eta}_h$, $\mathbf{a}_{\mathrm{w},\,h}$, $\mathbf{a}_{\mathrm{s},\,h}$, $\mathbf{v}_h$, and $\mathbf{w}_h$ are replaced by their discretised counterparts, here represented by the generic variable $\mathbf{X}_h$:

$$\mathbf{X}_h = \sum_{i=1}^{N_\mathrm{n}} \mathbf{X}_i N_i(\mathbf{x}). \tag{5.24}$$

Now, we can eliminate the discretised test functions $\boldsymbol{\eta}_i(t)$:

$$\overbrace{\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w} \sum_{i=1}^{N_\mathrm{n}} N_{ip} \sum_{j=1}^{N_\mathrm{n}} N_{jp}}^{m_{\mathrm{w},\,j}} \mathbf{a}_{\mathrm{w},\,j} = \overbrace{\int_{\partial\Omega} \sum_{i=1}^{N_\mathrm{n}} N_{ip} \bar{p} \mathbf{I} \,\mathrm{d}S + \sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w} \sum_{i=1}^{N_\mathrm{n}} N_{ip} \cdot \mathbf{b}}^{\mathbf{F}_{\mathrm{ext,\,w}}} - \underbrace{\left(\boldsymbol{\nabla} \cdot \sum_{i=1}^{N_\mathrm{n}} N_{ip} \right) p V_p}_{\mathbf{F}_{\mathrm{int,w}}} - \underbrace{\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w} \sum_{i=1}^{N_\mathrm{n}} N_{ip} \cdot \frac{g}{k} n \left(\sum_{i=1}^{N_\mathrm{n}} \mathbf{w}_i N_{ip} - \sum_{i=1}^{N_\mathrm{n}} \mathbf{v}_i N_{ip} \right)}_{\mathbf{F}_{\mathrm{drag}}} \tag{5.25}$$

$$\overbrace{\sum_{p=1}^{N_\mathrm{p}} M_\mathrm{s}(1-n) \sum_{i=1}^{N_\mathrm{n}} N_{ip} \sum_{j=1}^{N_\mathrm{n}} N_{jp}}^{m_{\mathrm{s},\,j}} \mathbf{a}_{\mathrm{s},\,j} = - \sum_{p=1}^{N_\mathrm{p}} M_\mathrm{w} n \sum_{i=1}^{N_\mathrm{n}} N_{ip} \sum_{i=1}^{N_\mathrm{n}} N_{jp} \mathbf{a}_{\mathrm{w},\,j} + \underbrace{\sum_{p=1}^{N_\mathrm{p}} M_p \sum_{i=1}^{N_\mathrm{n}} N_{ip} \cdot \mathbf{b} + \int_{\partial\Omega} \sum_{i=1}^{N_\mathrm{n}} N_{ip} \cdot \bar{\boldsymbol{\tau}} \,\mathrm{d}S}_{\mathbf{F}_{\mathrm{ext}}} - \underbrace{\boldsymbol{\nabla} \sum_{i=1}^{N_\mathrm{n}} N_{ip} : \boldsymbol{\sigma} V_p}_{\mathbf{F}_{\mathrm{int}}} . \tag{5.26}$$

Finally, introducing the lumped phase mass matrices

$$\mathbf{m}_\alpha = \sum_{p=1}^{N_\mathrm{p}} M_\alpha N_i N_j \tag{5.27}$$

$$\bar{\mathbf{m}}_\alpha = \sum_{p=1}^{N_\mathrm{p}} n_\alpha M_\alpha N_i N_j \tag{5.28}$$

where α denotes the phase, we get the spatially discretised system

$$\begin{aligned} \mathbf{m}_\mathrm{w} \mathbf{a}_\mathrm{w} &= \mathbf{F}_{\mathrm{ext,w}} - \mathbf{F}_{\mathrm{int,w}} - \mathbf{F}_{\mathrm{drag}} \\ \bar{\mathbf{m}}_\mathrm{s} \mathbf{a}_\mathrm{s} &= -\bar{\mathbf{m}}_\mathrm{w} \mathbf{a}_\mathrm{w} + \mathbf{F}_{\mathrm{ext}} - \mathbf{F}_{\mathrm{int}}. \end{aligned} \tag{5.29}$$

Note that $\mathbf{F}_{\mathrm{int}}$ is the internal force of the total stress $\boldsymbol{\sigma}$.

5.3 Time integration

The spatial discretisation and the initial and boundary conditions give the following system of ordinary differential equations:

$$\mathbf{m}_\mathrm{w} \frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} = \mathbf{F}_{\mathrm{ext,w}} - \mathbf{F}_{\mathrm{int,w}} - \mathbf{F}_{\mathrm{drag}} \tag{5.30}$$

$$\bar{\mathbf{m}}_\mathrm{s} \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = -\bar{\mathbf{m}}_\mathrm{w} \mathbf{a}_\mathrm{w} + \mathbf{F}_{\mathrm{ext}} - \mathbf{F}_{\mathrm{int}}, \tag{5.31}$$

where $\mathbf{m}_\mathrm{w}$, $\bar{\mathbf{m}}_\mathrm{w}$ and $\bar{\mathbf{m}}_\mathrm{s}$ are mass matrices for water and solid phase, as defined in Equation (5.29); $\mathrm{d}\mathbf{w}/\mathrm{d}t = \mathbf{a}_\mathrm{w}$ is the acceleration of the water phase, and $\mathrm{d}\mathbf{v}/\mathrm{d}t = \mathbf{a}_\mathrm{s}$ is the acceleration of the solid phase.

To obtain the velocities $\mathbf{v}$ and $\mathbf{w}$, the above equations need to be integrated over the timestep from time t to time $t + \Delta t$, as follows

$$\int_t^{t+\Delta t} \mathbf{m}_\mathrm{w} \frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t}\,\mathrm{d}t = \int_t^{t+\Delta t} \left(\mathbf{F}_{\mathrm{ext,w}} - \mathbf{F}_{\mathrm{int,w}} - \mathbf{F}_{\mathrm{drag}}\right)\mathrm{d}t \tag{5.32}$$

$$\int_t^{t+\Delta t} \bar{\mathbf{m}}_\mathrm{s} \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\,\mathrm{d}t = \int_t^{t+\Delta t} \left(-\bar{\mathbf{m}}_\mathrm{w}\mathbf{a}_\mathrm{w} + \mathbf{F}_{\mathrm{ext}} - \mathbf{F}_{\mathrm{int}}\right)\mathrm{d}t. \tag{5.33}$$

Reordering terms and applying the simple Euler forward time integration from Equation (4.27), leads to

$$\mathbf{m}_\mathrm{w}\left(\mathbf{w}^{n+1} - \mathbf{w}^n\right) \approx \left(\mathbf{F}^n_{\mathrm{ext,w}} - \mathbf{F}^n_{\mathrm{int,w}} - \mathbf{F}^n_{\mathrm{drag}}\right)\Delta t \tag{5.34}$$

$$\bar{\mathbf{m}}_\mathrm{s}\left(\mathbf{v}^{n+1} - \mathbf{v}^n\right) \approx \left(-\bar{\mathbf{m}}_\mathrm{w}\mathbf{a}^n_\mathrm{w} + \mathbf{F}^n_{\mathrm{ext}} - \mathbf{F}^n_{\mathrm{int}}\right)\Delta t, \tag{5.35}$$

where superscripts n and $n+1$ denote time steps at times t and $t + \Delta t$, respectively.

Rewriting Equations (5.34) and (5.35) to solve for the velocities at time step $n+1$, gives

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \mathbf{m}_\mathrm{w}^{-1}\left(\mathbf{F}^n_{\mathrm{ext,w}} - \mathbf{F}^n_{\mathrm{int,\,w}} - \mathbf{F}^n_{\mathrm{drag}}\right)\Delta t \tag{5.36}$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \bar{\mathbf{m}}_\mathrm{s}^{-1}\left(-\bar{\mathbf{m}}_\mathrm{w}\mathbf{a}^n_\mathrm{w} + \mathbf{F}^n_{\mathrm{ext}} - \mathbf{F}^n_{\mathrm{int}}\right)\Delta t. \tag{5.37}$$

The mass matrices $\mathbf{m}_\mathrm{w}$ and $\bar{\mathbf{m}}_\mathrm{s}$ need to be inverted. However, as they are diagonally lumped as the total mass matrix defined in Equation (4.23), the system of equations is decoupled. Consequently, Equations (5.36) and (5.37) are solved iteratively and separately at each node by dividing the force residuals in the parentheses with the scalar nodal phase mass.

The discretised governing equations Equations (5.30) and (5.31) are solved at nodes for the accelerations. The nodal accelerations are used to update both the nodal velocities and the particle velocities to avoid the so-called small mass problem, which is discussed in more depth in Section 4.5. Let us therefore define nodal accelerations from Equations (5.36) and (5.37) as

$$\mathbf{a}^n_\mathrm{w} = \mathbf{m}_\mathrm{w}^{-1}\left(\mathbf{F}^n_{\mathrm{ext,w}} - \mathbf{F}^n_{\mathrm{int,w}} - \mathbf{F}^n_{\mathrm{drag}}\right) \tag{5.38}$$

$$\mathbf{a}^n_\mathrm{s} = \bar{\mathbf{m}}_\mathrm{s}^{-1}\left(-\bar{\mathbf{m}}_\mathrm{w}\mathbf{a}^n_\mathrm{w} + \mathbf{F}^n_{\mathrm{ext}} - \mathbf{F}^n_{\mathrm{int}}\right). \tag{5.39}$$

By use of the interpolation functions in the same manner as for the single-phase MPM of Section 4.5, the particle velocities are updated directly from the nodal accelerations through

$$\mathbf{w}_p^{n+1} = \mathbf{w}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_\mathrm{w}^n N_{ip}^n, \tag{5.40}$$

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_\mathrm{s}^n N_{ip}^n. \tag{5.41}$$

Then the particle displacement is updated from the particle solid velocity:

$$\mathbf{u}_p^{n+1} = \Delta t \mathbf{v}_p^{n+1}. \tag{5.42}$$

Note that only the solid velocity $\mathbf{v}_p$ influences the displacement $\mathbf{u}_p$. The particle strain increment can, for instance, be found from the particle velocity gradient

$$\mathbf{L}_p^{n+1} = \boldsymbol{\nabla} \mathbf{v}_p^*, \tag{5.43}$$

where $\mathbf{v}_p^*$ is the particle solid velocity interpolated from the nodal updated velocity $\mathbf{v}^{n+1}$. The constitutive model gives the effective stress response of the particle based on the strain increment.

The newly computed phase velocities are used to compute the new pore pressure p^{n+1} at the particle, which is also integrated in time with a simple forward Euler step:

$$p^{n+1} \approx p^n + \frac{\mathrm{d}p}{\mathrm{d}t} \Delta t, \tag{5.44}$$

where porepressure rate $\mathrm{d}p/\mathrm{d}t$ is found from the storage equation defined in Equation (3.37). This gives

$$p^{n+1} = p^n + \frac{K_\mathrm{w}}{n} \left[(1-n) \boldsymbol{\nabla} \cdot \mathbf{v}^{n+1} + n \boldsymbol{\nabla} \cdot \mathbf{w}^{n+1} \right] \Delta t. \tag{5.45}$$

Note that when the phase accelerations are almost equal, the phase velocities will be too. Consequently, the water phase has similar displacements as the solid phase. However, when this is not true, then the pore pressure is computed with velocity values sampled at different locations. Therefore, the **v**-**w** formulation does not guarantee that fluid mass is conserved [8].

Critical time step. There will be pore pressure oscillations because of the type of mathematical problem that poromechanics produces. These oscillations will worsen with smaller time steps. The abrupt change in load at the draining boundary disturbs the system.

Vermeer & Verruijt [79] suggested a lower limit for the time step in such cases, written as

$$\Delta t_{\mathrm{w}} \geq \frac{1}{6} \frac{(\Delta h)^2}{\theta c_v}, \tag{5.46}$$

where Δh is the cell size closest to the draining boundary, $0 \geq \theta \geq 1$ determines type of time integration, and c_v is the coefficient of consolidation.

5.4 Outline of one time step

Repeating here the anatomy of an explicit time step as performed in the Uintah framework [29], first presented in Section 4.5.2. Equations are added in order to transform an existing explicit MPM code into a code that can model pore pressure generation and dissipation. Refer to Figure 4.1 for an illustration of the phases described below.

1. Mapping phase

Particle quantities are mapped from particles to nodes.

1) Apply external loads on boundary particles

2) Map particle masses to nodes:

$$m_i^n = \sum_{p=1}^{N_{\mathrm{p}}} M_p N_{ip}^n \tag{5.47}$$

$$m_{\mathrm{w},i}^n = \sum_{p=1}^{N_{\mathrm{p}}} M_{p,\mathrm{w}} N_{ip}^n \tag{5.48}$$

3) Compute particle momentums:

$$\mathbf{p}_p^n = (1-n) M_p \mathbf{v}_p^n \tag{5.49}$$

$$\mathbf{p}_{p,\mathrm{w}}^n = n M_{p,\mathrm{w}} \mathbf{w}_p^n \tag{5.50}$$

4) Compute nodal phase velocities from Equations (5.49) and (5.50):

$$\mathbf{v}_i^n = \frac{\sum_{p=1}^{N_\mathrm{p}} \mathbf{p}_p^n N_{ip}^n}{m_i^n} \tag{5.51}$$

$$\mathbf{w}_i^n = \frac{\sum_{p=1}^{N_\mathrm{p}} \mathbf{p}_{p,\mathrm{w}}^n N_{ip}^n}{m_{\mathrm{w},i}^n} \tag{5.52}$$

5) Map particle external loads to node:

$$\mathbf{F}_{\mathrm{ext}}^n = \sum_{p=1}^{N_\mathrm{p}} \mathbf{F}_{p,\,\mathrm{ext}}^n N_{ip}^n \tag{5.53}$$

6) Compute nodal internal force from particle total stress:

$$\mathbf{F}_{\mathrm{int}}^n = \sum_{p=1}^{N_\mathrm{p}} V_p \nabla N_{ip}^n : \left(\boldsymbol{\sigma}'^n + p^n \mathbf{I}\right) \tag{5.54}$$

And compute nodal internal fluid force and nodal fluid drag force:

$$\mathbf{F}_{\mathrm{int,w}}^n = \sum_{p=1}^{N_\mathrm{p}} V_p \nabla N_{ip}^n : p^n \mathbf{I} \tag{5.55}$$

$$\mathbf{F}_{\mathrm{drag}}^n = \left(\mathbf{w}_i^n - \mathbf{v}_i^n\right) \sum_{p=1}^{N_\mathrm{p}} N_{ip} \frac{ng}{k} M_{p,\mathrm{w}} \tag{5.56}$$

Note that the node subscript i is omitted for simplicity in the notation of the internal, external and drag nodal forces.

2. Lagrangian phase

Momentum equations (5.38) and (5.39) are solved for nodal accelerations $\mathbf{a}_i^n$. First, we solve for fluid acceleration, as it is used to compute the solid acceleration. Furthermore, the following is done:

1) Compute particle velocities from nodal acceleration.

$$\mathbf{w}_p^{n+1} = \mathbf{w}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_\mathrm{w}^n N_{ip}^n \tag{5.57}$$

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \Delta t \sum_{i=1}^{N_\mathrm{n}} \mathbf{a}_\mathrm{s}^n N_{ip}^n \tag{5.58}$$

2) Compute nodal velocities

$$\mathbf{w}_i^{n+1} = \mathbf{w}_i^n + \mathbf{a}_{\mathrm{w}}^n \Delta t \tag{5.59}$$

$$\mathbf{v}_i^{n+1} = \mathbf{v}_i^n + \mathbf{a}_{\mathrm{s}}^n \Delta t \tag{5.60}$$

3) Compute particle solid velocity gradients

$$\mathbf{L}_p^{n+1} = \nabla \mathbf{v}_p^* = \nabla \left(\sum_{i=1}^{N_{\mathrm{n}}} \mathbf{v}_i^{n+1} N_{ip}^n \right) \tag{5.61}$$

4) Update particle volume

5) Apply the constitutive model on particles to get $\boldsymbol{\sigma}_p^{n+1}$

6) Update pore pressure from Equation (5.45).

3. Convective phase

Particles are moved and grid properties are deleted. The particle displacements and positions are updated from the nodal velocities:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \sum_{i=1}^{N_{\mathrm{n}}} \mathbf{v}_i^{n+1} N_{ip}^n \tag{5.62}$$

It matters little who first arrives at an idea,
rather what is significant is how far that idea can go

Sophie Germain

CHAPTER 6

ITERATIVELY COUPLED IMPLICIT MPM-FVM: DERIVATION

This chapter contains the main contribution of this thesis. It describes an implicit, dynamic, one-point MPM iteratively coupled with FVM displacement-pressure formulation for saturated soils. The governing equations are discretised in space and time. A preconditioner is applied to obtain a split system of equations, which is justified both mathematically and physically. The split coupled problem is solved iteratively by MPM for the displacements and FVM for the pore pressures. The implementation of the method is described in Chapter 7 and verified in Chapter 8.

Chapter outline

6.1 Governing equations

The formulation adopted here is a dynamic **u**-p formulation, which was introduced and derived in Section 3.3. The governing equations are repeated here for convenience. The acceleration of either phase is denoted as **a** in this chapter because the **u**-p formulation assumes no relative acceleration of water and solid phases. First, we need the conservation of momentum for the mixture:

$$\nabla \cdot \boldsymbol{\sigma}' + \nabla p + \rho \mathbf{b} - \rho \mathbf{a} = 0, \tag{6.1}$$

where $\boldsymbol{\sigma}'$ is the effective stress, p is the pore pressures, ρ is the saturated density, **b** is the body force, and **a** the acceleration. Then, we need the mass conservation equation with fluid flow or storage equation with $\dot{\varepsilon}_{\mathrm{v}} = \nabla \cdot \mathbf{v}$[1]:

$$\frac{n}{K_{\mathrm{w}}} \dot{p} - \dot{\varepsilon}_{\mathrm{v}} - \nabla \cdot \mathbf{q} = 0 \tag{6.2}$$

The effective stress $\boldsymbol{\sigma}'$ is computed from a constitutive soil model of choice, as

$$\boldsymbol{\sigma}' = \mathscr{G}(\mathbf{F}), \tag{6.3}$$

where **F** is the deformation gradient. Furthermore, **q** is the specific discharge defined by

$$\mathbf{q} = -\frac{k}{\rho_{\mathrm{w}} g} \left[\nabla p + \rho_{\mathrm{w}} (\mathbf{a} - \mathbf{b}) \right], \tag{6.4}$$

where k is the hydraulic conductivity.

The initial conditions are given as

$$\boldsymbol{\sigma}'(\mathbf{x}, t_0) = \boldsymbol{\sigma}'_0 \qquad \text{and} \qquad p(\mathbf{x}, t_0) = p_0, \tag{6.5}$$

and the Dirichlet boundary conditions are

$$\mathbf{u}(\mathbf{x}, t) = \bar{\mathbf{u}}(t) \qquad \text{on } \partial\Omega_{\mathbf{u}} \tag{6.6}$$

$$p(\mathbf{x}, t) = \bar{p}(t) \qquad \text{on } \partial\Omega_{p}. \tag{6.7}$$

and the Neumann boundary conditions are

$$\boldsymbol{\sigma}(\mathbf{x}, t) \cdot \mathbf{n} = \bar{\boldsymbol{\tau}}(t) \qquad \text{on } \partial\Omega_{\boldsymbol{\tau}} \tag{6.8}$$

$$\mathbf{q}(\mathbf{x}, t) \cdot \mathbf{n} = \bar{\mathbf{q}}(t) \qquad \text{on } \partial\Omega_{\mathbf{q}}, \tag{6.9}$$

[1] This relation holds for large deformations too.

6.2 Weak form of the governing equations

As mentioned in Chapter 4, the spatial discretisation process of MPM and FEM are similar because the conservation equations are solved at the nodes of the background mesh. Therefore, discretisation techniques where the equations are split between FEM and FVM are also applicable to MPM. In this section, the conservation of momentum equation (6.1) is discretised at nodes, which is common for FEM and single-phase MPM (see Chapter 4), while the conservation of mass equation (6.2) is discretised at cell centers as with FVM. Similar to FEM and MPM, FVM also operates on a weak formulation of the governing equations. While a common approach with FEM and MPM is to multiply the governing equations with a test function and then integrate over the entire domain Ω, FVM is derived by a volume integral over the individual element only. However, FVM may also be derived in the same manner as FEM and MPM, if the test function is piecewise constant [86, 88, 103, 104]. The transformation process from strong form to weak form is described in more depth in Section 4.2 for the single-phase MPM.

First, let us define the solution function spaces for the field variables $\mathbf{u}$ and p and their essential boundary conditions as

$$\mathscr{U} = \{\mathbf{u} : \Omega \to \mathbb{R}^3 \quad | \quad \mathbf{u} \in H^1, \quad \mathbf{u} = \bar{\mathbf{u}} \text{ on } \partial\Omega_{\mathbf{u}}\} \tag{6.10}$$

$$\mathscr{P} = \{p : \Omega \to \mathbb{R} \quad | \quad p \in L^2, \quad p = \bar{p} \text{ on} \partial\Omega_p\}, \tag{6.11}$$

where H^1 is the Sobolev space of degree 1 and L^2 is the space of square integrable functions. This means that the displacement field $\mathbf{u}$ is discretised by linear (degree 1) shape functions, and the pore pressure field p by constant (degree 0) shape functions. The finite-dimensional approximated spaces $\mathscr{U}_h$ and $\mathscr{P}_h$ are the corresponding subspaces of the infinite-dimensional solution spaces $\mathscr{U}$ and $\mathscr{P}$. We are seeking the solution of a discrete approximation of these spaces by finding $(\mathbf{u}_h, p_h) \in \mathscr{U}_h \times \mathscr{P}_h$.

The spaces of the corresponding test functions $\boldsymbol{\eta}$ and φ are

$$\mathscr{U}_0 = \{\boldsymbol{\eta} : \Omega \to \mathbb{R}^3 \quad | \quad \boldsymbol{\eta} \in H^1, \quad \boldsymbol{\eta} = \mathbf{0} \text{ on } \partial\Omega_{\mathbf{u}}\} \tag{6.12}$$

$$\mathscr{P}_0 = \{\varphi : \Omega \to \mathbb{R} \quad | \quad \varphi \in L^2, \quad \varphi = 0 \text{ on } \partial\Omega_p\}, \tag{6.13}$$

The approximate test functions $\boldsymbol{\eta}_h$ and φ_h are defined on the corresponding subspaces $\mathscr{U}_{0,h}$ and $\mathscr{P}_{0,h}$ and use the same shape functions as defined in Equation (6.22). The weak solution of the problem is obtained by multiplying Equations (6.1) and (6.2) with the corresponding test functions $\{\boldsymbol{\eta}_h, \varphi_h\}$ and integrating over the domain Ω.

6.2.1 Weak form of the fluid mass equation

Multiplying Equation (6.2) with the test function φ_h over the domain Ω, gives

$$\int_\Omega \varphi_h \frac{n}{K_\mathrm{w}} \dot{p}_h \,\mathrm{d}\Omega - \int_\Omega \varphi_h \dot{\varepsilon}_{v,h} \,\mathrm{d}\Omega - \int_\Omega \varphi_h \boldsymbol{\nabla} \cdot \mathbf{q}_h \,\mathrm{d}\Omega = 0. \tag{6.14}$$

With φ_h being piecewise constant, meaning that

$$\varphi_{h,n}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ in element } n \\ 0 & \text{otherwise,} \end{cases} \tag{6.15}$$

Equation (6.14) can be interpreted as FVM with the conservation of mass being imposed on an element-by-element basis. Let us separate the terms of the generalised Darcy's law (6.4) and write the third term of Equation (6.14) as an integral over element volume Ω_n:

$$\int_{\Omega_n} \boldsymbol{\nabla} \cdot \mathbf{q}_h \,\mathrm{d}\Omega_n = \int_{\Omega_n} \boldsymbol{\nabla} \cdot \hat{\mathbf{q}}_h \,\mathrm{d}\Omega_n + \int_{\Omega_n} \boldsymbol{\nabla} \cdot \frac{k}{g} \mathbf{b} \,\mathrm{d}\Omega_n - \int_{\Omega_n} \boldsymbol{\nabla} \cdot \frac{k}{g} \mathbf{a}_h \,\mathrm{d}\Omega_n, \tag{6.16}$$

where the pore pressure gradient term is defined as

$$\hat{\mathbf{q}} \equiv -\frac{k}{(\rho_\mathrm{w} g)} \boldsymbol{\nabla} p. \tag{6.17}$$

With constant permeability k over the element, we can apply the divergence theorem on the first term on the right hand side of Equation (6.16) which gives:

$$\int_{\Omega_n} \boldsymbol{\nabla} \cdot \hat{\mathbf{q}}_h \,\mathrm{d}\Omega_n = \int_{\partial\Omega_n} \hat{\mathbf{q}}_h \cdot \mathbf{n} \,\mathrm{d}S = \sum_{f \in \mathrm{faces}} \int_{\partial\Omega_f} \hat{\mathbf{q}}_h \cdot \mathbf{n}_f \,\mathrm{d}S, \tag{6.18}$$

where boundary surface normal $\mathbf{n}$ is the outward normal. The same procedure can be applied to the body force term and the inertial term of Equation (6.16). In this thesis, we disregard the boundary effects of open boundaries, which means that the body force term is 0 due to the constant gravitational field. The inertial term is kept as a volume integral to maintain its similarity to the velocity divergence term. Inserting Equations (6.16) and (6.18) into Equation (6.14) leads to the approximate weak form of the fluid mass conservation equation over an element n:

$$\begin{aligned} r_{\mathrm{mass},\,n} = &\int_{\Omega_n} \frac{n}{K_\mathrm{w}} \dot{p}_h \,\mathrm{d}\Omega_n - \int_{\Omega_n} \boldsymbol{\nabla} \cdot \mathbf{v}_h \,\mathrm{d}\Omega_n - \sum_{f \in \mathrm{faces}} \int_{\partial\Omega_f} \hat{\mathbf{q}}_h \cdot \mathbf{n}_f \,\mathrm{d}S \\ &+ \frac{k}{g} \int_{\Omega_n} \boldsymbol{\nabla} \cdot \mathbf{b} \,\mathrm{d}\Omega_n - \frac{k}{g} \int_{\Omega_n} \boldsymbol{\nabla} \cdot \mathbf{a}_h \,\mathrm{d}\Omega_n = 0, \end{aligned} \tag{6.19}$$

where $\boldsymbol{\nabla} \cdot \mathbf{v} = \dot{\varepsilon}_\mathrm{v}$.

6.2.2 Weak form of the momentum equation

The weak problem of Equation (6.1) is similar to the weak problem derived for the single-field MPM in Equation (4.6). The difference is the linear split of the total stress $\boldsymbol{\sigma} = \boldsymbol{\sigma}' + p\mathbf{I}$. Replacing the total stress $\boldsymbol{\sigma}$ with the effective stress $\boldsymbol{\sigma}'$ and the pore pressure $p\mathbf{I}$ from Equation (4.6) gives

$$\begin{aligned}\int_\Omega \nabla\boldsymbol{\eta}_h : \boldsymbol{\sigma}_h \,\mathrm{d}\Omega &= \int_\Omega \nabla\boldsymbol{\eta}_h : \boldsymbol{\sigma}'_h \,\mathrm{d}\Omega + \int_\Omega \nabla\boldsymbol{\eta}_h : p_h\mathbf{I}\,\mathrm{d}\Omega \\ &= \int_\Omega \nabla\boldsymbol{\eta}_h : \boldsymbol{\sigma}'_h \,\mathrm{d}\Omega + \int_\Omega \nabla\cdot\boldsymbol{\eta}_h p_h \,\mathrm{d}\Omega, \end{aligned} \tag{6.20}$$

because $\nabla\boldsymbol{\eta}_h : \mathbf{I} = \mathrm{tr}\left(\nabla\boldsymbol{\eta}_h\right) = \nabla\cdot\boldsymbol{\eta}_h$. Inserting Equation (6.20) into Equation (4.6) then gives the coupled weak form of the momentum equation as:

$$\begin{aligned}\mathbf{r}_{\mathrm{mom}} = &-\int_\Omega \boldsymbol{\eta}_h \rho \mathbf{a}_h \,\mathrm{d}\Omega + \int_{\partial\Omega_\tau} \boldsymbol{\eta}_h \bar{\boldsymbol{\tau}} \,\mathrm{d}S - \int_\Omega \nabla\boldsymbol{\eta}_h : \boldsymbol{\sigma}'_h \,\mathrm{d}\Omega \\ &- \int_\Omega \nabla\cdot\boldsymbol{\eta}_h p_h \,\mathrm{d}\Omega + \int_\Omega \boldsymbol{\eta}_h \rho \mathbf{b} \,\mathrm{d}\Omega = \mathbf{0}. \end{aligned} \tag{6.21}$$

6.3 Spatial discretisation

This section will finish the spatial discretisation process, and we will end up with a system of equations on matrix form similar to FEM, MPM and FVM. The residuals will be considered separately.

We start by partitioning the domain Ω with boundary $\partial\Omega$ into nonoverlapping elements such that $\Omega = \cup_{N_\mathrm{e}}\Omega_e$, with N_e the number of elements in domain and Ω_e the partitioned volume. With finite-dimensional subspaces, an approximate solution of the field is written as

$$\mathbf{u}_h = \sum_{i=1}^{N_\mathrm{n}} \mathbf{u}_i(t) N_i(\mathbf{x}) \tag{6.22}$$

$$p_h = \sum_{n=1}^{N_\mathrm{e}} p_n(t) \varphi_n(\mathbf{x}), \tag{6.23}$$

where N_n is the number of nodes, N_e is the number of elements, φ_n is the piecewise constant shape function of the pressure p which is identical to the test function φ_h (6.15), and N_i is the linear shape function at node i defined in Equation (4.14). The other kinematic variables $\mathbf{v}_h = \dot{\mathbf{u}}_h$ and $\mathbf{a}_h = \ddot{\mathbf{u}}_h$ are discretised as $\mathbf{u}_h$ in Equation (6.22).

6.3.1 Spatial discretisation of the residual of momentum conservation.

Inserting the interpolation functions as defined in Equations (6.22) and (6.23) into the weak problem of Equation (6.21), and discretising the body into particles as in the process described in Section 4.3, we obtain the following residual equation for a node i:

$$\mathbf{r}_{\text{mom},\,i} = -\overbrace{\sum_{p=1}^{N_\text{p}} M_p N_{ip} \left(\sum_{j=1}^{N_\text{n}} \mathbf{a}_j N_{jp} \right)}^{\sum_{j=1}^{N_\text{n}} m_{ij}\mathbf{a}_j} + \overbrace{\sum_{p=1}^{N_\text{p}} M_p N_{ip} \mathbf{b}_p + \hat{\boldsymbol{\tau}}_i}^{\mathbf{F}_{\text{ext},\,i}} \\ \underbrace{- \sum_{p=1}^{N_\text{p}} V_p \nabla N_{ip} : \boldsymbol{\sigma}'}_{\mathbf{F}'_{\text{int},\,i}} \underbrace{- \sum_{n=1}^{N_\text{e}} \nabla \cdot N_{ip} V_p \varphi_n \, p_n}_{\mathbf{Q}_{1,ni}} = \mathbf{0}. \quad (6.24)$$

As for the single-phase MPM, the consistent mass matrix defined here is replaced with a lumped mass matrix. Creating a global system of equations for the residual of momentum conservation, gives

$$\mathbf{r}_\text{mom} = -\mathbf{M}\mathbf{a} - \mathbf{F}_\text{int} - \mathbf{Q}_1\mathbf{p} + \mathbf{F}_\text{ext} = \mathbf{0}, \quad (6.25)$$

where $\mathbf{M}$ is the lumped mass matrix, and $\mathbf{Q}_1$ the coupling matrix. Note that the internal force from effective stress $\mathbf{F}'_\text{int}$ is the same as the internal force from the total stress $\mathbf{F}_\text{int}$ in the single-phase formulation because $\mathbf{F}_\text{int}$ depends on effective stress $\boldsymbol{\sigma}'$ only. The apostrophe is omitted from $\mathbf{F}'_\text{int}$ for simplicity for the remainder of this chapter. Also note that the internal fluid force term $\mathbf{Q}_1\mathbf{p}$ is the only difference in the definition of $\mathbf{r}_\text{mom}$ for coupled (6.25) and single-phase formulation (4.46).

6.3.2 Spatial discretisation of the residual of mass conservation.

The pore pressures are discretised by piecewise constant shape functions, defined in Equation (6.23). Replacing the variables with their discretised counterparts defined in Equations (6.22) and (6.23) into Equation (6.19),

yields

$$r_{\text{mass},\,n} = \overbrace{\frac{k}{g}\int_{\Omega_n}\sum_{i=1}^{N_\text{n}}\boldsymbol{\nabla}\cdot N_{in}\,\mathrm{d}\Omega_n\,\mathbf{a}_i}^{Q_{2,in}} - \overbrace{\int_{\Omega_n}\sum_{i=1}^{N_\text{n}}\boldsymbol{\nabla}\cdot N_{in}\,\mathrm{d}\Omega_n\,\mathbf{v}_i}^{Q_{2,in}} + \underbrace{\int_{\Omega_n}\frac{n}{K_\text{w}}\,\mathrm{d}\Omega_n}_{C_n}\dot{p}_n - \sum_{f\in\text{faces}}\int_{\partial\Omega_f}\hat{\mathbf{q}}_h\cdot\mathbf{n}_f\,\mathrm{d}S + \underbrace{\frac{k}{g}\int_{\Omega_n}\boldsymbol{\nabla}\cdot\mathbf{b}\,\mathrm{d}\Omega_n}_{F_{\text{ext, w, }n}} = 0, \quad (6.26)$$

over element n. Continuity is forced on the boundary between elements by the flux term defined in Equation (6.18). There are several approaches of approximating the flux $\hat{\mathbf{q}}$. The particular choice of flux approximation method does not influence the derivation that follows and is further explained in Section 7.1.5. When assembled, the flux term is written as $\mathbf{Tp}$ where $\mathbf{T}$ is the transmissibility matrix and $\mathbf{p}$ is an n-length vector of all element centre pore pressures p_n.

The assembled global system can then be written as

$$\mathbf{r}_\text{mass} = \frac{k}{g}\mathbf{Q}_2\mathbf{a} - \mathbf{Q}_2\mathbf{v} + \mathbf{C}\dot{\mathbf{p}} - \mathbf{T}\mathbf{p} + \mathbf{F}_\text{ext,w} = \mathbf{0}, \quad (6.27)$$

where $\mathbf{C}$ is the fluid compressibility matrix which is a diagonal matrix and $\mathbf{Q}_2 = \mathbf{Q}_1^\mathsf{T}$ is the coupling matrix.

6.4 Fully implicit system

The two residuals defined in Equations (6.25) and (6.27) must be discretised in time. Displacement variable $\mathbf{u}$ needs to be integrated twice in time, while pore pressure variable $\mathbf{p}$ needs to be integrated once in time. The time differentials are integrated by use of the Newmark-β method [102] for the displacements $\mathbf{u}$ and simple backwards Euler for pore pressures $\mathbf{p}$ [105]:

$$\begin{aligned}
\mathbf{a}^{n+1} &= \frac{1}{\beta\Delta t^2}\left(\mathbf{u}^{n+1} - \mathbf{u}^n\right) - \frac{1}{\beta\Delta t}\mathbf{v}^n - \frac{1-2\beta}{\beta}\mathbf{a}^n \\
\mathbf{v}^{n+1} &= \frac{\gamma}{\beta\Delta t}\left(\mathbf{u}^{n+1} - \mathbf{u}^n\right) + \left(1-\frac{\gamma}{\beta}\right)\mathbf{v}^n + \Delta t\left(1-\frac{\gamma}{2\beta}\right)\mathbf{a}^n \\
\dot{\mathbf{p}}^{n+1} &= \frac{\mathbf{p}^{n+1} - \mathbf{p}^n}{\Delta t},
\end{aligned} \quad (6.28)$$

where $\gamma = 1/2$ and $\beta = 1/4$ corresponds to constant acceleration during the time interval Δt, while $\beta = 1/6$ corresponds to linear acceleration [102]. The equations for acceleration $\mathbf{a}^{n+1}$ and velocity $\mathbf{v}^{n+1}$ can be simplified by noticing that $\mathbf{u}^n = \mathbf{0}$ due to the grid resetting of MPM.

The Newmark-β method with $\gamma = 1/2$ is second-order accurate and is unconditionally stable when $\beta \geq \gamma/2$ for linear structural dynamics, although Hughes [106] proved stability cannot be guaranteed in the nonlinear regime. Furthermore, there is no high-frequency numerical damping with this scheme [107]. The backwards Euler method is also unconditionally stable but is first-order accurate. The backwards Euler method has damping properties, which is beneficial for diffusion equations because the response should be smooth regardless of load [107]. For example, in the case of coupled flow and displacements, an instantaneous applied load will still result in a smoothed pore pressure field.

We develop the Newton-Raphson set-up of the system of equations as in Section 4.6 on single-phase implicit MPM. A Taylor expansion around the fixed-point with respect to the field variables $\mathbf{u}$ and $\mathbf{p}$ at time step $n+1$ gives:

$$\begin{aligned} \mathbf{r}_{\text{mom}}^{k+1} &\approx \mathbf{r}_{\text{mom}}^{k} + \frac{\partial \mathbf{r}_{\text{mom}}}{\partial \mathbf{u}} \mathrm{d}\mathbf{u}^{k+1} + \frac{\partial \mathbf{r}_{\text{mom}}}{\partial \mathbf{p}} \mathrm{d}\mathbf{p}^{k+1} = 0 \\ \mathbf{r}_{\text{mass}}^{k+1} &\approx \mathbf{r}_{\text{mass}}^{k} + \frac{\partial \mathbf{r}_{\text{mass}}}{\partial \mathbf{u}} \mathrm{d}\mathbf{u}^{k+1} + \frac{\partial \mathbf{r}_{\text{mass}}}{\partial \mathbf{p}} \mathrm{d}\mathbf{p}^{k+1} = 0, \end{aligned} \tag{6.29}$$

where superscripts k and $k+1$ denote iterations within the current time step. Because superscripts k and $k+1$ imply that we are at time step $n+1$, the superscript $n+1$ is omitted for clarity.

Differentiating Equation (6.25) with respect to $\mathbf{u}^k$ gives

$$\begin{aligned} \frac{\partial \mathbf{r}_{\text{mom}}}{\partial \mathbf{u}} &= -\mathbf{M} \frac{\partial \mathbf{a}^k}{\partial \mathbf{u}^k} - \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{u}} \\ &= -\left(\frac{1}{\beta \Delta t^2} \mathbf{M} + \mathbf{K}^k \right) = -\bar{\mathbf{K}}^k, \end{aligned} \tag{6.30}$$

where $\partial \mathbf{F}_{\text{int}} / \partial \mathbf{u} = \mathbf{K} = \mathbf{K}_{\text{mat}} + \mathbf{K}_{\text{geo}}$ is the nonlinear stiffness matrix defined in Equation (4.51) in Section 4.6. Differentiating Equation (6.25) with respect to $\mathbf{p}^k$ gives

$$\frac{\partial \mathbf{r}_{\text{mom}}}{\partial \mathbf{p}} = -\mathbf{Q}_1. \tag{6.31}$$

Introducing the direct time integration from Equation (6.28) into Equation (6.27) gives

$$\mathbf{r}_{\text{mass}}^{k} = \mathbf{Q}_2 \left[\frac{k}{g} \left(\frac{1}{\beta \Delta t^2} \mathbf{u}^k - \frac{1}{\beta \Delta t} \mathbf{v}^n - \frac{1-2\beta}{\beta} \mathbf{a}^n \right) - \left(\frac{\gamma}{\beta \Delta t} \mathbf{u}^k + \left[1 - \frac{\gamma}{\beta} \right] \mathbf{v}^n + \Delta t \left[1 - \frac{\gamma}{2\beta} \right] \mathbf{a}^n \right) \right] + \mathbf{C} \frac{1}{\Delta t} \left(\mathbf{p}^k - \mathbf{p}^n \right) - \mathbf{T} \mathbf{p}^k + \mathbf{F}_{\text{ext, w}}^{k} = \mathbf{0}. \quad (6.32)$$

Consquently, the partial derivatives of $\mathbf{r}_{\text{mass}}$ with respect to the field variables $\mathbf{u}^k$ and $\mathbf{p}^k$ become

$$\frac{\partial \mathbf{r}_{\text{mass}}}{\partial \mathbf{u}} = -\frac{1}{\Delta t} \mathbf{Q}_2 \left[\frac{\gamma}{\beta} - \frac{k}{g} \frac{1}{\beta \Delta t} \right] = -\frac{1}{\Delta t} \bar{\mathbf{Q}}_2 \quad (6.33)$$

$$\frac{\partial \mathbf{r}_{\text{mass}}}{\partial \mathbf{p}} = -\frac{1}{\Delta t} (\Delta t \mathbf{T} - \mathbf{C}). \quad (6.34)$$

Multiplying Equation (6.29) with -1 and multiplying $\mathbf{r}_{\text{mass}}^{k+1}$ with Δt leads to a system in block matrix form defined as

$$\mathbf{A}^k \, \mathrm{d}\mathbf{x}^k = -\mathbf{r}^k, \quad (6.35)$$

where

$$\mathbf{A}^k = \begin{bmatrix} -\partial \mathbf{r}_{\text{mom}} / \partial \mathbf{u} & -\partial \mathbf{r}_{\text{mom}} / \partial \mathbf{p} \\ -\Delta t \partial \mathbf{r}_{\text{mass}} / \partial \mathbf{u} & -\Delta t \partial \mathbf{r}_{\text{mass}} / \partial \mathbf{p} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{K}}^k & \mathbf{Q}_1 \\ \bar{\mathbf{Q}}_2 & \Delta t \mathbf{T} - \mathbf{C} \end{bmatrix} \quad (6.36)$$

$$\mathrm{d}\mathbf{x}^k = \begin{bmatrix} \mathrm{d}\mathbf{u}^k \\ \mathrm{d}\mathbf{p}^k \end{bmatrix} \qquad \mathbf{r}^k = - \begin{bmatrix} \mathbf{r}_{\text{mom}}^k \\ \Delta t \mathbf{r}_{\text{mass}}^k \end{bmatrix}, \quad (6.37)$$

and the variables are updated by

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathrm{d}\mathbf{x}^k \quad (6.38)$$

when Equation (6.35) is solved at iteration k, preparing for the next iteration at $k+1$. The dimensions of the different blocks of the coefficient matrix $\mathbf{A}^k$ are $(dN_{\text{n}}) \times (dN_{\text{n}})$ for the upper left, with N_{n} the number of nodes and d the number of dimensions; $(dN_{\text{n}}) \times N_{\text{e}}$ for the upper right, with N_{e} the number of elements or cells; $N_{\text{e}} \times (dN_{\text{n}})$ and $N_{\text{e}} \times N_{\text{e}}$ for the lower left and right, respectively.

Remark. The system of equations in Equation (6.35) with coefficient matrix in Equation (6.36) is defined for the dynamic case of **u**-p formulation. In general, there are three cases of **u**-p formulation. The first case is dynamic, that is, all inertial terms are included. The second case is quasi-static, that is, all inertial terms are excluded. The third case is the exclusion of inertial terms in the fluid only. In the second and third case, $\bar{\mathbf{Q}}_2$ reduces to $\mathbf{Q}_2$ if the solid velocity $\mathbf{v} = \dot{\mathbf{u}}$ is integrated with simple backwards Euler as for the pore pressure rate $\dot{\mathbf{p}}$. Additionally, $\bar{\mathbf{K}}^k$ reduces to $\mathbf{K}^k$ in the second case. The residuals $\mathbf{r}^k_{\text{mom}}$ and $\mathbf{r}^k_{\text{mass}}$ must be adjusted accordingly.

6.5 Iteratively coupling the fully implicit coupled system

A system on the form of $\mathbf{A}\,\mathbf{x} = \mathbf{b}$, can be solved directly by computing the inverse of $\mathbf{A}$ and then solve $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. However, this is rarely a good approach as the computational cost of inverting the coefficient matrix $\mathbf{A}$ can be high especially if the dimension n of the system is very large. Iterative methods that avoid matrix inversions are equally as good, if not better, than direct methods. This is due to round-off errors and the computer's inability to represent floating-point numbers accurately anyway.

There are many iterative methods to choose from. Preconditioned stationary iterative methods of the Richardson family are written on the form

$$\mathbf{x}^{k+1} = \left(\mathbf{I} - \mathbf{P}^{-1}\mathbf{A}\right)\mathbf{x}^k + \mathbf{P}^{-1}\mathbf{b}, \tag{6.39}$$

where $\mathbf{P}^{-1}$ is a preconditioning matrix that aims to improve convergence by reducing the condition number of the system. Premultiplying all terms with $\mathbf{P}$ gives a system without inverted matrices:

$$\mathbf{P}\mathbf{x}^{k+1} = (\mathbf{P} - \mathbf{A})\mathbf{x}^k + \mathbf{b}. \tag{6.40}$$

The method can also be rewritten to solve for the incremental $\mathrm{d}\mathbf{x}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$:

$$\mathbf{P}\,\mathrm{d}\mathbf{x}^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k = -\mathbf{r}^k. \tag{6.41}$$

Block-structured systems may also use this approach. Consequently, a preconditioner $\mathbf{P}$ can be applied to the nonlinear system defined in Equation (6.35).

A good choice of preconditioner could be an upper triangular block preconditioner

$$\mathbf{P} = \begin{bmatrix} \bar{\mathbf{K}} & \mathbf{Q}_1 \\ & \tilde{\mathbf{S}}_A \end{bmatrix} \tag{6.42}$$

where the Schur complement approximation is

$$\tilde{\mathbf{S}}_A = \Delta t \mathbf{T} - \mathbf{C} - \alpha \mathbf{M}_p, \tag{6.43}$$

with $\alpha = 1/K_{\mathrm{dr}}$ and $\mathbf{M}_p$ is the pressure mass matrix

$$\mathbf{M}_p = \int_\Omega \varphi_i \varphi_j \, \mathrm{d}\Omega, \tag{6.44}$$

which for piecewise constant interpolation functions φ gives a diagonal matrix. The pressure mass matrix $\mathbf{M}_p$ acts as a preconditioner by ensuring diagonal dominance of the coefficient matrix $\tilde{\mathbf{S}}_A$. This particular choice of Schur complement approximation can be justified physically since it corresponds to the fixed-stress operator split discussed in Section 2.4.2, as shown by White *et al.* [90].

The resulting preconditioned system of equations in Equation (6.41) is triangular, and can be solved by iteratively solving each of the following subsystems:

$$\tilde{\mathbf{S}}_A \, \mathrm{d}\mathbf{p}^k = \mathbf{r}^k_{\mathrm{mass}} \tag{6.45}$$

$$\bar{\mathbf{K}}^k \, \mathrm{d}\mathbf{u}^k = \mathbf{r}^k_{\mathrm{mom}} - \mathbf{Q}_1 \, \mathrm{d}\mathbf{p}^k. \tag{6.46}$$

Two different solution strategies—or even softwares—may be used for solving the preconditioned system in Equations (6.45) and (6.46) because the coupling between the two systems is only through terms on the right-hand side. The derivations shown in this and the previous sections have been applied for coupling schemes of FEM and FVM [86, 88], where the former is solving for displacements and the latter for pore pressures. As noted in Chapter 4, the equations of motion are solved for displacements at the nodes in MPM in the same manner as FEM. The difference lies in the nodal interpolation, as explained in Section 4.3, with regards to particle location within the cell and weighting functions of integration point values. The resulting discrete system of equations for the Lagrangian phase is however similar for both MPM and FEM. Consequently, the displacement system of Equation (6.46) can be solved by a single-phase implicit MPM code with an added pore pressure term. The details of the discretisation and

set-up procedure for Equation (6.46) is therefore not covered here, as the details are in Chapter 4. The details on adapting the single-phase implicit MPM code to include the pore pressure term, is covered in Chapter 7.

From the matrix assemblies, we note that $\tilde{\mathbf{S}}_A$ has the same structure as the transmissibility matrix $\mathbf{T}$: sparse and symmetric. Furthermore, $\tilde{\mathbf{S}}_A$ is constant over the time step $n+1$. Sparse and symmetric systems can be solved efficiently by the conjugate gradient (CG) method. The conjugate gradient method does not require the sparse matrix to be fully assembled, thus saving memory. There are many ways to represent a sparse matrix in computer code. The CG solver from Hypre [108] use a so-called 7-stencil representation which associates the transmissibility entries for the centre cell and its neighbours with the centre cell.

Remark. The choice of K_{dr} depends on the type and dimension of problem modelled. That means that $K_{\mathrm{dr}}^{\mathrm{1D}} = E_{\mathrm{oed}}$, that is the constrained or oedometer modulus, for 1D oedometer problems, and $K_{\mathrm{dr}}^{\mathrm{3D}} = K$ for similar problems in 3D. Of course, the exact value of K_{dr} cannot be obtained for problems with complex boundary conditions. In that case, the choice of K_{dr} based on dimension is a good estimate. Kim [24] proves that a safe choice regardless of problem type and dimension is the $K_{\mathrm{dr}}^{\mathrm{3D}}$ (drained bulk modulus), as it will remain unconditionally stable. The other way around, that is, using the constrained modulus $K_{\mathrm{dr}}^{\mathrm{1D}}$ in higher dimension problems, leads to conditional stability.

6.6 Chapter summary

In this chapter we have seen that:

- the governing equations can be discretised in time and space by different time integration methods and spatial shape functions;
- a fully coupled implicit system of equations can be preconditioned so that the subproblems can be solved independently, leading to an iteratively coupled system;

From Chapter 4 we know that the resulting system of equations from the discretisation procedure of MPM, is equal in matrix form as FEM. Consequently, MPM and FVM can be iteratively coupled with MPM solving for displacements and FVM for pore pressures.

Chapter 7 demonstrates how to adapt the iteratively coupled system derived in the current chapter to a single-phase implicit MPM code.

CHAPTER 7

ITERATIVELY COUPLED IMPLICIT MPM-FVM: ADAPTATION

This chapter uses the framework derived in Chapter 6. The chapter discusses how to implement the fully discretised iteratively coupled system of equations into a single-phase MPM code. The FVM properties are adapted to the MPM framework to include the changes in cell contents during simulation.

Chapter outline

7.1 Creating the flow problem solver

The flow solution part, the FVM, must be created. Taking advantage of the Cartesian regular mesh, a 7-stencil representation is a good and easy choice. The code should also be able to track cell centre variables and compute cell centre gradients to nodes. To solve the system of equations resulting from the initial problem set-up, a prewritten library could be used, and is also recommended. Furthermore, an outer loop check of convergence of the combined solution from both the flow solver and the mechanical solver is needed.

7.1.1 Definition of control volume and cell average

At the start of the simulation during particle initialization, the particle volumes are computed from the geometrical cell volume as

$$\Omega_p = \frac{V_\mathrm{e}}{N_\mathrm{ep}}, \tag{7.1}$$

where Ω_p is the particle volume and V_e the geometric cell/element volume, with N_ep the number of particles per cell. However, the total particle volume of each cell will change during the course of the simulation since the particles will cross over from one cell to the next, in addition to the particle volume change due to loading. The cell volume is a property related to the material deformation and stresses and should therefore represent the material volume, not the geometrical volume. Consequently, the control volume of one cell is equal to the sum of all particle volumes within that particular cell at the start of the time step n, so that

$$V_c^n = \sum_p \Omega_p^n, \tag{7.2}$$

where V_c denotes the control or material volume as opposed to the geometrical cell volume V_e.

The areas through which the flux is flowing, on the other hand, are the same as the geometrical face area of the cell[1]. That is because it is the geometrical distance between pore pressure heads that governs the flux value, not material properties.

[1] While the flux area is equal to the geometrical area of cell face for cells that are not on the boundary, this is not true for the boundary cells. An improvement of this is beyond the scope of this thesis.

The cell average of the measured value φ is found from averaging the product of volume and field value

$$\bar{\varphi} = \frac{1}{V_c} \int_V \varphi \, \mathrm{d}V. \tag{7.3}$$

With MPM there is no continuous field function of φ to integrate over the cell volume, so Equation (7.3) is discretized into

$$\bar{\varphi} = \frac{1}{V_c} \sum_p \varphi_p \Omega_p. \tag{7.4}$$

Higo *et al.* [14] averaged the pore pressure value by dividing with the number of particles in cell. That approach equals Equation (7.4) when the particles have approximately equal volumes.

The cell volume assignment and the cell pore pressure average $\bar{p}^n$ are computed at the start of each time step. They are kept constant during the Lagrangian phase. At the end of the timestep, during the advection phase but before the particles move, the particles are assigned the updated cell pore pressure average $\bar{p}^{n+1}$.

7.1.2 Porosity, mass and volume updates

The mass of material points do not change for single-phase MPM. Mass is also conserved for two-point multiphase MPM [12, 13] and one-point multiphase MPM when porosity is constant [68].

During consolidation processes the change in volume equals the expellation of water. Water volume diminishes while solid volume remains constant. Or, mass of water diminishes, while mass of soil grains is unchanged.

The mass of one material point can be written as

$$M_p = m_\mathrm{w} + m_\mathrm{s} = \rho_\mathrm{w} V_\mathrm{w} + \rho_\mathrm{s} V_\mathrm{s} = n \rho_\mathrm{w} V + m_\mathrm{s}, \tag{7.5}$$

where ρ_w is the density of water, ρ_s the density of soil grains, both constant; V, V_w and V_s are the total volume and volume of water phase and solid phase, respectively, but V_s is constant; and n is the porosity or volume fraction of water. With ρ_s and V_s constant, then it follows that m_s is constant.

After each time step porosity n and total particle volume V is updated, and consequently, the total particle mass is updated too, as follows:

$$M_p^{n+1} = m_\mathrm{s} + n^{n+1} V^{n+1} \rho_\mathrm{w}. \tag{7.6}$$

7.1.3 The coupling matrices $\mathbf{Q}_2 = \mathbf{Q}_1^\mathsf{T}$

An entry in the coupling matrix $\mathbf{Q}_2$ for cell c is

$$\mathbf{Q}_{2,c} = \sum_{i \in c} \boldsymbol{\nabla} \cdot N_{ci} V_c, \tag{7.7}$$

where indices c and i denotes the cell and nodes respectively, N_{ci} is the interpolation from node n to cell centre c, and V_c is the cell volume. The fully written out $\mathbf{Q}_2$ matrix has a block structure with blocks of size $1 \times d$ per cell node i where d is the dimension of the problem.

7.1.4 The compressibility and mass pressure matrices $\mathbf{C}$ and $\alpha\mathbf{M}_p$

The compressibility and mass pressure matrices are diagonal matrices where the entries are

$$\mathbf{C}_c = \frac{n}{K_\mathrm{w}} V_c \mathbf{I} \tag{7.8}$$

$$\alpha\mathbf{M}_{p,c} = \frac{1}{K_\mathrm{dr}} V_c \mathbf{I}, \tag{7.9}$$

where c denotes the cell, V_c the cell volume, and n is the material porosity with K_w the bulk modulus of water and $\alpha = 1/K_\mathrm{dr}$ the drained bulk compressibility of the soil.

7.1.5 Transmissibility matrix T

FVM shows local mass conservation, meaning that the flux out of one face of a cell is equal but with opposite signs of the flux into the neighbouring cell through the same face. The transmissibility matrix T represents the flux between cells. Its entries are found from the integral of the face normal component of the pore pressure gradient

$$\hat{\mathbf{q}} = -\frac{k}{\rho_\mathrm{w} g} \boldsymbol{\nabla} p \tag{7.10}$$

over the element face area. In this section we are going to assemble the transmissibility matrix $\mathbf{T}$ defined in Section 6.3.1. The transmissibility matrix is usually constructed geometrically.

Any grid discretisation can be used with FVM, so it is possible to use the same grid as the MPM discretisation. In this work, a cell-centred scheme on a Cartesian regular hexahedral grid is adopted. With a Cartesian grid

these face normals are the same as the global Cartesian axes. In addition, there is no need to search for the neighbouring cells used for computing the flux through the cell as the surface, as they are found by adding or subtracting 1 from the current normalised cell coordinates in the direction of the face normal.

Yet another advantage with a Cartesian grid is that a line connecting the cell centres of two neighbouring cells is normal to the cell boundary. This is the *orthogonality property*. When it holds, we can use the simplest flux schemes for computing the transmissibility between cells: the two-point flux approximation (TPFA), without losing consistency. TPFA cannot be used on meshes that deform, because the deformation may break the orthogonality property. Because the mesh of MPM is fixed, deformation of that kind is no issue, and TPFA can safely be adopted. There are, however, more advanced flux schemes, but the exploration of those is beyond the scope of this thesis.

TPFA is approximating the pore pressure gradient between the neighbouring cells by using two points, namely the pore pressure value of the current cell and its neighbour cell in either of the $2d$ directions, with d being the number of dimensions. The following derivation is for the 3-dimensional case. Per finite volume standard notations, the neighbour cells are commonly named after their abbreviated compass locations: *N*orth, *E*ast, *S*outh, *W*est, and the addition of *T*op and *B*ottom in 3D, with P denoting the current cell. Additionally, the edges between the centre cell and the neighbours are denoted with the corresponding lowercase letters of the neighbours.

The flux $q_e = \hat{\mathbf{q}}_e \cdot \mathbf{n}_e$ between the centre cell P and the east cell E, can be linearly approximated by

$$q_e = \frac{k}{\rho_\mathrm{w} g} \frac{p_P - p_E}{\Delta \mathbf{x}_e} \tag{7.11}$$

where $\Delta \mathbf{x}_e$ is the distance between the cell centres of cell P and E. With a regular grid, the denominator is constant for each cell in each dimension. However, the cell centre distance does not have to be constant, it may vary from cell to cell depending on the mesh.

The flux is computed at the faces. To obtain the volumetric flow rate $\dot{V}_e$ through the east face e, the flux q_e is multiplied with the area of the face A_e: $\dot{V} = qA$. The total flow rate through the cell is the sum of the flow rates

on all its faces, leading to

$$\dot{V}_{\text{total}} = \sum_f \frac{k}{\rho_w g} \frac{p_P - p_f}{\Delta \mathbf{x}_f} A_f \tag{7.12}$$

$$= \frac{k}{\rho_w g} \sum_f \frac{A_f}{\Delta \mathbf{x}_f} p_P - \frac{k}{\rho_w g} \sum_f \frac{A_f}{\Delta \mathbf{x}_f} p_f \tag{7.13}$$

$$= T_P p_P - \sum_f T_f p_f, \tag{7.14}$$

where f is an index referring to each of the six faces of a 3D cell. We note that

$$T_P = \sum_f T_f. \tag{7.15}$$

The transmissibility matrix is the matrix that results from assembling a system of equations from Equation (7.14) for all cells. The final assembled transmissibility matrix $\mathbf{T}$ is in 3D a 7-banded symmetric sparse matrix with dimension equal to the number of cells.

Because of the coupling of FVM to MPM, there must be a way to mark the empty cells. The empty cells, that is, the cells that do not contain any material points, are automatically given a pore pressure $p = 0$. To avoid resizing the transmissibility matrix during the simulation runtime, empty cells are given the value 1 in the cell centre, and 0 elsewhere. This will not negatively affect the Schur complement $\tilde{\mathbf{S}}_A$, and the conjugate gradient method may still be applied.

7.2 Flow boundary conditions

The flow boundary conditions with the structured FVM using a TPFA scheme are applied to the boundary faces, and derived with the help of ghost cells. TPFA is a central differencing scheme, and can be replaced with higher order schemes and schemes that also use the corner neighbour cells. The ghost cells are a means to derive the flux values at the boundary faces, and they are not used in the simulation.

Let us assume that there is a boundary on the east face of the boundary cell P. This means that the cell E is outside the boundary. Consequently, the transmissibility matrix must be altered to include the boundary condition because the ghost east cell pore pressure p_E can now be written in terms of the center cell pore pressure p_P.

Regardless of type of boundary condition the transmissibility coefficient $T_E = 0$, while the centre cell coefficient T_P is changed depending on boundary condition type. The right hand side of Equation (6.45) should also be adjusted if the boundary value is not equal to 0. However, since we are dealing with either no-flux or drained boundaries, all applied boundary values are 0 and consequently no adjustment is needed nor presented herein. The next sections will derive the adjusted $\tilde{T}_P$ for each of the two types of boundary conditions in detail.

7.2.1 Neumann boundary conditions

Neumann boundary conditions assign a given flux $\bar{q}$ in the normal direction of the boundary face. Symmetric boundary conditions are a special case of Neumann boundary conditions where the boundary flux $\bar{q} = 0$. Zero flux or symmetric boundary conditions can be obtained by setting the pore pressure of the ghost cell to $p_E = p_P$, as this will make the flux through the boundary face equal to 0. Inserting this into Equation (7.14) gives

$$T_P p_P - \sum_{\substack{f=N,S,E,\\W,T,B}} T_f p_f = T_P p_P - T_E p_P - \sum_{\substack{f=N,S,\\W,T,B}} T_f p_f \tag{7.16}$$

$$= (T_P - T_E) p_P - \sum_{\substack{f=N,S,\\W,T,B}} T_f p_f. \tag{7.17}$$

Therefore the adjusted centre cell coefficient is

$$\tilde{T}_P = T_P - T_E = T_W + T_N + T_S + T_T + T_B. \tag{7.18}$$

7.2.2 Dirichlet boundary conditions

These boundary conditions are for imposing a given pore pressure $\bar{p}$. It is used to assign the open, drained surface, that is, the surface with pore pressure $p = 0$. As with the Neumann boundary conditions, the boundary face coefficient is $T_E = 0$.

The correction of the centre cell coefficient can be derived as follows. Let the east cell E be a ghost cell and set its value to $p_E = -p_P$. This gives $p = 0$ at the boundary by use of linear interpolation. Inserting $p_E = -p_P$

into Equation (7.14) gives

$$T_P p_P - \sum_{\substack{f=N,S,E,\\ W,T,B}} T_f p_f = T_P p_P - T_E(-p_P) - \sum_{\substack{f=N,S,\\ W,T,B}} T_f p_f \tag{7.19}$$

$$= (T_P + T_E) p_P - \sum_{\substack{f=N,S,\\ W,T,B}} T_f p_f. \tag{7.20}$$

Therefore the adjusted centre cell coefficient is

$$\tilde{T}_P = T_P + T_E = 2T_E + T_W + T_N + T_S + T_T + T_B. \tag{7.21}$$

7.2.3 Moving boundary conditions

Consider a slope where intially the groundwater table is coinciding with the surface, see Figure 7.1. When this slope is undergoing large deformations, the final drained boundary is not at the same location as the initial drained boundary, assuming that the groundwater table still is coinciding with the slope surface.

Applying the boundary conditions to simulate an open boundary need only be done once in a problem with small deformations as the material boundary will not change substantially from the initial configuration. But in a problem with large deformations, the material boundary and hence the boundary conditions need to be updated. To accomodate for this, the material boundary conditions are applied at the beginning of each time step when the transmissibility matrix T is assembled. The grid boundaries are fixed.

The transmissibility matrix T depend only on the grid and the permeability, hence this matrix will be constant during an entire simulation if permeability is constant and boundaries are unchanged. However, when FVM is coupled with MPM, the boundary of the material will change during the course of the simulation. Therefore the boundary conditions should be able to move with the deforming material.

The moving boundary is applied when the neighbour cell has a cell volume of less than some treshold value $V_{\min}$. At the beginning of each timestep all cell volumes of domain are initialized to 0, before the sum of all particles volumes within the cell is assigned. Therefore, any cell with particles will have a positive non-zero value of the cell volume.

With $V_{\min} = 0$ the moving boundary is assigned on the cell faces between material cells and completely empty cells. It may be reasonable to set $V_{\min} = \alpha V$, where V is the initial fully-filled cell volume and $0 \leq \alpha \leq 1$. For

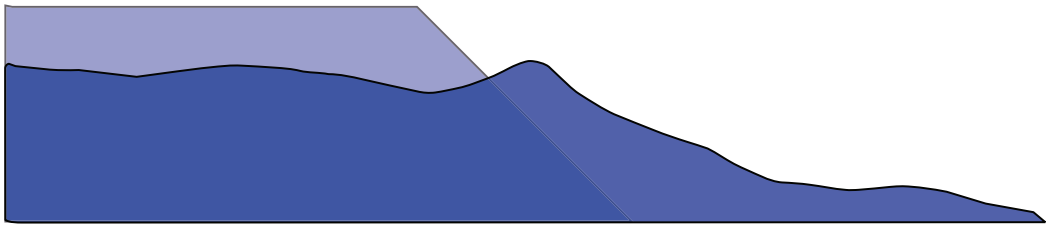


Figure 7.1: Sketch of a slope in initial state (light) and deformed state (dark). The deformed slope geometry differs greatly from the initial slope geometry.

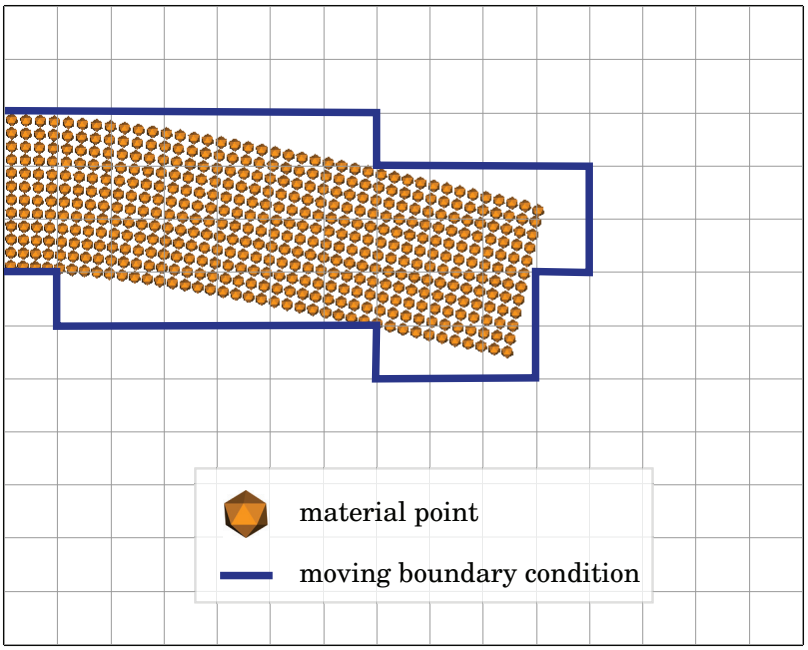


Figure 7.2: Moving finite volume boundary conditions. On each timestep the boundary of the material is determined by identifying the cells which has at least 1 empty neighbour cell. When detected the boundary condition is applied to the transmissibility matrix as described in Section 7.2.

instance, for an initial 4-particle cell, a cell can be considered empty if it contains only 1 particle in the context of drained boundary conditions, which suggests $\alpha = 0.25$. In this thesis, however, $V_{\min} = 0$.

The extra cost of applying updated drained boundaries is of $\mathcal{O}(1)$ since it depends on a table lookup. This is because any given cell has known neighbours; no search is needed.

This approach described here is limited because the exact material boundary is not detected, illustrated in Figure 7.2. In Figure 7.2 a very coarse mesh is used, but by refining the mesh, the inaccuracy stemming from the crude outline of the boundary conditions is improved. The inaccuracy with the boundary following the cell outline will lead to slightly wrong pore pressure gradients on the actual material boundary.

7.3 Adding pore pressure to single-phase MPM

The single-phase code used in this thesis, the `ImpMPM` module of Uintah software [29], is based on the framework of Guilkey & Weiss [18]. Refer to Section 4.6 for other implicit MPM formulations. Due to the iteratively coupled approach, any implicit formulation may be used, as long as the flow solver is given a displacement increment. The single-phase implicit MPM code solves the following incremental system of equations:

$$\bar{\mathbf{K}}^k \, \mathrm{d}\mathbf{u}^k = \mathbf{F}_{\text{ext}}^{n+1} - \mathbf{F}_{\text{int}}^k - \mathbf{M}\left(\frac{1}{\beta \Delta t^2} \mathbf{u}^k - \frac{1}{\beta \Delta t} \mathbf{v}^n - \frac{1-2\beta}{\beta} \mathbf{a}^n \right) \tag{7.22}$$

where $\mathbf{F}_{\text{int}}^k$ depends on the stress state $\boldsymbol{\sigma}^k$, the global tangential stiffness matrix is $\bar{\mathbf{K}}^k$, $\mathrm{d}\mathbf{u}^k$ is the new change in displacement, and $\mathbf{F}_{\text{ext}}^{n+1}$ is the external force applied at time step $n+1$. The inertial term on the right hand side of Equation (7.22) is written out with the Newmark integration of acceleration $\mathbf{a}^k$ from Equation (6.28)[2] The superscripts k and $k+1$ denote local iteration numbers, where k is the result from the previous iteration and $k+1$ is the result of the next. Any superscript with k denotes a value at time step $n+1$, so for simplicity, superscript $n+1$ is omitted whenever k is present. Displacement $\mathbf{u}^{k+1}$ is updated from

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathrm{d}\mathbf{u}^k. \tag{7.23}$$

[2]Recall that $\mathbf{u}^n = \mathbf{0}$ at the beginning of each timestep, so that at the end of the timestep $\mathbf{u}^{n+1}$ equals the displacement increment $\mathrm{d}\mathbf{u} = \sum_k \mathrm{d}\mathbf{u}^k$ of the step.

The iteratively coupled solution of incremental displacements defined in Equation (6.46) is fully written out as:

$$\bar{\mathbf{K}}^k \, \mathrm{d}\mathbf{u}^k = \mathbf{F}_{\text{ext}}^{n+1} - \mathbf{F}_{\text{int}}^k - \mathbf{M}\left(\frac{1}{\beta \Delta t^2}\mathbf{u}^k - \frac{1}{\beta \Delta t}\mathbf{v}^n - \frac{1-2\beta}{\beta}\mathbf{a}^n\right) - \mathbf{Q}_1 \mathbf{p}^k - \mathbf{Q}_1 \, \mathrm{d}\mathbf{p}^k. \tag{7.24}$$

Noticing that $\mathbf{p}^{k+1} = \mathbf{p}^k + \mathrm{d}\mathbf{p}^k$ allows us to group the pore pressure terms above so that the iteratively coupled system for solving for the incremental displacements $\mathrm{d}\mathbf{u}^k$ is

$$\bar{\mathbf{K}}^k \, \mathrm{d}\mathbf{u}^k = \mathbf{F}_{\text{ext}}^{n+1} - \mathbf{F}_{\text{int}}^k - \mathbf{M}\left(\frac{1}{\beta \Delta t^2}\mathbf{u}^k - \frac{1}{\beta \Delta t}\mathbf{v}^n - \frac{1-2\beta}{\beta}\mathbf{a}^n\right) - \mathbf{Q}_1 \mathbf{p}^{k+1}. \tag{7.25}$$

The only difference between Equations (7.22) and (7.25) is the pore pressure contribution term. Note that $\mathbf{F}_{\text{int}}$ depends solely on effective stress $\boldsymbol{\sigma}'$, which is equal to total stress $\boldsymbol{\sigma}$ for the single-phase MPM. Consequently, to turn a single-phase implicit MPM code into an iteratively coupled MPM code, the only change needed is to add the pore pressure contribution found from solving Equation (6.45).

The coupling matrix $\mathbf{Q}_1$ for a node i is found from

$$\mathbf{Q}_{1,i} = \sum_{c \in i} \boldsymbol{\nabla} \cdot N_{ic} \mathbf{I} V_c, \tag{7.26}$$

where the gradient from node i to cell centre c is constant for a regular Cartesian mesh. An alternative way of introducing the pore pressure term $\mathbf{Q}_1 \mathbf{p}^{k+1}$ into Equation (7.25), is by including it in the internal force term to get $\mathbf{F}_{\text{int, total}}^k$. The advantage is that the coupling matrix $\mathbf{Q}_1$ is not assembled. An additional advantage is the physical interpretation of finding the internal force based on total stress in the material. All particles within one cell have the same pore pressure p equal to the cell pore pressure $\bar{p}_c$. Introducing the particle pore pressure p in the internal force expression for node i, gives:

$$\mathbf{F}_{\text{int, total}}^k = \sum_{p=1}^{N_\text{p}} \boldsymbol{\nabla} \cdot N_{ip} \left(\boldsymbol{\sigma}_p'^k + p^{k+1} \mathbf{I}\right) V_p^k. \tag{7.27}$$

Note that the volume of cell V_c defined in Equation (7.3) is equal to the sum of particle volumes V_p in the cell c at start of time step. Also note that the shape functions are chosen to be linear, which means that its gradient is constant. Hence it does not matter where within the cell the pore pressure value is sampled from.

7.4 Outline of one time step

For one time step there are two main tasks: solving the flow problem with output of pore pressure p, and solving the mechanical problem with output of displacement $\mathbf{u}$. When convergence is achieved, regular MPM housekeeping is performed.

1. Mapping phase

Identical to the mapping phase of single-phase MPM, adding the following steps:

1) Compute new cell volume:

$$V_c^n = \sum_p V_p^n \tag{7.28}$$

2) Compute cell average pore pressure:

$$\bar{p}^n = \frac{1}{V_c^n} \sum_p p_p^n V_p \tag{7.29}$$

In addition, the nodal displacement vector should be initialized to $\mathbf{u}^0 = \mathbf{0}$.

2. Lagrangian phase

The flow problem and the mechanical problem are solved iteratively, using the other output as input, until global convergence is reached.

1) Initialization. Schur complement approximation

$$\tilde{\mathbf{S}}_A = \Delta t \mathbf{T} - \mathbf{C} - \alpha \mathbf{M}_p \tag{7.30}$$

is assembled from Equations (7.8), (7.9) and (7.14). Boundary conditions are applied to the transmissibility matrix $\mathbf{T}$.

2) Solve flow problem in Equation (6.45) based on $\mathbf{u}^k$ from the mechanical problem output.
   i. Assemble right hand side $\mathbf{r}_{\text{mass}}^k$ in Equation (6.32) of the flow problem.
   ii. Input $\tilde{\mathbf{S}}_A$ and $\mathbf{r}_{\text{mass}}^k$ to a library solver of choice. In this implementation the conjugate gradient solver of Hypre [108] was used.

3) Solve the mechanical problem as described in Section 4.6.1, but add contribution of $\mathbf{p}^{k+1}$. This can be done in two ways:

i. directly through computing the internal force $\mathbf{F}^k_{\text{int, total}}$ as in Equation (7.27) where it is noted that the particle pore pressure p^{k+1} is equal to the corresponding cell pore pressure $\bar{p}^{k+1}$ computed by the flow solver.

ii. add the coupled pore pressure term $-\mathbf{Q}_1\mathbf{p}^{k+1}$ to the right hand side of Equation (4.58), as shown in Equation (6.46).

4) Check for global convergence [90]:

$$\left\|\mathbf{r}^k_{\text{mass}}\right\| < \text{ITOL}. \tag{7.31}$$

If no convergence go to step (2) again.

3. Convection phase

The convection phase is identical to the mapping phase of single-phase MPM, with the addition of the following steps:

1) Update the particle pore pressure $p^{n+1} = \bar{p}^{k+1}$ directly from the computed cell pore pressure.

2) Update particle porosity and mass.

7.5 Chapter summary

In this chapter we have seen that:

- several FVM quantities must be adapted to MPM because of change in cells during simulation;
- FVM boundary conditions must be updated to adapt to the changing material boundary;
- only minor adjustments are needed to extend an existing implicit MPM code to include the generation and dissipation of pore pressures.

The next chapter demonstrates and verifies the coupled MPM-FVM method, and compares results with the explicit coupled MPM presented in Chapter 5.

On two occasions I have been asked, — "Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?"

Charles Babbage, 1864

CHAPTER 8

VERIFICATION AND VALIDATION

This chapter aims to verify the two coupled MPM approaches that were presented in Chapters 5 and 6. The two approaches differ in three ways: one is an explicit one-point velocity formulation, and the other is an implicit displacement-pressure formulation that coupled MPM with FVM. Both approaches are verified by the one-dimensional consolidation problem, and briefly compared to each other and the analytical solution. To validate that the iteratively coupled MPM-FVM method works for the large deformation problems, a poroelastic cantilever beam subject to large deflection is modelled.

Chapter outline

8.1 Verification: One-dimensional consolidation problem

The one-dimensional consolidation problem is modelled to verify that the both coupled MPM approaches presented in this thesis solve the underlying mathematical diffusion equation. The problem is described by the Terzaghi one-dimensional consolidation equation

$$c_v \frac{\partial^2 p}{\partial z^2} = \frac{\partial p}{\partial t}, \tag{8.1}$$

where p is the pore pressure, z the spatial variable, and the consolidation coefficient c_v defined here as

$$c_v = \frac{k}{\rho_\mathrm{w} g \left(\frac{1}{K_\mathrm{dr}^\mathrm{1D}} + \frac{n}{K_\mathrm{w}} \right)}, \tag{8.2}$$

where k is the hydraulic conductivity, ρ_w is the density of water, g is the magnitude of acceleration of gravity, n the porosity, $K_\mathrm{dr}^\mathrm{1D}$ is the one-dimensional constrained modulus, and K_w the bulk modulus of water.

The consolidation differential equation (8.1) is analogous to the heat equation. It may be solved analytically if given initial and boundary conditions. Here, one-way drainage has been assumed with $z = h$ at the impermeable boundary where h is the height of the soil column. The initial condition is

$$p(z,0) = p_0, \tag{8.3}$$

and the boundary conditions are

$$p(0,t) = 0 \tag{8.4}$$

$$\frac{\partial p}{\partial z}(h,t) = 0. \tag{8.5}$$

The analytical solution is obtained from Fourier series analysis and given as

$$p(z,t) = p_0 \frac{4}{\pi} \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{2k-1} \cos\left[(2k-1) \frac{\pi}{2} \frac{z}{h} \right] \exp\left[-(2k-1)^2 \frac{\pi^2}{4} T \right], \tag{8.6}$$

where the dimensionless time factor

$$T = \frac{c_v t}{h^2} \tag{8.7}$$

has been introduced.

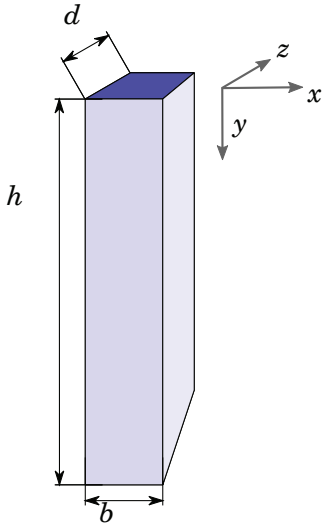


Figure 8.1: Schematic illustration of the model geometry. The height h of the column is 1 m, and the section area $b \times d = 0.1 \times 0.1\,\mathrm{m}^2$. The dark shaded area on the top of the column is the open boundary, while the lightly shaded sides and the bottom are impermeable boundaries.

8.1.1 Model setup

Model geometry. A soil column with height 1 m and square base area $0.1 \times 0.1\,\mathrm{m}^2$ is modelled. The top surface is an open surface, which equals a Dirichlet boundary condition with prescribed pore pressure $\bar{p} = 0$. The remaining 5 surfaces (bottom and 4 sides) are impermeable, which equals to symmetric or no-flux Neumann boundary conditions, with prescribed flux $\bar{q} = 0$. The geometry is illustrated in Figure 8.1.

Material properties. The soil is modelled as a saturated, isotropic elastic material with input parameters $K = 3.333\,333\,\mathrm{MPa}$ and $G = 5\,\mathrm{MPa}$, which corresponds to a constrained modulus of $K_{\mathrm{dr}}^{\mathrm{1D}} = 10\,\mathrm{MPa}$ and Poisson ratio $v = 0$. A compressible Neo-Hookean constitutive model was used, which reduces to Hooke's law in the small strain regime.

The poroelastic properties are hydraulic conductivity $k = 10^{-3}\,\mathrm{m/s}$, bulk modulus of water $K_w = 2.2\,\mathrm{GPa}$, and initial porosity $n = 0.4$. The water density is $\rho_w = 1\,000\,\mathrm{kg/m}^3$ and the saturated density $\rho = 2\,000\,\mathrm{kg/m}^3$.

Loading conditions. The initial effective stress in the soil is set to $\sigma_0' = 0\,\mathrm{kPa}$, and the initial pore pressure is set to $p_0 = 10\,\mathrm{kPa}$ (pressure). A distributed compressional load of $q = 10\,\mathrm{kPa}$ is applied at the top drained boundary at $t = 0^+\,\mathrm{s}$ for the implicit method and $t = 0.01\,\mathrm{s}$ for the explicit method. The load is constant for the rest of the simulation.

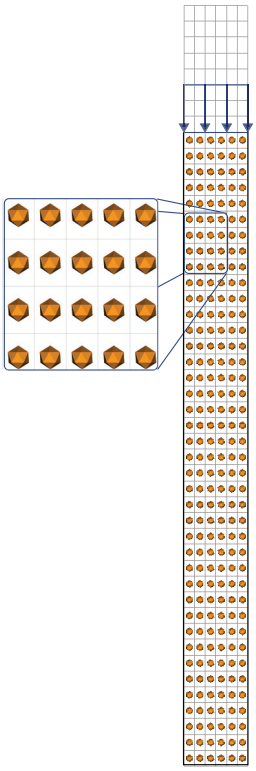


Figure 8.2: Model setup for simulation of Terzaghi 1D consolidation problem. The discretisation shows 1 material point per cell, with 6 cells in the horizontal x-direction, 40 cells in the vertical y-direction, and 1 cell in the in-plane z-direction. The black lines indicate no flow boundary conditions $\bar{q} = 0$. The open boundary with $\bar{p} = 0\,\text{kPa}$ is given a distributed load of $q = 10\,\text{kPa}$.

Discretisation. The problem was divided into 40 elements in the vertical y-direction with an extra 8 empty elements above the material body. There were 6 elements in the x-direction and 1 element in the z-direction. Each element had 1 material point. The discretisation is illustrated in Figure 8.2.

An additional simulation with 10+2 elements was also run for the implicit method to check for mesh influence on the solution.

Size of time steps. The simulation with the implicit method is carried out with time steps $\Delta t = 10^{-4}\,\text{s}$, $\Delta t = 10^{-3}\,\text{s}$, and $\Delta t = 10^{-2}\,\text{s}$. The purpose is to see how the time step size affects the solution. With oscillations due to the incompressible limit at the start of the simulation, the smallest time step $\Delta t = 10^{-4}\,\text{s}$ is chosen to check how the implicit method performs in this range. A larger time step of $\Delta t = 10^{-2}\,\text{s}$ was chosen to check how the implicit method performs with larger time steps.

The explicit method is limited by two criterions, so time step size was chosen as $\Delta t = 10^{-5}\,\text{s}$.

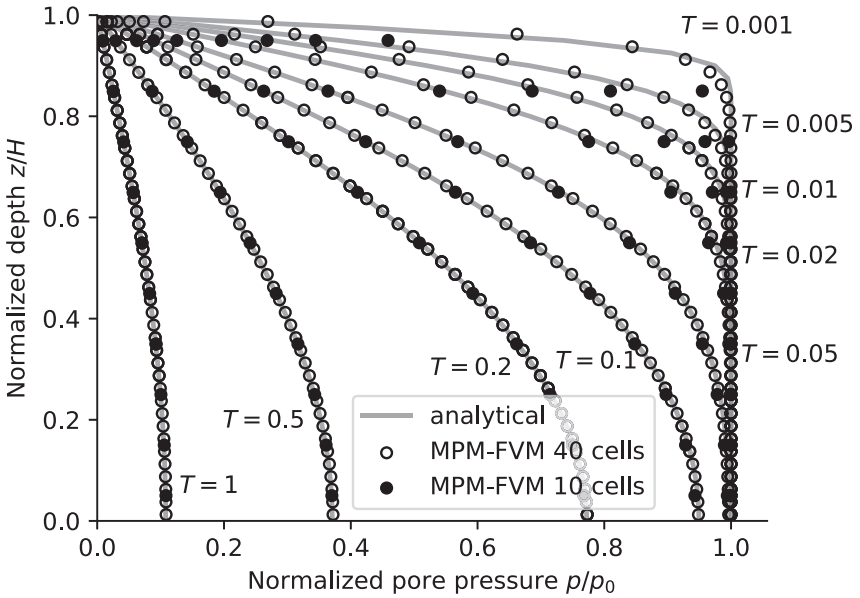


Figure 8.3: Excess pore pressure isochrones for the Terzaghi one-dimensional consolidation problem. Results from the iteratively coupled MPM-FVM with a mesh of 10 cells (filled circles) and a mesh of 40 cells (open circles) and time step $\Delta t = 0.001\,\mathrm{s}$.

8.1.2 Results and discussion

The iteratively coupled MPM-FVM produced results that are very close to the analytical solution even at times $T < 0.05$ (Figure 8.3). The quasi-static MPM-FVM showed no oscillations at small time steps as expected (see Figure 8.6), while the dynamic MPM-FVM method leads to some initial oscillations (see Figure 8.4). The oscillations experienced with the explicit method are greater than the oscillations experienced with the dynamic implicit method. As seen in Figure 8.5, the oscillations for the explicit method are so large that the results for the small time steps seem very far off. Any other choice of time step than $\Delta t = 10^{-5}\,\mathrm{s}$ for the explicit method resulted in a numerical breakdown. The oscillations come from the hyperbolic property of the dynamic differential equation and are worse for smaller time steps. The differences between the explicit and implicit methods that are noted here may also stem from the choice of primary variables of the respective formulations. The $\mathbf{u}$-p formulation will have

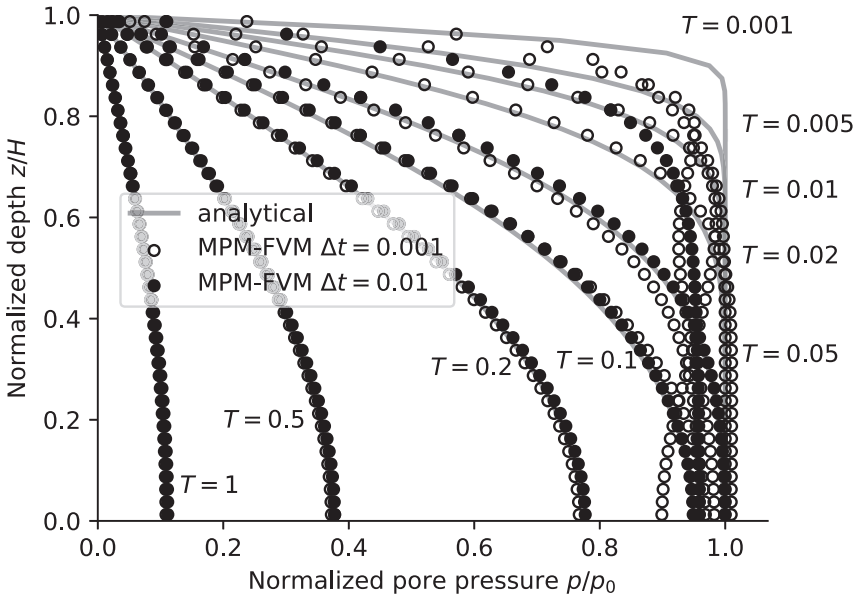


Figure 8.4: Excess pore pressure isochrones for the Terzaghi one-dimensional consolidation problem. Results from the implicit iteratively coupled MPM-FVM method with inertial terms included (dynamic) for time steps $\Delta t = 0.01\,\mathrm{s}$ (filled circles) and $\Delta t = 0.001\,\mathrm{s}$ (open circles).

a dampening effect in addition to the dampening effect introduced by the time integration scheme.

There are several measures to minimize the oscillations present in the explicit simulation. The mass can be scaled up so that the time step may be increased [68, 77] , so that the CFL condition and the condition from Vermeer & Verruijt [79] still hold. This approach skips the problems with the small time steps, and that may lead to other types of problems converging and not failing. As noted by Zabala & Alonso [68], mass-scaling can only be performed in the quasi-static regime anyway, which is a regime more suitable for implicit time integration. Jassim *et al.* [9] also introduced a local damping force for both the solid and the fluid phase. None of these approaches has been attempted in this thesis.

The advantage of the implicit MPM-FVM is the lack of oscillations in the undrained limit at the start of the simulation, and that larger time steps may be used while still obtaining an accurate enough solution, see

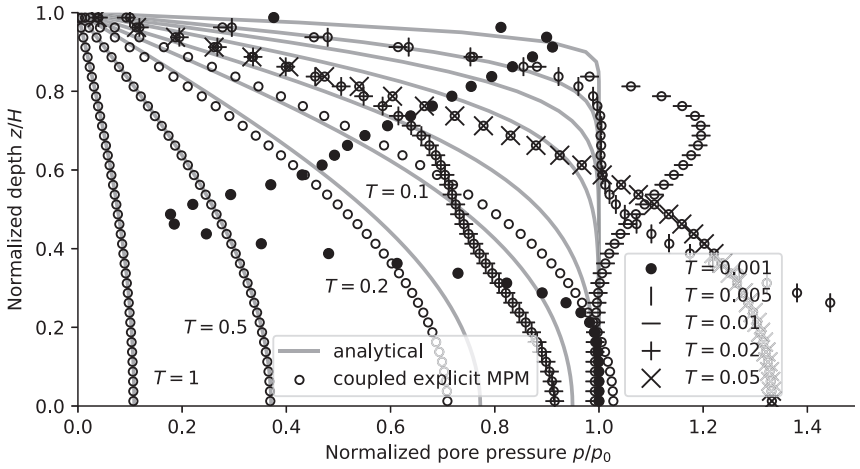


Figure 8.5: Excess pore pressure isochrones for the Terzaghi one-dimensional consolidation problem. Results from the coupled explicit dynamic MPM with a mesh of 40 cells and a time step $\Delta t = 10^{-5}$ s.

Figure 8.6 and Figure 8.4 for quasi-static and dynamic results, respectively. Without any additional measures, such large time steps would lead to a numerical breakdown of the coupled explicit MPM. For the dynamic case, the larger time step had higher accuracy for $T > 0.01$. This demonstrates the dampening effect from the time integration scheme.

8.1.3 Summary

Both methods converged to the correct result for the Terzaghi one-dimensional consolidation problem described by Equation (8.1).

Oscillations

- The iteratively coupled implicit quasi-static MPM-FVM had the most accurate results, with no oscillations even for $T < 0.02$.
- The iteratively coupled implicit dynamic MPM-FVM experienced some oscillations for $T < 0.02$, but the oscillations were not severe.
- The coupled explicit MPM had large oscillations for $T < 0.2$, so severe that the resulting consolidation curves look wrong.

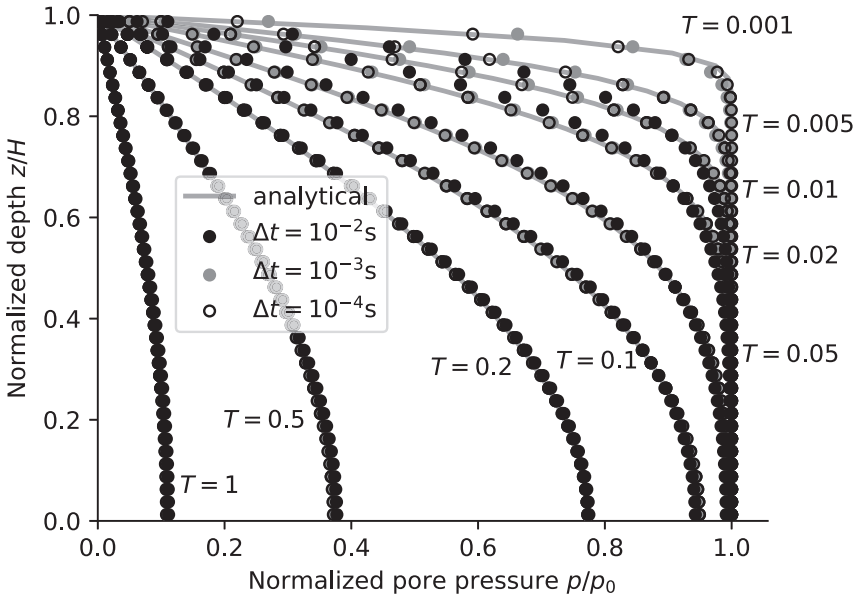


Figure 8.6: Excess pore pressure isochrones for the Terzaghi one-dimensional consolidation problem. Results from the implicit iteratively coupled MPM-FVM (quasi-static) with a mesh of 40 cells and time steps $\Delta t = 10^{-4}$ s, $\Delta t = 10^{-3}$ s, and $\Delta t = 10^{-2}$ s.

Time step size

- The coupled explicit MPM needed time step size $\Delta t = 10^{-5}$ s to converge.
- The iteratively coupled implicit dynamic MPM-FVM had good results with time step size $\Delta t = 10^{-2}$ s.
- The iteratively coupled implicit quasi-static MPM-FVM had very accurate results for time step sizes $\Delta t = 10^{-4}$ s and $\Delta t = 10^{-3}$ s, and tolerable good results for time step size $\Delta t = 10^{-2}$ s.

The time step size for the implicit MPM can, therefore, be of the order 10^2 to 10^3 larger than the time step size for the explicit MPM.

8.2 Validation: Large deflection of a poroelastic beam

The previous section verified that the iteratively coupled implicit MPM-FVM method (in this section refer to as *the proposed method* for brevity) did solve the mathematical equations of the one-dimensional consolidation problem. However, we also need to validate that the proposed method is able to model the physics of large deformation problems. Therefore, in this section, a poroelastic beam subject to large deflection is modelled.

Large deflection of an elastic beam is a common benchmark of MPM, as it demonstrates the capability of modelling large deformations and rotations without having to handle large strains at the same time. Large deflection of a poroelastic beam would show that there is a coupling of flow and displacements while the beam is experiencing large deformations. We would expect pore pressure generation where the beam is experiencing tension or compression from the bending moment generated by the load. We would expect suction where there are tension and pressure where there is compression. We would expect that the beam deflects further when the generated pore pressure dissipates. In short, we expect two-way coupling: that the displacements affect the pore pressure and the pore pressure affects the displacements. The results will be compared with results obtained by Li *et al.* [109], where a poroelastic beam with axial diffusion and no transverse diffusion was modelled with the finite difference method.

To summarise, in this section we seek to verify that

1. the method is able to model large deformations with particles crossing cells;
2. the method is two-way coupled;
3. the results are in line with results obtained in [109].

Analytical beam deflection. There exist no analytical solution to the saturated poroelastic beam bending problem, but there exist several elastic beam bending theories. The Euler-Bernoulli beam theory is valid for transverse loads and assumes no change in cross-sectional areas or length of centreline of the beam. Additionally, shear effects are assumed to be negligible compared to bending effects.

The limit between small and large deflections can be determined from the maximum slope of the beam. For example, with slope $\mathrm{d}y/\mathrm{d}x = 0.1$, the

denominator of the non-linear Euler-Bernoulli equation

$$\frac{y''}{\left[1+(y')^2\right]^{3/2}} = -\frac{M}{EI} \tag{8.8}$$

becomes 1.01504. The deviation from 1 is small, and therefore acceptable for small deflection theory.

In general, small deflection theory applied to large deflection problems overestimates the final deflection. When the beam is subject to large deflection, the moment arm of the loading will decrease, resulting in less moment applied to the beam. Consequently, with smaller moment the deflection will be less than the linearly extrapolated deflection obtained from the small deflection theory.

There exist several attempts at producing analytical solutions for elastic beams. However, the initial loading step of the poroelastic beam in this thesis is verified by the simple small deflection theory. The small deflection theory does not take into account changes of the x-coordinate. The analytical solution of deflection y for a cantilever beam subject to a distributed load w is found from

$$y(x) = -\frac{wL^4}{24EI}\left(\frac{x}{L}\right)^2\left[6-4\frac{x}{L}+\left(\frac{x}{L}\right)^2\right], \tag{8.9}$$

so that the maximum deflection $\Delta_{\max}$ is given as

$$\Delta_{\max} = \frac{wL^4}{8EI}, \tag{8.10}$$

where L is the length of the beam, E the elastic stiffness modulus and $I = d \cdot h^3/12$ the second moment of area about the y-axis, where h and d are the height and depth of the beam, respectively. The distributed load $w = q \cdot d$ where q is the surface load. The reasons for using *depth* to denote the more common *breadth* of a beam, is that the model in this section is in effect a 2D plane strain model, and as such, has no actual breadth.

Cross-sectional pore pressure moment. Li *et al.* [109] plotted the cross-sectional pore pressure moment M_p along the beam coordinates. Their finite difference method solution to the beam bending problem solved directly for M_p, defined as

$$M_p = -\int_A p\, y \, \mathrm{d}A, \tag{8.11}$$

where p is the pore pressure, y the distance to the centroidal y-axis[1], and A is the cross-sectional area. For beam bending problems, cross-sectional area A may be assumed to be constant during the entire duration of the simulation.

To compute the value of M_p for the MPM, the following was done:

- The cross-sectional area A was computed based on the problem geometry;
- The particles' initial x-position was used to group the particles into beam cross-sections;
- The particles' initial y-position were used to find the distance to the centroidal y-axis.

This approach is valid due to the assumption of negligible deformation of A.

8.2.1 Model setup

The model setup aims to reproduce the settings provided in [109] within the possibilities of the Uintah computational framework [29]. The deviations from the setup in [109] are remarked when encountered.

Geometry. A cantilever poroelastic beam is modelled. The length of the beam $L = 1\,\text{m}$ in the x-direction, with height $h = 0.3\,\text{m}$ in the y-direction, and depth $d = 0.1\,\text{m}$ in the z-direction, as shown in Figure 8.7.

The end at $x = 0$ is fixed while the end at $x = L$ is free to bend and move. The flow boundary conditions are $\bar{p} = 0$ at $x = 0$ and $x = L$, that is, permeable beam ends. The beam sides in the longitudinal direction are set as impermeable with applied flux $\bar{q} = 0$. While the impermeable boundary conditions are perfectly imposed on $z = 0$ and $z = d$ because they are applied at the boundaries of the computational grid, the same is not true for the y boundaries. As discussed in Section 7.2 and illustrated in Figure 7.2, there will be minor inaccuracies in the pore pressure gradient due to the flow boundaries being imposed on cell boundaries.

[1] In [109], z is the vertical direction, not y.

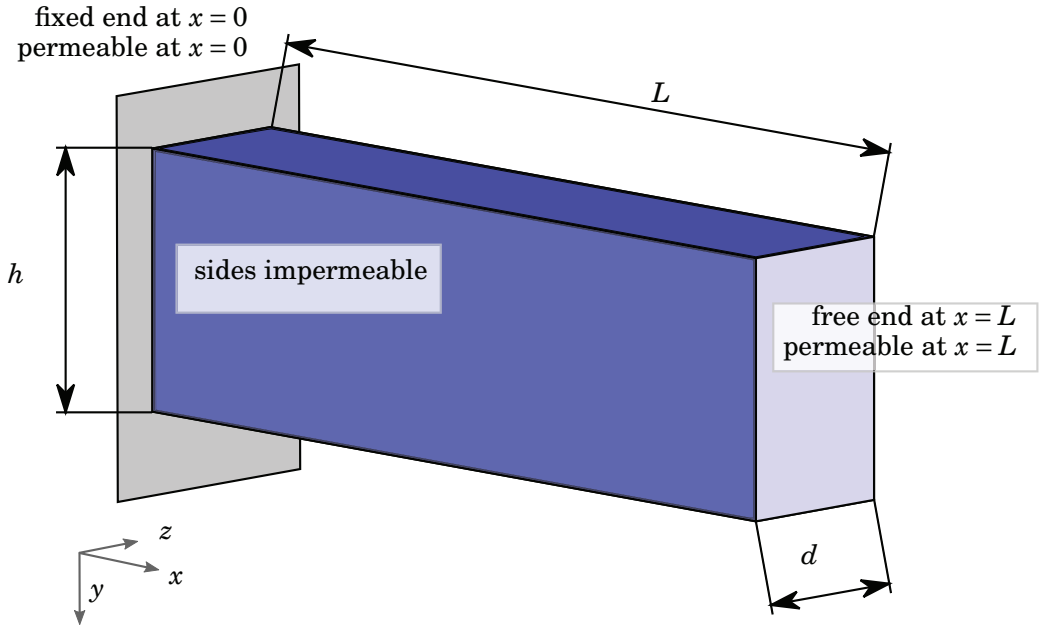


Figure 8.7: Geometry and boundary conditions of the poroelastic beam. The beam dimensions are $L = 1\,\text{m}$, $h = 0.3\,\text{m}$ and $d = 0.1\,\text{m}$.

Material properties. As with the one-dimensional consolidation problem in Section 8.1, the material is a fully-saturated, isotropic linear elastic material input parameters $K = 3.333333\,\text{MPa}$ and $G = 5\,\text{MPa}$.

The poroelastic properties are hydraulic conductivity $k = 10^{-4}\,\text{m/s}$, bulk modulus of water $K_w = 2.2\,\text{GPa}$, and initial porosity $n = 0.4$. The water density is $\rho_w = 1000\,\text{kg/m}^3$ and the saturated density $\rho = 2000\,\text{kg/m}^3$.

This material deviates from the material used in [109]. Li *et al.* [109] only allow diffusion in the axial direction and their material is transversely isotropic.

Loading conditions. First, we need to determine the size of applied distributed load w determined by the initial maximum normalised deflection $\Delta_{\max}$ shown in Figure 8.8 [109]. Using Equation (8.10) to solve for w, gives $w = 5.4\,\text{KN/m}$, which in turn gives the surface load

$$q = \frac{w}{b} = \frac{5.4\,\text{kN/m}}{0.1\,\text{m}} = 54\,\text{kPa}. \tag{8.12}$$

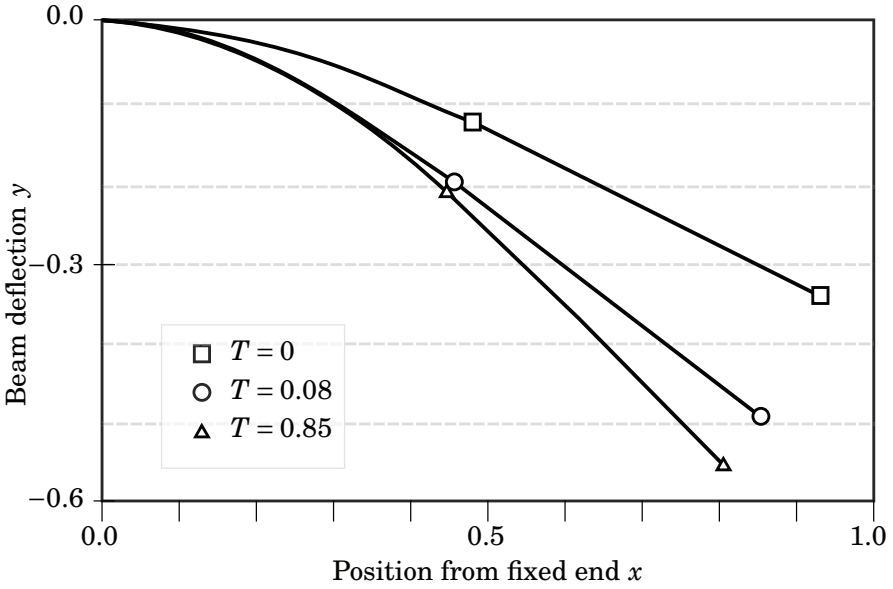


Figure 8.8: Deflection of the poroelastic cantilever beam at times $T = 0$, 0.08 and 0.85 with diffusion boundaries that are permeable at both ends. Fixed end is at $x = 0$ and free end is at $x = 1$. Digitally traced from Li, L *et al.* Large deflection analysis of poroelastic beams. *International Journal of Non-Linear Mechanics* **33,** 1–14 (1998), p.9, Figure 2a [109]. Copyright 1997, with permission by Elsevier Science Ltd.

The load q is applied along the top of the beam, starting from position $x = 0.05\,\text{m}$ from the fixed end running until the free end. The reason for this was numerical trouble with immediate large deflections of the beam if the beam had a distributed load along its entire length.

The load q is applied as immediately as possible in the quasi-static simulation, which in this case was at $t = 0.06\,\text{s}$. The quasi-static simulation is run to reproduce results in [109].

Discretisation. In order to compute the cross-sectional pore pressure moment M_p, which is the variable that is reported in [109], we need to properly catch the variation in pore pressure in the cross-section. Hence, the discretisation in the y-direction must be relatively fine, since the pore pressure is constant within each cell.

The computational grid is defined with $\Delta y = \Delta x = 0.015\,\text{m}$ and only 1

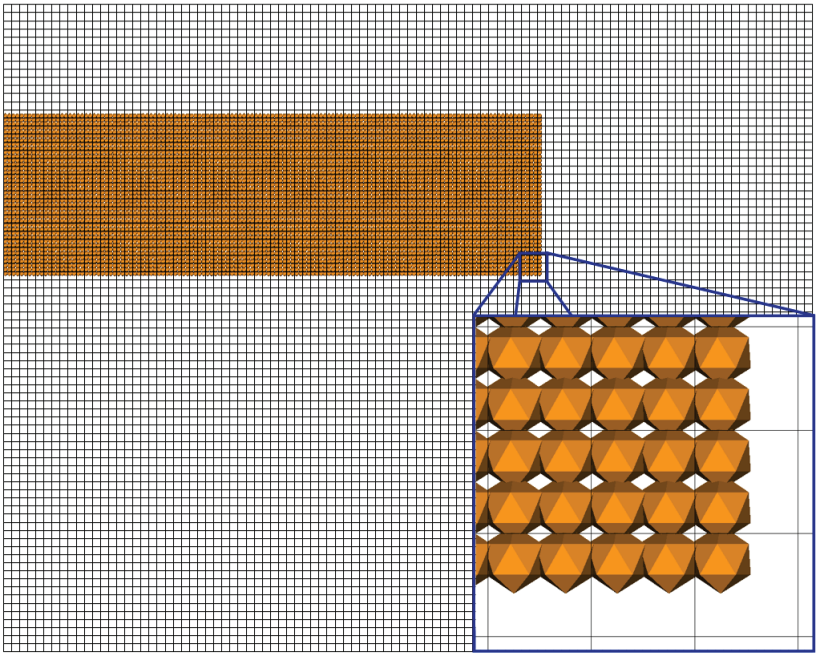
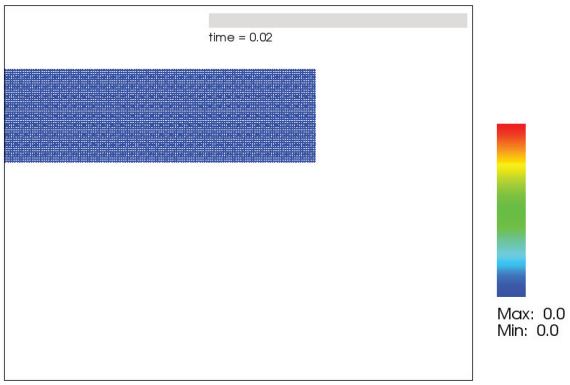


Figure 8.9: Discretisation of the poroelastic beam. Each cell has 2×2 particles.

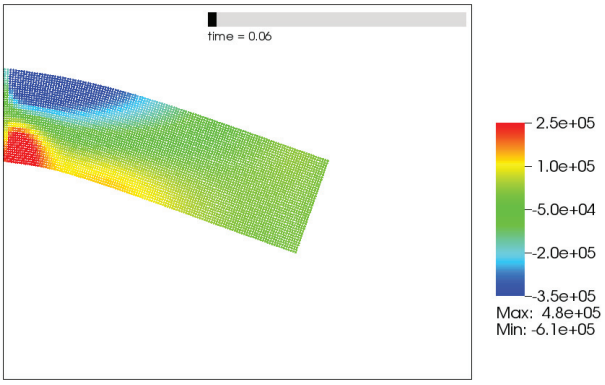
cell in the z-direction. With a mesh size of $y = 1.2\,\text{m}$, $x = 1.5\,\text{m}$ and $z = 0.1\,\text{m}$ this gives a total of 8000 cells, of which 1407 are cells that contains any beam material at initialization.

In each cell $2 \times 2 \times 1$ particles are initialised in x-, y-, and z-direction, respectively, see Figure 8.9. This gives 4 particles/cell. Not all cells were fully filled with particles at creation. A total of 5320 particles were created in the domain.

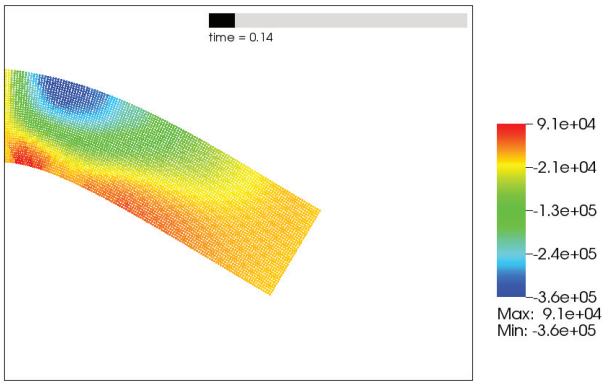
Size of time steps. The results of the one-dimensional consolidation problem presented in Section 8.1.3 showed that for equivalent times $T > 0.05$, a large time step of $\Delta t = 10^{-2}\,\text{s}$ produces accurate enough results. For the beam bending problem, we are most interested in results after $T > 0.05$. In addition, in order to minimize initial oscillations due to the abrupt change in loading, $\Delta t = 2 \cdot 10^{-2}$ was chosen.



(a) $t = 0.02\,\mathrm{s}$

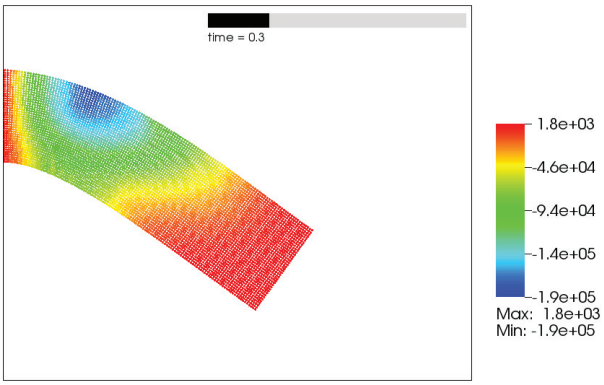


(b) $t = 0.06\,\mathrm{s}$

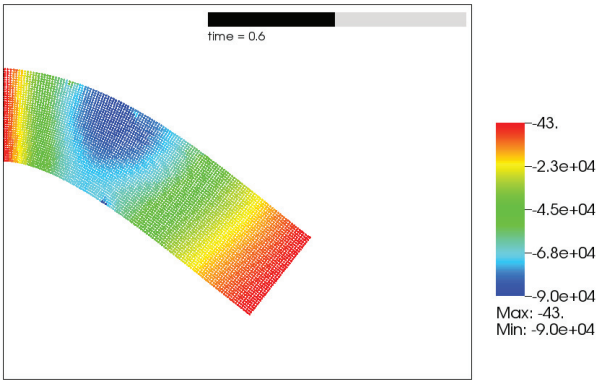


(c) $t = 0.14\,\mathrm{s}$

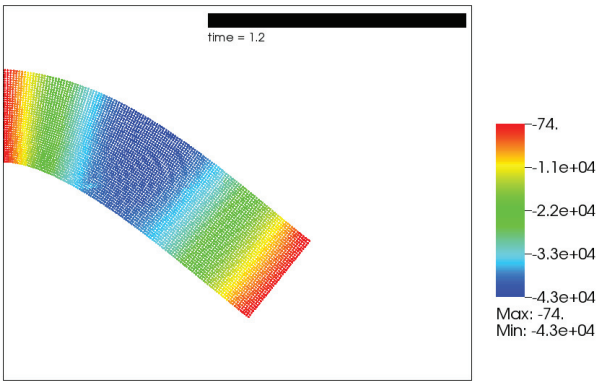
Figure 8.10: Pore pressure particle plots of bending poroelastic beam for times $t = 0.02$, 0.06, and 0.14 s. Values in Pa. Suction is negative, pressure is positive.



(a) $t = 0.3\,\mathrm{s}$



(b) $t = 0.6\,\mathrm{s}$



(c) $t = 1.2\,\mathrm{s}$

Figure 8.11: Pore pressure particle plots of bending poroelastic beam for times $t = 0.3$, 0.6, and 1.2 s. Values in Pa. Suction is negative, pressure is positive.

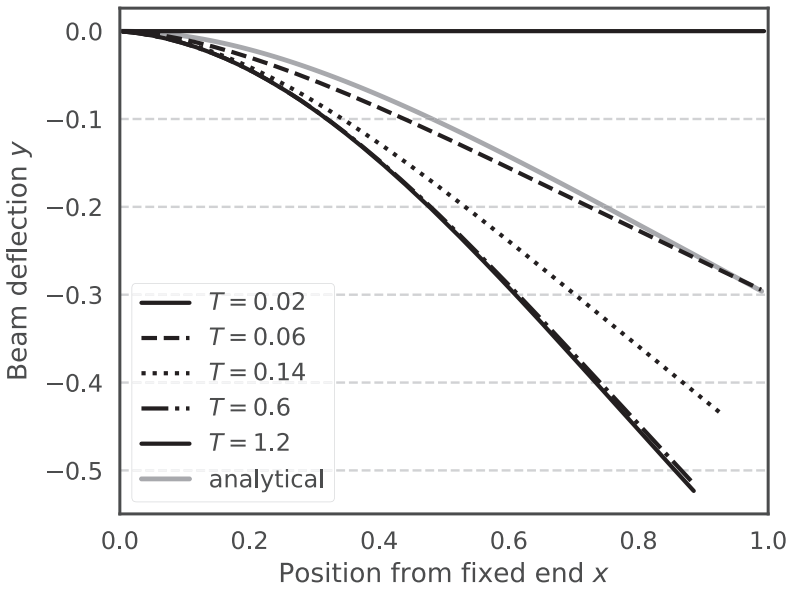


Figure 8.12: Deflection of the poroelastic cantilever beam at times $t = 0$ (no deflection), 0.1, 0.6 and 1.2 s (black lines) compared with small deflection theory (grey line). The deflection at $t = 0.6$ s is the maximum deflection obtained despite not all pore pressures have dissipated yet.

8.2.2 Results

Let us first inspect the pore pressure plot of the bending beam to see whether it is physically expected results, see Figures 8.10 and 8.11. We see that the bending induces pore pressures in the beam. There is suction in the upper part of the beam that experiences stretching, and there is pressure in the lower part of the beam that experiences compression, see Figure 8.10(b). This is physically viable. As the simulation progresses, we can observe the pore pressure dissipation. Towards the end of the simulation, there is no cross-sectional pore pressure moment because the pore pressures are equal per cross-section. Furthermore, we can clearly see the effect of the boundary conditions of the beam: the open ends and the closed sides, with a pore pressure distribution that resembles the pore pressure distribution of a one-dimensional two-way drainage consolidation problem. Therefore, the physics of the simulation makes sense.

Because the loading is applied in the negative Cartesian y-direction,

the beam is stretched. Therefore, the pore pressures in the beam at the end of the simulation is in suction.

The beam deflection curves at various time intervals are shown in Figure 8.12. The initial deflection of the beam should theoretically correspond to the linear elastic quasi-static solution of beam bending. As seen in Figure 8.12, there is a small deviation from the analytical solution, though only in the curvature. This may be due to the generated pore pressures which give a higher stiffness in that region of the beam. Nonetheless, the results from the initial load step give trust in the method in that the input of load and beam characteristics result in a predictable deflection.

After the initial deflection due to the immediately applied distributed load, we see an additional increase in deflection plotted in Figure 8.12. The load has already been applied and displacements have already resulted from that load, as discussed in the previous paragraph. Therefore, the additional deflection stems from the pore pressure dissipation. As pore pressures decrease, the beam will give a softer response. This shows that the pore pressures are able to influence the displacement response. This is in line with results obtained in Figure 8.8 [109].

Next, we look at Figure 8.14 to see and verify the evolution of the cross-sectional pore pressure moment M_p. The aim is to compare the results obtained from the method with results obtained by Li *et al.* [109], see Figure 8.13 [109]. The similarities between the two results depicted in Figures 8.13 and 8.14 are that M_p is fairly high close to the fixed end at $x = 0$ and quickly diminishing along the beam towards the free end at $x = L$. Additionally, M_p diminishes with time, corresponding to the pore pressure redistribution observed in Figures 8.10 and 8.11. We observe that the proposed method is able to generate pore pressures based on displacements and to dissipate the pore pressures while maintaining the largely deformed state.

There are, however, differences in the results depicted in Figures 8.13 and 8.14. The finite difference solution in Figure 8.13 converges towards a symmetric distribution of pore pressure moment along the beam, whereas the proposed method does not. The maximum pore pressure moment of the proposed method shown in Figure 8.14 does move towards the centre of the beam, but never reaches symmetry. There may be several reasons for this. The beam modelled with the proposed method allows transverse diffusion inside the beam, which allows the pore pressures to redistribute in the transverse direction. The pore pressure moment diminishes when the pore pressures above and below the beam centreline become equal. The transverse pore pressure gradient may be larger than the axial gradient

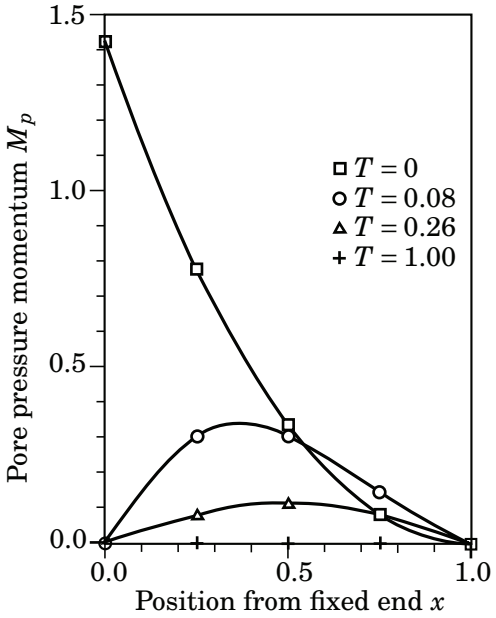


Figure 8.13: Normalised cross-sectional pore pressure moment M_p along poroelastic cantilever beam at times $T = 0$, 0.08, 0.26 and 1 with diffusion boundaries that are permeable at both ends. Fixed end is at $x = 0$ and free end is at $x = 1$. Digitally traced from Li, L *et al.* Large deflection analysis of poroelastic beams. *International Journal of Non-Linear Mechanics* **33,** 1–14 (1998), p.9, Figure 3a [109]. Copyright 1997, with permission by Elsevier Science Ltd.

due to the difference between suction and pressure. In turn, this may lead to faster pore pressure dissipation in the transverse direction than in the axial direction, quickly removing pore pressure moment as a result.

8.2.3 Summary

This section demonstrated that the proposed method is able to

1. model large deformation that also includes particles crossing cells;
2. couple flow and mechanics;
3. produce results that are similar to Li *et al.* [109].

Additionally, the moving boundary conditions seem to be implemented correctly.

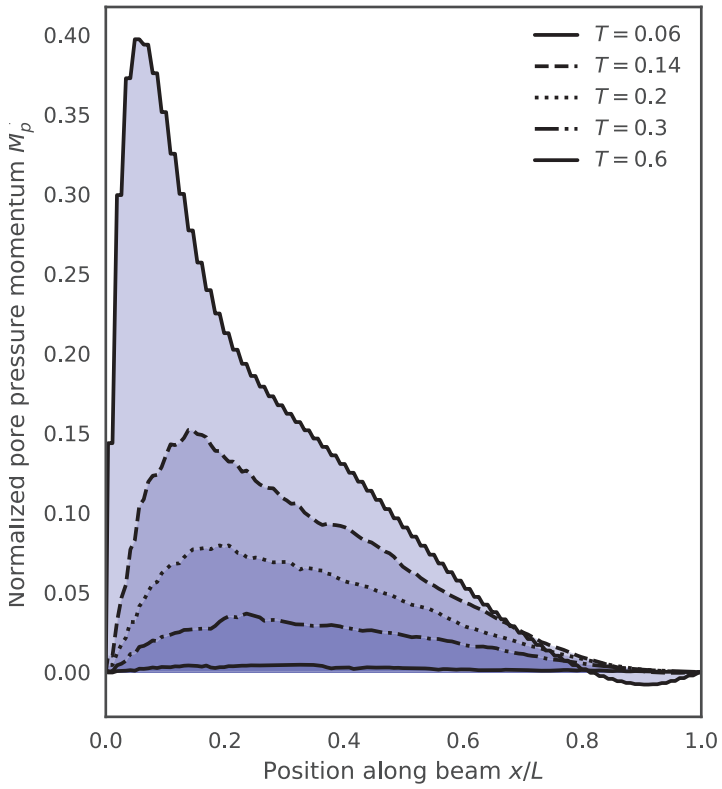


Figure 8.14: Normalised cross-sectional pore pressure moment M_p along the poro-elastic cantilever beam at times $t = 0$, 0.06, 0.2 and 0.8 s with diffusion boundaries that are permeable at both ends. Fixed end is at $x = 0$ and free end is at $x = 1$. Results obtained by the implicit iteratively coupled MPM-FVM method.

CHAPTER 9

CONCLUSION

This work has been focused on developing and implementing the material point method (MPM) to model the coupling between mechanical displacements and pore pressure generation and dissipation under large deformations of saturated porous media.

There exist several formulations for MPM with implicit time integration and several formulations of coupled MPM with explicit time integration. However, the combination of a coupled MPM approach with implicit time integration is new. The iteratively coupled implicit dynamic MPM-FVM method presented herein was developed with the goal of modelling large deformation problems with coupled hydromechanics for geotechnical engineering problems, but the method can be applied in other problems of flow through porous media.

Explicit velocity formulation. First, the explicit MPM solver of Uintah [29] was extended with the explicit velocity formulation originally adapted to MPM by Jassim *et al.* [9]. The velocity formulation assumes that solid and water accelerations are not equal. However, if that assumption is true, then the fluid mass is not conserved.

The implementation worked for the analytical one-dimensional consolidation problem. However, for the dimensionless time $T < 0.2$ the explicit formulation produced large oscillations, in line with existing knowledge. The explicit method needed time steps around 10^{-5} to give good results

for a hydraulic coefficient of $k = 10^{-3}$. With critical time step size being influenced by permeability, this would mean that for solving problems with a less permeable material, the time step size must be even smaller. As a consequence, long-term simulations with low-permeable materials would be very time-consuming.

To circumvent the problems mentioned here, there exist several approaches to add to the explicit formulation, such as mass scaling to reduce time step size and local damping to reduce oscillations. Mass scaling is most appropriate for almost quasi-static conditions [68]. None of those approaches was implemented in this work as it was beyond the scope.

Iteratively coupled implicit formulation. This thesis proposes a dynamic approach based on the $\mathbf{u}$-p formulation with implicit time integration. The proposed method uses an iteratively coupled approach called the fixed-stress operator split [24, 25, 90] to solve the fully implicit coupled system of equations. The iteratively coupled approach allows displacements and pore pressures to be solved by separate solvers. Previously, FEM has been coupled with FVM. In this thesis, the proposed method couples MPM with FVM. The coupling approach between MPM and FVM supports that there are many advantages originating from the similarities of FEM and MPM and applying methods initially derived for FEM to MPM.

The proposed method was verified by benchmarks. The first benchmark was the same analytical one-dimensional consolidation problem that was used to verify the explicit implementation. The method could properly solve the mathematical equation with time steps of the order of 10^{-3} s to 10^{-2} s for a hydraulic coefficient of $k = 10^{-3}$ m/s. Additionally, there were no oscillations for the quasi-static approach, but there were small oscillations for dimensionless time $T < 0.2$ when inertial terms were included.

The second benchmark was simulating the bending of a poroelastic beam. The proposed method was able to model large deformations while simultaneously generating and dissipating pore pressures. The coupled effect between flow and mechanics could be seen from the pore pressures being generated from the bending moment, and the beam deflection increased with time when pore pressure diminished. The results were along the same lines as published results on a similar problem.

Implications of research. The benefit of the proposed method is twofold. First, the method could be a valuable approach when modelling porous flow problems. Examples of such flow problems would be earthquakes,

slope failure, dam embankments, pile driving. Since the proposed method uses implicit time integration, it can use larger time steps than explicit approaches. Consequently, the proposed method is more suited for long-term simulations.

Second, the proposed method aids the process of extending a single-phase implicit MPM code to be able to model coupled flow. With the iteratively coupled approach, minor changes to the mechanical solver are necessary and a flow solver must be added. The flow component may draw on already existing features with the mechanical solver, such as the background mesh. Using a Cartesian background mesh also allowed for using the simplest flux scheme in FVM, namely the two-point flux approximation method.

Summary of contributions. The main contribution of this thesis is the development and implementation of an iteratively coupled implicit dynamic MPM-FVM method. The proposed method has been verified to work with consolidation and large deformations. Other minor contributions in this thesis are

- the extension of a single-phase explicit MPM code to coupled hydromechanics;
- development of using the material volume of FVM cells instead of the geometrical cell volume in order to account for the changing material volume due to particle migration between cells;
- development of updated FVM boundary conditions based on the current material body configuration;
- simulation and analysis of the bending of a poroelastic beam.

One never notices what has been done;
one can only see what remains to be done.

Marie Curie

CHAPTER 10

RECOMMODATIONS FOR FURTHER WORK

The areas of further research can be divided into two parts: further refinement of the proposed method itself, and extensions of the proposed method to model geotechnical problems. The goal of refining the proposed method would be to obtain a more accurate method than the one presented in this thesis, and may be accomplished by including the following features:

Advanced moving boundary conditions The moving flow boundary conditions in this work are following the cell borders. To increase accuracy, the boundary conditions may be attempted to follow the surface of the material point body.

Advanced flux schemes The simple two-point flux approximation scheme has been used in this work. There exist several other schemes that might be beneficial in order to model flow diagonally across the mesh.

Shape functions of higher order In this work, the standard linear shape functions were used for the MPM interpolation. However, the research front in MPM uses more advanced shape functions, either by using B-splines or other higher order shape functions, but also using GIMP. Higher order shape functions for the displacements, allow for potentially using shape functions for the cell pore pressures too.

For adapting the proposed method for solving geotechnical engineering problems, both advanced material models and the ability to model unsaturated soils are needed:

Advanced material models. The verification of the iteratively coupled implicit dynamic MPM-FVM method used an isotropic elastic material model. Advanced material models should be implemented to investigate the proposed method's usability within the field of geotechnical engineering.

Extension to unsaturated soils. Iteratively coupled methods for FEM and FVM are able to model unsaturated soils. The proposed method herein should be extended to allow for unsaturated soils.

Introduce contact formulations. In geotechnical engineering, many problems include soil-water-structure interaction. For the proposed method to be of practical interest, contact formulations should be included.

REFERENCES

1. Darby, SE, Rinaldi, M & Dapporto, S. Coupled simulations of fluvial erosion and mass wasting for cohesive river banks. *Journal of Geophysical Research* **112,** F03022 (2007).
2. Ceccato, F & Simonini, P. Numerical study of partially drained penetration and pore pressure dissipation in piezocone test. *Acta Geotechnica* **12,** 195–209 (2017).
3. Cowin, SC. Bone poroelasticity. *Journal of biomechanics* **32,** 217–38 (1999).
4. Zhang, J, Datta, AK & Mukherjee, S. Transport processes and large deformation during baking of bread. *AIChE Journal* **51,** 2569–2580 (2005).
5. Więckowski, Z. The material point method in large strain engineering problems. *Computer Methods in Applied Mechanics and Engineering* **193,** 4417–4438 (2004).
6. Ma, S & Zhang, X. in *Computational Mechanics* 156–166 (Springer Berlin Heidelberg, 2009).
7. Sołowski, WT & Sloan, SW. Evaluation of material point method for use in geotechnics. *International Journal for Numerical and Analytical Methods in Geomechanics* **39,** 685–701 (2015).
8. Soga, K, Alonso, E, Yerro, A, Kumar, K & Bandara, S. Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method. *Géotechnique* **66,** 248–273 (2016).
9. Jassim, I, Stolle, D & Vermeer, P. Two-phase dynamic analysis by material point method. *International Journal for Numerical and Analytical Methods in Geomechanics* **37,** 2502–2522 (2013).
10. Yerro, A, Alonso, EE & Pinyol, NM. The material point method for unsaturated soils. *Géotechnique* **65,** 201–217 (2015).

11. Zhang, DZ, Zou, Q, Vanderheyden, WB & Ma, X. Material point method applied to multiphase flows. *Journal of Computational Physics* **227,** 3159–3173 (2008).
12. Abe, K, Soga, K, Bandara, S, Asce, M & Bandara, S. Material Point Method for Coupled Hydromechanical Problems. *Journal of Geotechnical and Geoenvironmental Engineering* **140,** 04013033 (2014).
13. Bandara, S & Soga, K. Coupling of soil deformation and pore fluid flow using material point method. *Computers and Geotechnics* **63,** 199–214 (2015).
14. Higo, Y, Oka, F, Kimoto, S, Morinaka, Y, Goto, Y & Chen, Z. A Coupled MPM-FDM Analysis Method for Multi-Phase Elasto-Plastic Soils. *Soils and Foundations* **50,** 515–532 (2010).
15. Sheng, D & Sloan, SW. Time stepping schemes for coupled displacement and pore pressure analysis. *Computational Mechanics* **31,** 122–134 (2003).
16. Wang, B, Vardon, PJ, Hicks, MA & Chen, Z. Development of an implicit material point method for geotechnical applications. *Computers and Geotechnics* **71,** 159–167 (2016).
17. Love, E & Sulsky, DL. An unconditionally stable, energy–momentum consistent implementation of the material-point method. *Computer methods in applied mechanics and engineering* **195,** 3903–3925 (2006).
18. Guilkey, JE & Weiss, JA. Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. *International Journal for Numerical Methods in Engineering* **57,** 1323–1338 (2003).
19. Sulsky, D & Kaul, A. Implicit dynamics in the material-point method. *Computer Methods in Applied Mechanics and Engineering* **193,** 1137–1170 (2004).
20. Beuth, L. *Formulation and Application of a Quasi-Static Material Point Method* PhD thesis (Universität Stuttgart, 2012).
21. Fung, LSK, Buchanan, L & Wan, RG. Coupled Geomechanical-thermal Simulation for Deforming Heavy-oil Reservoirs. *Journal of Canadian Petroleum Technology* **33,** 22–28 (1994).
22. Settari, A & Mourits, FM. A Coupled Reservoir and Geomechanical Simulation System. *SPE Journal* **3,** 219–226 (1998).

23. Minkoff, SE, Stone, C, Bryant, S, Peszynska, M & Wheeler, MF. Coupled fluid flow and geomechanical deformation modeling. *Journal of Petroleum Science and Engineering* **38,** 37–56 (2003).

24. Kim, J. *Sequential Methods for Coupled Geomechanics and Multiphase Flow* PhD thesis (Stanford, 2010).

25. Mikelić, A & Wheeler, MF. Convergence of iterative coupling for coupled flow and geomechanics. *Computational Geosciences* **17,** 455–461 (2013).

26. Mikelić, A, Wang, B & Wheeler, MF. Numerical convergence study of iterative coupling for coupled flow and geomechanics. *Computational Geosciences* **18,** 325–341 (2014).

27. Dean, RH, Gai, X, Stone, CM & Minkoff, SE. A Comparison of Techniques for Coupling Porous Flow and Geomechanics. *SPE Journal* **11,** 132–140 (2006).

28. Phillips, PJ & Wheeler, MF. A coupling of mixed and continuous Galerkin finite element methods for poroelasticity I: the continuous in time case. *Computational Geosciences* **11,** 131–144 (2007).

29. C-SAFE & SCI. *Uintah v1.6.0* 2017.

30. Potts, DM. *Finite element analysis in geotechnical engineering* (Thomas Telford Publishing, 1999).

31. Sheng, D, Nazem, M & Carter, JP. Some computational aspects for solving deep penetration problems in geomechanics. *Computational Mechanics* **44,** 549–561 (2009).

32. Monaghan, JJ. An introduction to SPH. *Computer Physics Communications* **48,** 89–96 (1988).

33. Evans, MW & Harlow, FH. *The particle-in-cell method for hydrodynamic calculations* tech. rep. (Los Alamos Scientific Laboratory, Los Alamos, 1957).

34. Brackbill, J & Ruppel, H. FLIP: A Method for Adaptively Zoned, Particle-in-Cell Calculations of Fluid Flows in Two Dimensions. *Journal of Computational Physics* **65,** 314–343 (1986).

35. Sulsky, D & Brackbill, J. A numerical method for suspension flow. *Journal of Computational Physics* **96,** 339–368 (1991).

36. Sulsky, D, Chen, Z & Schreyer, HL. A particle method for history-dependent materials. *Computer methods in applied mechanics and engineering* **118,** 179–196 (1994).

37. Sulsky, D, Zhou, SJ & Schreyer, HL. Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications* **87,** 236–252 (1995).

38. Li, S & Liu, WK. Meshfree and particle methods and their applications. *Applied Mechanics Reviews* **55,** 1–34 (2002).

39. Bardenhagen, SG, Brackbill, J & Sulsky, D. The material-point method for granular materials. *Computer methods in applied mechanics and engineering* **187,** 529–541 (2000).

40. Więckowski, Z, Youn, SK & Yeon, JH. A particle-in-cell solution to the silo discharging problem. *International Journal for Numerical Methods in Engineering* **45,** 1203–1225 (1999).

41. Guilkey, JE, Harman, T & Banerjee, B. An Eulerian-Lagrangian approach for simulating explosions of energetic devices. *Computers and Structures* **85,** 660–674 (2007).

42. Sulsky, D, Schreyer, HL, Peterson, K, Kwok, R & Coon, M. Using the material-point method to model sea ice dynamics. *Journal of Geophysical Research: Oceans* **112,** C02S90 (2007).

43. Sandoval Murillo, J, Osen, R, Hiermaier, S & Ganzenmüller, G. Towards understanding the mechanism of fibrous texture formation during high-moisture extrusion of meat substitutes. *Journal of Food Engineering* **242,** 8–20 (2019).

44. Ionescu, I, Guilkey, JE, Berzins, M, Kirby, RM & Weiss, JA. Simulation of Soft Tissue Failure Using the Material Point Method. *Journal of Biomechanical Engineering* **128,** 917–924 (2006).

45. Gan, Y & Chen, Z. A study of the zona piercing process in piezodriven intracytoplasmic sperm injection. *Journal of Applied Physics* **104,** 044702 (2008).

46. Daviet, G & Bertails-Descoubes, F. A Semi-Implicit Material Point Method for the Continuum Simulation of Granular Materials. *ACM Transactions on Graphics* **35,** 102:1–102:13 (2016).

47. Stomakhin, A, Schroeder, C, Chai, L, Teran, J & Selle, A. A material point method for snow simulation. *ACM Transactions on Graphics* **32,** 102:1–102:9 (2013).

48. Ma, J, Wang, D & Randolph, M. A new contact algorithm in the material point method for geotechnical simulations. *International Journal for Numerical and Analytical Methods in Geomechanics* **38,** 1197–1210 (2014).

49. Chen, ZPZP, Zhang, X, Qiu, XM & Liu, Y. A frictional contact algorithm for implicit material point method. *Computer Methods in Applied Mechanics and Engineering* **321,** 124–144 (2017).

50. Yerro, A, Rohe, A & Soga, K. Modelling internal erosion with the material point method. *Procedia Engineering* **175,** 365–372 (2017).

51. Belytschko, T, Liu, WK, Moran, B & Elkhodary, KI. *Nonlinear Finite Elements for Continua and Structures* 2nd ed. (Wiley, 2014).

52. Nairn, JA. Material Point Method Calculations with Explicit Cracks. *CMES - Computer Modeling in Engineering and Sciences* **4,** 649–663 (2003).

53. Steffen, M, Kirby, RM & Berzins, M. Decoupling and balancing of space and time errors in the material point method (MPM). *International Journal for Numerical Methods in Engineering* **82,** 1207–1243 (2009).

54. Bardenhagen, SG & Kober, EM. The generalized interpolation material point method. *CMES - Computer Modeling in Engineering and Sciences* **5,** 477–495 (2004).

55. Guilkey, JE & Weiss, JA. *An implicit time integration strategy for use with the material point method* in *Computational Fluid and Solid Mechanics* (2001), 216–219.

56. Iaconeta, I, Larese, A, Rossi, R & Oñate, E. An Implicit Material Point Method Applied to Granular Flows. *Procedia Engineering* **175,** 226–232 (2017).

57. Beuth, L, Benz, T, Vermeer, PA & Więckowski, Z. Large deformation analysis using a quasi-static material point method. *Journal of Theoretical and Applied Mechanics* **38,** 45–60 (2008).

58. Charlton, TJ, Coombs, WM & Augarde, CE. iGIMP: An implicit generalised interpolation material point method for large deformations. *Computers and Structures* **190,** 108–125 (2017).

59. Cummins, SJ & Brackbill, JU. An implicit particle-in-cell method for granular materials. *Journal of Computational Physics* **180,** 506–548 (2002).

60. Nair, A & Roy, S. Implicit Time Integration in the Generalized Interpolation Material Point Method for Finite Deformation Hyperelasticity. *Mechanics of Advanced Materials and Structures* **19,** 465–473 (2012).

61. Sanchez, J, Schreyer, H, Sulsky, D & Wallstedt, P. Solving quasi-static equations with the material-point method. *International Journal for Numerical Methods in Engineering* **103,** 60–78 (2015).

62. Bürg, M, Lim, LJ & Brinkgreve, R. Application of a second-order implicit material point method. *Procedia Engineering* **175,** 279–286 (2017).

63. Zienkiewicz, C, Chang, CT & Bettess, P. Drained, undrained, consolidating and dynamic behaviour assumptions in soils. *Géotechnique* **30,** 385–395 (1981).

64. Zienkiewicz, OC & Shiomi, T. Dynamic Behaviour of Saturated Porous Media; The Generalized Biot Formulation and Its Numerical Solution. *International Journal for Numerical and Analytical Methods in Geomechanics* **8,** 71–96 (1984).

65. Zhang, H, Wang, K & Chen, Z. Material point method for dynamic analysis of saturated porous media under external contact/impact of solid bodies. *Computer Methods in Applied Mechanics and Engineering* **198,** 1456–1472 (2009).

66. Higo, Y, Kimoto, S, Oka, F, Morinaka, Y, Goto, Y & Chen, Z. *An MPM-FDM coupled simulation method for analyzing partially-saturated soil responses* in *Prediction and Simulation Methods for Geohazard Mitigation* (2009), 219–225.

67. Mackenzie-Helnwein, P, Arduino, P, Shin, W, Moore, JA & Miller, GR. Modeling strategies for multiphase drag interactions using the material point method. *International Journal for Numerical Methods in Engineering* **83,** 295–322 (2010).

68. Zabala, F & Alonso, E. Progressive failure of Aznalcóllar dam using the material point method. *Géotechnique* **61,** 795–808 (2011).

69. Zheng, Y, Gao, F, Zhang, H, Liu, M & Lu, M. Improved convected particle domain interpolation method for coupled dynamic analysis of fully saturated porous media involving large deformation. *Computer Methods in Applied Mechanics and Engineering* **257,** 150–163 (2013).

70. Liu, C, Sun, Q, Jin, F & Zhou, GGD. A fully coupled hydro-mechanical material point method for saturated dense granular materials. *Powder Technology* **314,** 110–120 (2017).

71. Yang, Y, Sun, P & Chen, Z. Combined MPM-DEM for Simulating the Interaction Between Solid Elements and Fluid Particles. *Communications in Computational Physics* **21,** 1258–1281 (2017).

72. Higo, Y, Nishimura, D & Oka, F. *Dynamic analysis of unsaturated embankment considering the seepage flow by a GIMP-FDM coupled method* in *Computer Methods and Recent Advances in Geomechanics* (eds Oka, F, Murakami, A, Uzuoka, R & Kimoto, S) (CRC Press, 2014), 1762–1766.

73. Steffen, M, Wallstedt, P, Guilkey, J, Kirby, RM & Berzins, M. Examination and Analysis of Implementation Choices within the Material Point Method (MPM). *CMES - Computer Modeling in Engineering and Sciences* **31,** 107–127 (2008).

74. Sadeghirad, A, Brannon, RM & Burghardt, J. A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations. *Internation journal for numerical methods in engineering* **86,** 1435–1456 (2011).

75. Steffen, M, Kirby, RM & Berzins, M. Analysis and reduction of quadrature errors in the material point method (MPM). *International Journal for Numerical Methods in Engineering* **76,** 922–948 (2008).

76. Gan, Y, Sun, Z, Chen, Z, Zhang, X & Liu, Y. Enhancement of the material point method using B-spline basis functions. *International Journal for Numerical Methods in Engineering* **113,** 411–431 (2018).

77. Al-Kafaji, IKJ. *Formulation of a Dynamic Material Point Method (MPM) for Geomechanical Problems* PhD thesis (Universität Stuttgart, 2013).

78. Coombs, WM, Charlton, TJ, Cortis, M & Augarde, CE. Overcoming volumetric locking in material point methods. *Computer Methods in Applied Mechanics and Engineering* **333,** 1–21 (2018).

79. Vermeer, PA & Verruijt, A. An accuracy condition for consolidation by finite elements. *International Journal for Numerical and Analytical Methods in Geomechanics* **5,** 1–14 (1981).

80. Kularathna, S & Soga, K. Projection method in material point method for modeling incompressible materials. *Procedia Engineering* **175,** 57–64 (2017).

81. Longuemare, P, Mainguy, M, Lemonnier, P, Onaisi, A, Gérard, C & Koutsabeloulis, N. Geomechanics in Reservoir Simulation: Overview of Coupling Methods and Field Case Study. *Oil & Gas Science and Technology – Rev. IFP* **57,** 471–483 (2002).

82. Park, KC. Stabilization of partitioned solution procedure for pore fluid-soil interaction analysis. *International Journal for Numerical Methods in Engineering* **19,** 1669–1673 (1983).

83. Felippa, CA & Park, KC. Staggered transient analysis procedure for coupled mechanical systems: Formulation. *Computer Methods in Applied Mechanics and Engineering* **24,** 61–111 (1980).

84. Prevost, JH. Partitioned solution procedure for simultaneous integration of coupled-field problems. *Communications in Numerical Methods in Engineering* **13,** 239–247 (1997).

85. Idelsohn, SR & Oñate, E. Finite volumes and finite elements: two 'good friends'. *International Journal for Numerical Methods in Engineering* **37,** 3323–3341 (1994).

86. Jha, B & Juanes, R. A locally conservative finite element framework for the simulation of coupled flow and reservoir geomechanics. *Acta Geotechnica* **2,** 139–153 (2007).

87. Kim, J, Tchelepi, H & Juanes, R. Stability and convergence of sequential methods for coupled flow and geomechanics: Fixed-stress and fixed-strain splits. *Computer Methods in Applied Mechanics and Engineering* **200,** 1591–1606 (2011).

88. Jha, B & Juanes, R. Coupled multiphase flow and poromechanics: A computational model of pore pressure effects on fault slip and earthquake triggering. *Water Resources Research* **50,** 3776–3808 (2014).

89. Asadi, R, Ataie-Ashtiani, B & Simmons, CT. Finite volume coupling strategies for the solution of a Biot consolidation model. *Computers and Geotechnics* **55,** 494–505 (2014).

90. White, JA, Castelletto, N & Tchelepi, HA. Block-partitioned solvers for coupled poromechanics: A unified framework. *Computer Methods in Applied Mechanics and Engineering* **303,** 55–74 (2016).

91. Donea, J, Huerta, A, Ponthot, JP & Rodríguez-Ferran, A. *Arbitrary Lagrangian-Eulerian Methods* in *Encyclopedia of Computational Mechanics. Volume 1: Fundamentals* (eds Stein, E, de Borst, R & Hughes, TJR) (John Wiley & Sons, 2004), 413–437.

92. Green, AE & Naghdi, PM. A Theory of Mixtures. *Archive for Rational Mechanics and Analysis* **24,** 243–263 (1967).

93. Clayton, C, Steinhagen, HM, Powrie, W, Terzaghi, K & Skempton, AW. Terzaghi's theory of consolidation, and the discovery of effective stress (Compiled from the work of K. Terzaghi and A. W. Skempton). *Proceedings of the Institution of Civil Engineers - Geotechnical Engineering* **113,** 191–205 (1995).

94. De Boer, R & Ehlers, W. The development of the concept of effective stresses. *Acta Mechanica* **83,** 77–92 (1990).

95. Borja, RI & Alarcón, E. A mathematical framework for finite strain elastoplastic consolidation. Part 1: Balance laws, variational formulation, and linearization. *Computer Methods in Applied Mechanics and Engineering* **122,** 145–171 (1995).

96. Coussy, O. *Poromechanics* (Wiley, Chichester, 2004).

97. Zienkiewicz, OC, Chan, AHC, Pastor, M, Paul, DK & Shiomi, T. Static and dynamic behaviour of soils: a rational approach to quantitive solutions. I. Fully saturated problems. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* **429,** 285–309 (1990).

98. Burgess, D, Sulsky, D & Brackbill, JU. Mass Matrix Formulation of the FLIP Particle-in-Cell Method. *Journal of Computational Physics* **103,** 1–15 (1992).

99. Cook, RT, Malkus, DS, Plesha, ME & Witt, RJ. *Concepts and applications of finite element analysis* 4th ed. (John Wiley & Sons, 2002).

100. Wang, B. *Slope failure analysis using the material point method* PhD thesis (Technische Universiteit Delft, 2016).

101. Courant, R, Friedrichs, K & Lewy, H. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen* **100,** 32–74 (1928).

102. Newmark, N. A Method of Computation for Structural Dynamics. *Journal of the Engineering Mechanics Division* **85,** 67–94 (1959).

103. Oñate, E & Idelsohn, SR. *A comparison between finite element and finite volume methods in CFD* in *Computational Fluid Dynamics* (eds Hirsch, C, Périaux, J & Kordulla, W) (Brussels, Belgium, 1992), 93–100.

104. Li, ZC & Wang, S. The finite volume method and application in combinations. *Journal of Computational and Applied Mathematics* **106,** 21–53 (1999).

105. Simon, BR, Wu, JSS, Zienkiewicz, OC & Paul, DK. Evaluation of u-w and u-π finite element methods for the dynamic response of saturated porous media using one-dimensional models. *International Journal for Numerical and Analytical Methods in Geomechanics* **10,** 461–482 (1986).

106. Hughes, TJR. A note on the stability of Newmark's algorithm in nonlinear structural dynamics. *International Journal for Numerical Methods in Engineering* **11,** 383–386 (1977).

107. Hughes, TJR. Stability, convergence and growth and decay of energy of the average acceleration method in nonlinear structural dynamics. *Computers & Structures* **6,** 313–324 (1976).

108. Falgout, RD & Yang, UM. *hypre: A Library of High Performance Preconditioners* in *Computational Science — ICCS 2002* (eds Sloot, PMA, Hoekstra, AG, Tan, CJK & Dongarra, JJ) (Springer Berlin Heidelberg, Berlin, Heidelberg, 2002), 632–641.

109. Li, L, Schulgasser, K & Cederbaum, G. Large deflection analysis of poroelastic beams. *International Journal of Non-Linear Mechanics* **33,** 1–14 (1998).

Symbols

$\mathbf{a}_\mathrm{s}$	acceleration of solid phase
$\mathbf{a}_\mathrm{w}$	acceleration of water phase
$\mathbf{a}$	(solid) acceleration vector
$\mathbf{a}_h$	approximate (solid) acceleration vector
$\mathbf{a}_i$	(solid) acceleration vector evaluated at node i
$\mathbf{a}_p$	(solid) acceleration vector evaluated at $\mathbf{x}_p$
$\mathbf{b}$	solution vector; body force
$\mathbf{b}_h$	approximate body force
$\mathbf{b}_p$	body force evaluated at $\mathbf{x}_p$
c	dilational wave speed
c_v	coefficient of consolidation
d	number of dimensions, 1, 2 or 3; depth of soil column; depth of beam cross-section
e	void ratio, $V_\mathrm{w}/V_\mathrm{s}$ for saturated soils
f	a function; element face
$\mathbf{f}_p^\mathrm{trac}$	boundary particle traction force
g	magnitude of the acceleration of gravity
h	height of soil column; height of beam cross-section
Δh	cell size closest to draining boundary
k	hydraulic conductivity; as superscript: variable evaluated at iteration k
L	length of beam
m	total mass of mixture; fluid mass (Chapter 2)
m_i	diagonally lumped mass matrix
m_{ij}	consistent mass matrix
$\mathbf{m}_\alpha$	diagonally lumped mass matrix for phase α
$\bar{\mathbf{m}}_\alpha$	average diagonally lumped mass matrix for phase α

n	porosity, V_w/V for saturated soils; as superscript: variable evaluated at time step n
n_α	volume proportion of phase α to total volume
n_s	volume proportion of solids to total volume, $1-n$ for saturated soils
$\mathbf{n}$	surface normal vector
$\mathbf{n}_f$	surface normal vector of element face f
p	pore pressure
p_0	initial pore pressure
$\bar{p}$	prescribed pore pressure
$\bar{p}^n$	cell pore pressure average at time step n
$\dot{p}$	pore pressure rate
p_E	pore pressure in east cell E
p_f	pore pressure in neighbour cell that shares face f
p_h	approximate pore pressure
$\mathbf{p}_i^n$	nodal momentum in node i at time step n
p_P	pore pressure in centre cell P
$p(t)$	prescribed pressure traction in Uintah
q	surface load
q_e	flux on east face of cell
$\mathbf{q}$	specific discharge, Darcy velocity, $n(\mathbf{w}-\mathbf{v})$
$\hat{\mathbf{q}}$	pore pressure gradient term of flux $\mathbf{q}$
$\mathbf{r}$	residual
$r_{\text{mass},\,n}$	residual of mass for element n
$\mathbf{r}_{\text{mom}}$	residual of momentum
t	time; current time
t_0	intial time
t_n	time at time step n
Δt	time step size
Δt_c	critical time step size
Δt_w	critical time step size for water
$\mathbf{u}$	displacement vector
$\mathbf{u}_0$	initial displacement vector
$\bar{\mathbf{u}}$	prescribed displacement vector
$\dot{\mathbf{u}}$	velocity $\mathbf{v}$
$\ddot{\mathbf{u}}$	acceleration $\mathbf{a}$
$\mathbf{u}_h$	approximate solution of displacements
$\mathbf{u}_i$	nodal displacements

$\mathbf{v}$	(solid) velocity vector
$\mathbf{v}_0$	initial (solid) velocity vector
$\bar{\mathbf{v}}$	prescribed (solid) velocity vector
$\dot{\mathbf{v}}$	(solid) acceleration vector $\mathbf{a}_\mathrm{s}$
$\mathbf{v}_h$	approximate (solid) velocity vector
$\mathbf{v}_i$	(solid) velocity vector evaluated at node i
$\mathbf{v}_p^*$	particle (solid) velocity vector interpolated from nodal velocities
w	distributed load
$\mathbf{w}$	water velocity vector
$\mathbf{w}_0$	initial water velocity vector
$\bar{\mathbf{w}}$	prescribed water velocity vector
$\dot{\mathbf{w}}$	water acceleration vector $\mathbf{a}_\mathrm{w}$
$\mathbf{w}_h$	approximate water velocity vector
$\tilde{\mathbf{w}}$	interstitial velocity, $\mathbf{w}-\mathbf{v}$
$\mathbf{x}$	vector of unknowns; current coordinate
$\Delta\mathbf{x}_e$	distance between cell centres of centre cell and east cell
$\Delta\mathbf{x}_f$	distance between cell centres of centre cell and cell that shares face f
$\mathbf{x}_i$	nodal coordinate
$\mathbf{x}_p$	current material point coordinate
$\Delta\mathbf{x}$	cell dimension
Δx_min	minimum cell dimension
y	deflection of beam
z	spatial variable in one-dimensional consolidation problem

A	surface area
A_τ	traction surface area
$\mathbf{A}$	coefficient matrix
$\mathbf{B}$	strain-displacement matrix $\nabla\mathbf{N}$
$\mathbf{B}_\mathrm{NL}$	non-linear strain-displacement matrix
$\mathbf{C}$	fluid compressibility matrix
$\mathbf{D}$	stiffness matrix
E	elastic stiffness modulus
E_oed	oedometer or constrained modulus
$\mathbf{F}$	deformation gradient
$\mathbf{F}_\mathrm{drag}$	nodal drag force vector

$\mathbf{F}_{\text{ext}}$	nodal external force vector
$\mathbf{F}_{\text{ext, w}}$	nodal external solid force vector
$\mathbf{F}_{\text{int}}$	nodal internal force vector
$\mathbf{F}_{\text{int, w}}$	nodal internal water force vector
$\sum \mathbf{F}$	sum of forces
G	shear modulus
H^1	Sobolev space of degree 1
I	second moment of area
$\mathbf{I}$	second order identity tensor
J	Jacobian, determinant of $\mathbf{F}$
K	bulk modulus
K_{dr}	drained bulk modulus
K_{s}	bulk modulus of soil grains
K_{u}	undrained bulk modulus
K_{w}	bulk modulus of water ≈ 2.2 GPa
$\mathbf{K}$	tangent stiffness matrix
$\bar{\mathbf{K}}$	tangent stiffness matrix with inertial terms
$\mathbf{K}_{\text{geo}}$	non-linear geometric stiffness matrix
$\mathbf{K}_{\text{mat}}$	material stiffness matrix
L	linear momentum
L^2	space of square integrable functions
$\mathbf{L}_p$	particle velocity gradient
M	Biot modulus
M_p	concentrated mass in material point; cross-sectional pore pressure moment
$\mathbf{M}$	global mass matrix
$\mathbf{M}_p$	pressure mass matrix
$\mathbf{N}$	shape function matrix
N_{bp}	number of boundary particles
N_{e}	number of elements in domain
N_i	shape function evaluated at node i
N_{in}	shape function at node i evaluated at cell centre of element n
N_{ip}	shorthand notation for $N_i(\mathbf{x}_p)$
N_{n}	total number of grid nodes
N_{p}	total number of subdomains Ω_p^0 in Ω^0
P_0	element with constant pressures
$\mathbf{P}^{\beta\alpha}$	momentum between phases β and α
$\mathbf{P}$	preconditioning matrix

Q_1	element with displacements in linear space
$\mathbf{Q}_1$	coupling matrix
$\mathbf{Q}_2$	coupling matrix
$\tilde{\mathbf{Q}}_2$	coupling matrix with inertial terms
R	some subregion
$\mathbb{R}$	real one-dimensional space
$\mathbb{R}^3$	real three-dimensional space
S	integral over surface of element
$\tilde{\mathbf{S}}_A$	Schur complement approximation matrix
T	dimensionless time factor $c_v t/h^2$
T_f	transmissibility coefficient of face f
T_P	transmissibility coefficient of centre cell P
$\tilde{T}_P$	adjusted transmissibility coefficient of centre cell P for boundary conditions
$\mathbf{T}$	transmissibility matrix
V	total volume
V_c	material element volume
$\dot{V}_e$	volumetric flow rate through east face of cell
$\dot{V}_{\text{total}}$	total volumetric flow rate through all cell faces
V_{e}	geometric element volume
V_{min}	minimum volume for boundary assignment
V_p	material point volume
V_{s}	volume of solid
V_{w}	volume of water
$\mathbf{X}$	initial coordinate
$\mathbf{X}_p^0$	initial centroid position vector of subdomain Ω_p^0
DTOL	user-defined displacement tolerance
ITOL	user-defined tolerance for the global convergence of the coupled MPM-FVM
RTOL	user-defined residual tolerance
$\mathscr{G}$	general constitutive model
$\mathscr{O}()$	big O notation
$\mathscr{P}$	functional space for solution of pore pressures p
$\mathscr{P}_0$	functional space for the test function φ
$\mathscr{P}_{0,h}$	functional subspace of $\mathscr{P}_0$for the approximate test function φ_h

$\mathscr{P}_h$	functional subspace of $\mathscr{P}$ for the approximate solution of pore pressures p_h
$\mathscr{T}$	loading function
$\mathscr{U}$	functional space for solution of displacements $\mathbf{u}$
$\mathscr{U}_0$	functional space for the test function $\boldsymbol{\eta}$
$\mathscr{U}_{0,h}$	functional subspace of $\mathscr{U}_0$for the approximate test function $\boldsymbol{\eta}_h$
$\mathscr{U}_h$	functional subspace of $\mathscr{U}$ for the approximate solution of displacements $\mathbf{u}_h$
$\mathscr{V}_0$	functional space for the test function $\boldsymbol{\eta}$
α	a phase; $1/K_{\mathrm{dr}}$
β	another phase; Newmark method coefficient
γ	Newmark method coefficient
δ	infinitesimal change; Dirac delta function
$\boldsymbol{\varepsilon}$	strain tensor
ε_{v}	volumetric strain
$\dot{\varepsilon}_{\mathrm{v}}$	volumetric strain strate
$\boldsymbol{\eta}$	test function
$\boldsymbol{\eta}_h$	approximate test function
$\boldsymbol{\eta}_i$	test function evaluated at node i
$\boldsymbol{\eta}_p$	test function evaluated at $\mathbf{x}_p$
θ	coefficient for time integration
κ	absolute permeability scalar
$\boldsymbol{\kappa}$	absolute permeability tensor
μ_{w}	dynamic viscosity of water
ν	Poisson ratio
ρ	total density
ρ_α	density of phase α
ρ^α	average density of phase α
ρ_p	material point density
ρ_{s}	density of solid
ρ^{s}	average density of solid phase
ρ_{w}	density of water
ρ^{w}	average density of water phase
$\boldsymbol{\sigma}$	stress tensor
$\boldsymbol{\sigma}_0$	initial stress tensor
$\boldsymbol{\sigma}'$	effective stress tensor

$\boldsymbol{\sigma}'_0$	initial effective stress tensor
$\boldsymbol{\sigma}_h$	approximate stress tensor
$\boldsymbol{\sigma}_p$	stress tensor evaluated at $\mathbf{x}_p$
$\boldsymbol{\sigma}_\mathrm{s}$	solid stress tensor
$\boldsymbol{\sigma}^s$	specific stress tensor $\boldsymbol{\sigma}/\rho$
$\boldsymbol{\sigma}^s_p$	specific stress tensor $\boldsymbol{\sigma}/\rho$ evaluated at $\mathbf{x}_p$
σ_v	volumetric stress
$\dot{\sigma}_\mathrm{v}$	volumetric stress strate
$\boldsymbol{\sigma}_\mathrm{w}$	water stress tensor
$\boldsymbol{\tau}$	boundary traction force
$\bar{\boldsymbol{\tau}}$	prescribed boundary traction force
$\hat{\boldsymbol{\tau}}_i$	integrated prescribed boundary traction force at node i
ϕ	a Eulerian function
φ	test function for pore pressures
φ_h	approximate test function for pore pressures
φ_n	shape function in element n for pore pressures
Δ_max	maximum deflection
Φ	a Lagrangian function
Ω	a body, volume or domain; deformed body, volume or domain
Ω_0	a body, volume or domain at initial time t_0
Ω^0	the original domain
Ω^0_p	the original subdomains
Ω_e	element domain
Ω_n	volume of element n
Ω_p	partice volume
$\partial\Omega$	boundary of the domain
$\partial\Omega_f$	surface of element face f
$\partial\Omega_n$	surface of element n
$\partial\Omega_p$	pore pressure boundary, subset of $\partial\Omega$
$\partial\Omega_\mathbf{q}$	water flux boundary, subset of $\partial\Omega$
$\partial\Omega_{\boldsymbol{\tau}}$	traction boundary, subset of $\partial\Omega$
$\partial\Omega_\mathbf{u}$	displacement boundary, subset of $\partial\Omega$
$\partial\Omega_\mathbf{w}$	water velocity boundary, subset of $\partial\Omega$
∂R	boundary of a subregion
∇	vector partial differential operator, $[\partial_x, \partial_y, \partial_z]$

Acronyms

ALE	arbitrary Lagrangian-Eulerian
CEL	coupled Eulerian-Lagrangian
CFL	Courant-Friedrichs-Lewy
CPDI	convected particle domain interpolation
CPT	the cone penetration test
DDMP	dual domain material point
DEM	the discrete element method
FDM	the finite difference method
FEM	the finite element method
FLIP	the fluid-implicit-particle method
FVM	the finite volume method
GIMP	generalised interpolation material point
MPM	the material point method
PDE	partial differential equations
PIC	particle-in-cell
SPH	smoothed particle hydrodynamics
TPFA	two-point flux approximation