



HAL
open science

Micro-macro modeling of the long-term behavior of well-seals under hydraulic-gas loading using the Discrete Element Method (DEM)

Robert Alexander Caulk

► **To cite this version:**

Robert Alexander Caulk. Micro-macro modeling of the long-term behavior of well-seals under hydraulic-gas loading using the Discrete Element Method (DEM). Chemical and Process Engineering. Université Grenoble Alpes [2020-..], 2022. English. NNT: 2022GRALI003 . tel-03625717

HAL Id: tel-03625717

<https://tel.archives-ouvertes.fr/tel-03625717>

Submitted on 31 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

Spécialité : 2MGE : Matériaux, Mécanique, Génie civil, Electrochimie

Arrêté ministériel : 25 mai 2016

Présentée par

Robert CAULK

Thèse dirigée par **Bruno Chareyre**, Maître de conférences,
Université Grenoble Alpes

préparée au sein du **Laboratoire Sols, Solides, Structures, et Risques**
dans l'**École Doctorale I-MEP2 - Ingénierie - Matériaux, Mécanique, Environnement, Energétique, Procédés, Production**

Modélisation micro-macro par la méthode des éléments discrets du comportement à long term des scellements de puits sous sollicitation hydraulique-gaz

Micro-macro modeling of the long-term behavior of well seals under hydraulic-gas loading using the Discrete Element Method

Thèse soutenue publiquement le **28 janvier 2022**
devant le jury composé de :

Christophe Martin

Directeur de Recherches, Centre national de la recherche scientifique, Président

Sebastia Olivella

Professeur des universités, Politècnica de Catalunya, Rapporteur

Alessandro Tarantino

Professeur des universités, University of Strathclyde Glasgow, Rapporteur

Vincent Richefeu

Maître de conférences, Université Grenoble Alpes, Examineur

George Pinder

Professeur des universités, University of Vermont, Examineur

Nadia Mokni

Ingenieur de Recherche, Institut de Radioprotection et de Sûreté Nucléaire, Invitée

Bruno Chareyre

Maître de conférences, Université Grenoble Alpes, Directeur de thèse



To each of my four grandparents, thank you for giving me a strong foundation to foster my curiosity.

Acknowledgments

If a PhD is an accumulation of academic interests and the blossoming of a young researcher's toolbox, *then* I owe my diverse academic journey to a variety of people and institutions. I would be nothing without the faculty at the University of Vermont, where I was introduced to scientific honesty, curiosity, and critical thinking. But my academic journey, and my strong beliefs that I've adopted along the way, wouldn't be complete without my introduction to the open source software community. I must thank the entire Yade community for showing me the rewards associated with freely sharing ideas, code, and assistance in the realm of open-science. I should make a special thanks to the "Yade guru", Bruno Chareyre, for spending countless hours impressing upon me his unique and effective approach to problem solving. His ability to solve complex problems with simple solutions is reminiscent of a quote by Sir Isaac Newton: *"truth is ever to be found in the simplicity, and not in the multiplicity and confusion of things."* Of course, there are countless other academics I should thank here, but as I've learned best, I will coarsely discretize them by the institutions that they comprise: the Grenoblois from Laboratoire 3SR who make an incredibly fertile and supportive research environment, the patient people at the Institut de Radioprotection et de Sûreté Nucléaire who understand the unique challenges of modeling clay, and the compassionate people from the Jacob's School of Engineering who helped me discover France. These institutions, and the people within, took a chance on my work and helped me along the way. They therefore deserve credit for my development as a scientist. Thank you.

But, a PhD is not simply an accumulation of academic interests. The holistic requirements of such an endeavor include a broad spectrum of skills beyond math and science; it would be futile to try and list them. Likewise, it would be futile to list the people responsible for helping me build these skills - but there is no doubt that all of my friends, old and new, and families, big and small, each played key roles in supporting me and my journey. If I can coarsely mention these groups: I would like to point out that my Grenoblois community was integral to my painstaking adaptation to French culture, and especially supportive during the global pandemic. To my Vermont friends, who somehow remain supportive to me, even as I wander across time-zones far from the safety of the Burlington. All of my childhood friends in Colorado who make time for me when I visit - you remind me of what is truly important. I shouldn't forget my long-list of squash partners from around the world - what a privilege it has been, thank you for helping me stay sane through out my academic journey. To my New England family, who will drop anything to help me in my times of need. And to my Colorado family, your support may very well be a key-stone in my career. Many of these people, whom I am lucky enough to call true friends, hold impressively high moral and intellectual standards. There is no doubt that half of what I considered "impossible" has been deemed possible by these incredible people, and that alone paved my intellectual journey. I hope I can return the favor some day.

Contents

Contents	v
Abstract	xv
Résumé	xvi
INTRODUCTION	1
1 Introduction	3
1.1 Overview	3
1.2 Broad motivation	3
1.3 Objective	5
1.4 Outline	6
1.5 Record of support	7
BACKGROUND	9
2 Background	11
2.1 Origins of flow in porous media	11
2.2 Swelling clay, the fundamentals	11
2.2.1 Hydro-mechanical modeling	11
2.2.2 Role of suction	13
2.2.3 Role of heterogeneity	13
2.2.4 History of permeability evolution in engineered clay barriers	14
2.3 Discrete Element Method	14
2.4 Numerical methods for poromechanics in granular media	15
2.5 Computational limitations in poromechanical modeling	16
MODEL DEVELOPMENTS	17
3 A conceptual model for micro-scale crack developments in partially saturated compacted clay pellets	19
3.1 Summary	19
3.2 Introduction	19
3.3 Methods	20
3.3.1 Mechanical Model	20
3.3.2 Partially saturated flow model	21
3.3.3 Volumetric Swelling Model	23
3.3.4 Assignment and evolution of porosity	24
3.3.5 Hydro-mechanical model	25
3.3.6 Crack model	26
3.4 Alpha shape boundary conditions	28
3.5 Compacted clay hydration example	28
3.5.1 Model configuration	29
3.5.2 Model calibration	29

3.5.3	Measuring permeability numerically	30
3.5.4	Effect of confining pressure on permeability evolution	31
3.5.5	Effect of heterogeneity on crack developments	32
3.5.6	Effect of crack model on permeability evolution	32
3.5.7	Evolution of porosity distribution	34
3.5.8	Convergence study	37
3.5.9	Spatial scale effects	37
3.5.10	Practical reproduction of results	38
3.6	Conclusion	40
4	A framework for modeling hydro-mechanical processes in clay pellet-powder mixtures	43
4.1	Introduction	43
4.2	Methods	43
4.2.1	Mechanical Model	43
4.2.2	Mass Transport Model	44
4.2.3	Volumetric Swelling Model	47
4.2.4	Hydro-mechanical Model	49
4.3	Solving the non-linear problem	49
4.4	Numerical stability	51
4.5	Defining structural heterogeneity	52
4.6	Up-scaling permeability data from micro-scale models	53
4.7	Practical implementation	53
4.8	Experimental validation and results	54
4.9	Conclusions	57
4.10	Future advancements	57
5	Computational acceleration of the hydro-mechanical coupling	59
5.1	Summary	59
5.2	Methods	59
5.2.1	Discrete Element Method contact model	59
5.2.2	Pore Finite Volume (PFV) Scheme	60
5.3	Acceleration techniques	61
5.3.1	Matrix Factor Reuse	61
5.3.2	Multithreaded factorization	62
5.3.3	GPU Accelerated Factorization	63
5.3.4	Parallel Task Management	63
5.4	Test setup	64
5.4.1	Computer Details	64
5.4.2	Model details	64
5.4.3	Data description	64
5.5	Results and Discussion	65
5.5.1	Increasing mesh resolution for partially saturated problems	67
5.6	Conclusions	70
5.7	Stability of the coupled algorithm	70
5.8	Practical reproduction of results	72
	CONCLUSION	73
6	Conclusions and Perspectives	75
6.1	Hypothesis	75
6.2	Major conclusions	76

6.3	Perspectives	77
6.4	Acknowledgments	78
RÉSUMÉ EN FRANÇAIS		79
7	Résumé en français	81
7.1	Introduction Générale	81
7.2	Objectif de la thèse	82
7.3	Modèles conceptuels	82
7.3.1	Modèle hydromécanique à l'échelle macro pour les mélanges granulés-poudre	82
7.3.2	Modèle hydromécanique à l'échelle microscopique pour argile compactée hétérogène	84
7.3.3	Accélérer la solution du couplage hydro-mécanique	86
7.4	Conclusions	88
APPENDIX A: SUPPORTING MATERIALS		91
.1	Macro-scale pellet-powder mixture python script	93
.2	Micro-scale compacted clay script	101
Bibliography		119

List of Figures

1.1	Placement of a clay pellet-powder mixture engineered barrier in a future underground disposal facility, Cigéo.	3
1.2	MX80 bentonite clay pellets at various stages of hydration, with crack network [14]	4
1.3	Lab-scale (1/10th) engineered barrier hydration experiment (MX80 bentonite pellet-powder mixture) 20/80 in dry mass [14]	4
1.4	Overview of scales of interest, micro and macro scales modeled in Chapter 3 and 4	6
2.1	Layers of montmorillonite with interlayer cations and water molecules.	12
2.2	Diagram of clay structure and porosity scales [7]	12
2.3	A simplistic representation of normal, k_n , and shear, k_s , stiffnesses in Hooke's law used to estimate forces between two particles, a and b	15
2.4	Yade open source DEM software	15
3.1	Visualization of the flow model and DEM geometries. Points spaced out to aid with labelling. Contact forces only occur for interacting radii.	21
3.2	Pressure saturation curve for MX80 clay pellet [14] and the effect of changing porosity [32].	24
3.3	Initial porosity distribution comparison top) Xray CT image gray values, 7 mm diameter and pixel size $4.4\mu m$ bottom) model porosity values with cracked cells in green.	25
3.4	Pellet scanning in the tomograph of Laboratoire 3SR.	25
3.5	Geometrical areas used for pressure and viscous force integration.	26
3.6	Geometrical quantities associated with crack network model.	28
3.7	top) Non-alpha triangulation with flat tetrahedra on boundary bottom) alpha trimmed triangulation	28
3.8	Model boundary conditions, initial conditions, and dimensions.	29
3.9	Left) Volumetric strain calibration. Black "x" showing data from [14] Right) Strength and stiffness calibration. Red "x" showing data from [17]	31
3.10	Evolution of gas and water conductivities with respect to volumetric saturation	31
3.11	Effect of confining pressure on water and gas permeability evolution. Point labels indicate water saturation.	32
3.12	Comparison of the effect of homogeneous and heterogeneous initial porosity fields on the development of cracks during volumetric swelling.	33
3.13	Effect of crack model on macroscopic permeability evolution during hydration.	34
3.14	Effect of drained and imbibed cracks on water paths and velocities at varying levels of average water saturation.	35
3.15	Evolution of crack and permeability fabric during hydration. Colored tiles represent the components of the crack fabric tensor.	35
3.16	Evolution of porosity distribution in pellet.	36
3.17	Convergence of characteristic output quantities for varying Discrete Element point cloud densities (N=number of DEM points) and associated finite element triangulation resolutions.	37
3.18	Statistical investigation of characteristic model outputs for various domain scales. μ and σ indicate the statistical mean and standard deviation, respectively.	39
4.1	CT scan imagery of bentonite clay pellet-powder mixture specimen used for hydration tests [15].	43
4.2	Domain discretization and geometric quantities of the mass transport equations for a clay pellet powder mix.	44

4.3	Visualization of key quantities for the coupled DEM mass transport model.	45
4.4	Loading path for partially saturated clay transitioning from unconfined to confined conditions (left) and the corresponding effective pressure (right).	48
4.5	Geometrical areas used for pressure and viscous force integration.	49
4.6	Demonstration of initial structural heterogeneity assignment. Weibull distributed initial powder volume fractions (triangulation). Random packing for monodisperse pellet sizes (sphere locations).	52
4.7	Permeability curve, Chap. 3, at the micro-scale for unconfined conditions. This permeability is used to inform the pellet permeability, k , in the present model.	53
4.8	Overview of algorithmic implementation.	54
4.9	Mock-up scale model geometry and boundary conditions	55
4.10	Powder and pellet swelling distribution, experimental observation vs numerical model $t = 0\%$ (left) and $t = 20\%$ (right).	56
4.11	Swelling pressure data comparison for numerical and experimental mock-up tests hydration tests.	57
5.1	Example of a 430 DOFs positive definite, symmetric, banded, sparse conductivity matrix (\mathbf{G})	61
5.2	a) Pressure difference for various remesh intervals after 1200 iterations and 10 parallel cores for DEM force summations b) Replicate tests using Remesh interval = 20 iterations and 10 parallel cores for DEM force summations	62
5.3	Yade DEM+PFV accelerated algorithm overview.	62
5.4	Example of one of the cubical DEM+PFV 1-D consolidation models used to test performance of GPU accelerated factorization.	65
5.5	Performance comparison for non-accelerated and fully accelerated algorithms.	66
5.6	a) Time required to factorize and analyze the conductivity matrix (Eq. 5.9). $t_{bg} = t_{factor} + t_{analyze}$ b) Zoomed in to show device timings for small packings (bottom)	67
5.7	Time required to allocate conductivity matrix (Eq. 5.9) to memory (left) and build the system of equations (Eq. 5.9) (right)	68
5.8	Time per iteration, optimal remesh interval (λ_{rm}) associated with v_{iter} and t_{bg} , and Cundall number for various conductivity matrix (Eq. 5.9) sizes	68
5.9	Performance gain from implementing OpenMP parallel task management (Fig. 5.3).	69
5.10	Wall time spent solving for pressures (Eq. 5.11), computing pore volumes, and computing forces (Eq. 5.12) (parallel task management in Fig. 5.3).	69
5.11	Mesh resolution limit before, $7e4$ degrees of freedom (top) and after acceleration techniques, $7e5$ degrees of freedom (bottom).	70
7.1	Placement of a clay pellet-powder mixture engineered barrier in an underground disposal facility, Cigéo.	81
7.3	Lab-scale engineered barrier (MX80 bentonite pellet-powder mixture) [14]	81
7.2	Hydration of an MX80 bentonite clay pellet, with crack network [14]	82
7.4	Imagerie tomographique d'un échantillon de mélange de pastilles et de poudre d'argile utilisé pour les tests d'hydratation [15].	82
7.5	Une représentation simplifiée des rigidités normales, k_n , et de cisaillement, k_s , dans la loi de Hooke utilisée pour estimer les forces entre deux particules, a et b	83
7.6	Domain discretization and geometric quantities of the mass transport equations for a clay pellet powder mix.	83
7.8	Permeability curve, Chap. 3, at the millimeter scale. This permeability is used to inform the pellet permeability, k , in the present model.	84
7.7	Visualisation des grandeurs clés pour le modèle de transport de masse DEM couplé.	84
7.9	Maquette de la géométrie du modèle à l'échelle et des conditions aux limites.	85

7.10	Comparaison des données de pression de gonflement pour les essais sur maquette numérique et expérimentale des tests d'hydratation.	86
7.11	Grandeurs géométriques associées au modèle de réseau de fissures.	86
7.12	Comparaison de la distribution de la porosité initiale en haut) Valeurs de gris de l'image Xray CT, 7 mm de diamètre et taille de pixel $4,4\mu m$ en bas) valeurs de porosité du modèle avec des cellules fissurées en vert.	86
7.13	Effet du modèle de fissure sur l'évolution de la perméabilité macroscopique au cours de l'hydratation.	87
7.14	Exemple d'une matrice de conductivité définie positive, symétrique, à bandes et clairsemée à 430 degrés de liberté (G)	87
7.15	Présentation de l'algorithme accéléré Yade DEM+PFV.	88

List of Tables

3.1	Partially saturated flow model parameters	30
3.2	Comparison of porosity distribution evolution for experimental data and present model.	34
4.1	Discrete Element Method model parameters for modified Hertz contact.	54
4.2	Mass transport model parameters.	55
5.1	Numerical specimen DEM microproperties	65

List of symbols

A_{ij}	Area of the incident facet between tetrahedrons i and j [m ²]
A_{ij}^a	Intersection area of DEM particle a with tetrahedra i and j [m ²]
$a_{c,ab,ij}$	Crack area between DEM points a and b and tetrahedra i and j
a	Coefficient for porosity dependent Van Genuchten curve parameter [-]
b	Coefficient for porosity dependent Van Genuchten curve parameter [-]
b_k	Intrinsic permeability coefficient [-]
C_{ij}	Discretized contour between systems i and j [m ²]
c_i	Constraint imposed on Newton Raphson minimization procedure [-]
χ	Time step stability factor [-]
d_n	Normal displacement at particle contact [m]
Δd_s	Tangential increment of displacement at particle contact [m]
ΔD_{ij}	Displacement between particles ij [m]
Δt	time step [s]
$\delta \mathbf{x}$	Vector of updates for minimization in Newton Raphson
E	Elastic modulus of the solid [Pa]
err	Root mean square error [-]
\mathbf{f}	Body force [N]
$\mathbf{f}_{ij}^{p,k}$	Fluid pressure force acting on particle k in pore throat ij [N]
\mathbf{f}_{ij}^V	Total viscous force acting on pore throat ij [N]
$\mathbf{f}_{ij}^{v,k}$	Viscous force acting on particle k by pore throat ij [N]
f_j	Function j derived from Van Genuchten for Newton Raphson optimization
ε	Volumetric deformation [m ³ /m ³]
\mathbf{Ex}	Vector of rate of volume change
\mathbf{G}	Symmetric conductivity matrix for flow solution
G	Shear modulus of the solid [Pa]
g_{ij}	Conductivity between tetrahedrons i and j [m ³ Pa ⁻¹ s ⁻¹]
γ	Gas-water surface tension [N/m]
γ_{int}	Interaction detection factor [-]
\mathbf{J}	Jacobian matrix
$k_{M,ij}$	Intrinsic permeability between tetrahedrons i and j [m ²]
$k_{rM,ij}$	Residual permeability factor between tetrahedrons i and j [-]
k_o	Intrinsic permeability [m ²]
$k_{n,ij}$	Normal spring stiffness between particles ij [N/m]
$k_{s,ij}$	Shear spring stiffness between particles ij [N/m]
\mathbf{K}	Global stiffness matrix
L	Distance between pressure measurements [m]
λ	Curve parameter function for Van Genuchten [-]
λ_o	Reference λ [-]
λ_{rm}	Remesh interval [iterations]
Λ	2nd order crack fabric tensor
Λ	Diagonal matrix of Eigenvalues
\mathbf{M}	Global mass matrix
\mathbf{m}	Particle mass for integration of Newton's 2nd law [kg]
m_s	Mass of solids for ρ_d estimate
n	Permeability coefficient parameter
μ	Water/gas dynamic viscosity [Pa·s]
ν	Poisson ratio of the solid [-]
\mathbf{n}_n	Unit vector normal to particle contact [-]
\mathbf{n}_s	Unit vector orthogonal to particle contact [-]

\mathbf{n}_{ij}	Unit vector normal to systems i and j [-]
η	Crack roughness factor [-]
Ω_i	Contour for tetrahedron i
ϕ	Porosity [-]
ϕ_{ij}	Average porosity of incident tetrahedrons i and j [-]
Φ	Mass-flux along domain contour [kg/s]
\mathbf{p}	Vector containing the pressure within each tetrahedron
p_i	Water/gas pressure in tetrahedron i [Pa]
p'^t	Effective pressure acting on pellets by clay powder during swelling, for a given time step t [Pa]
p_c	Effective pressure at confined conditions [Pa]
$p_{v,c}$	Maximum swelling pressure as a function of dry density at confined volume [Pa]
q_{ij}	Water/gas flux between tetrahedrons i and j [m ³ /s]
q	Source or sink of mass flux [kg/s]
\mathbf{Q}	Volumetric flux [m ³ /s]
\mathbf{r}	Vector of residuals
r	Sphere radius [m]
R	equivalent interaction radius [m]
ρ	bulk mass density [kg/m ³]
ρ_d	bulk mass dry density [kg/m ³]
$s_{p,i}$	Phase saturation (water or gas) of tetrahedron i [-]
$s_{r,p}$	Residual phase saturation [-]
$s_{s,p}$	Saturated phase saturation [-]
s_c	Saturation at confined conditions [-]
s_i	Suction of tetrahedron i [Pa]
S_{ij}^f	Fluid contour between pores ij [m ²]
t_{bg}	Time spent factorizing \mathbf{G} in background [s]
Θ	Full domain of two-phase flow problem [m ³]
$\partial\Theta$	Contour of full domain of two-phase flow problem [m ²]
u_{ab}	distance between particles ab [m]
\mathbf{u}	Water/gas fluid velocity [m/s]
\mathbf{v}	Contour velocity [m/s]
v_{iter}	Velocity of simulation [iters/s]
V_i	Total material volume [m ³]
$V_{t,i}$	Total volume of tetrahedron i [m ³]
\dot{V}	Volume change of tetrahedron i [m ³]
\mathbf{V}	Vector of rate of volume changes
$V_{s,i}$	Volume of solids in tetrahedron i [m ³]
ω	angular velocity [s ⁻¹]
\mathbf{x}_{ab}	Vector pointing from DEM point a to DEM point b
Ξ	Numerical specimen domain
\mathbf{x}	Body position [m]
ζ	Curve parameter for Van Genuchten [Pa]
ζ_o	Reference ζ [Pa]

Abstract

The present research stems from a pressing need to safely contain the radioactive waste by-products originating from popular nuclear power generation methods. These radioactive nuclear waste byproducts can remain dangerous for as little as 30 years or as long as 2 million years. Such geological time-scales warrant a geological disposal solution. Enter deep geological nuclear waste disposal facilities, such as the envisioned Cigéo facility in France. The research presented here is geared towards investigating clay-based seals for these disposal facilities. One of the candidate seals is comprised of a mixture of compacted clay spheres and clay powder. Although this mixture shows great long-term containment promise due to its swelling potential, low-permeability, and ease of installation, there remain a variety of questions surrounding micro- and macro-scale hydro-mechanical processes associated with the presence of material and structural heterogeneities. These questions are being actively investigated in an experimental setting, but the academic research focused on numerically modeling these processes remains under-developed due to the complexity of hydro-mechanical behavior in swelling clay materials. Despite this deficiency in the literature, numerical modeling still promises a wide range of analysis tools that experimentalists do not have access to, such as the ability to statistically analyze macro-scale structural heterogeneities or measure directional permeability at the micro-scale due to the development of cracks. Thus, with funding from the Institut de Radioprotection et de Sûreté Nucléaire, the present research presents a collection of novel conceptual models designed to account for the development of hydro-mechanical processes under anisotropic loading conditions in swelling clay-based nuclear waste seals. These conceptual models are implemented into flexible, open-sourced, scientific tools which are accelerated using modern tools to enable in-depth multi-scale numerical analyses. The work draws important conclusions about the evolution and manifestation of the micro-scale heterogeneities in compacted clay, as well as the macro-scale hydro-mechanical behaviors during anisotropic loading conditions in compacted clay-powder mixtures.

Résumé

La présente recherche découle d'un besoin pressant de confiner en toute sécurité les sous-produits des déchets radioactifs provenant des méthodes populaires de production d'énergie nucléaire. La radioactivité de divers sous-produits des déchets nucléaires peut rester dangereuse pendant aussi peu que 30 ans ou jusqu'à 2 millions d'années. De telles échelles de temps géologiques justifient une solution de stockage géologique comme les installations de stockage géologique en profondeur. La recherche qui est présentée ici est orientée vers l'étude des configurations de scellement des installations à base d'argile pour Cigéo (une installation prévue en France). L'une des techniques candidats est composé d'un mélange de sphères d'argile compactées et de poudre d'argile. Bien que le sceau à base d'argile montre une grande promesse de confinement à long terme en raison de son potentiel de gonflement et de sa faible perméabilité, il reste une variété de questions concernant les processus à micro et macro-échelle se manifestant et interagissant avec la présence d'hétérogénéités matérielles et structurelles. Ces questions sont activement étudiées dans un cadre expérimental, mais la recherche académique axée sur la modélisation numérique de ces processus reste sous-développée en raison de la complexité du comportement hydromécanique des matériaux argileux gonflants. Malgré cette carence de la littérature, la modélisation numérique promet encore un grand gamme d'outils d'analyse auxquels les expérimentateurs n'ont pas accès, comme la capacité d'analyser statistiquement des hétérogénéités structurelles à macro-échelle ou de mesurer la perméabilité directionnelle à la micro-échelle en raison du développement de fissures. Ainsi, grâce au financement de la Commission Canadienne de Sécurité Nucléaire et de l'Institut de Radioprotection et de Sécurité Nucléaire, la présente recherche présente une collection de nouveaux modèles conceptuels conçus pour rendre compte du développement de processus hydromécaniques dans des conditions de chargement anisotropes dans l'argile gonflante avec l'application aux scellements de déchets nucléaires. Ces modèles conceptuels sont mis en œuvre dans des outils scientifiques flexibles et open-source qui sont accélérés à l'aide d'outils modernes pour permettre des analyses numériques multi-échelles approfondies. Le travail tire des conclusions importantes sur l'évolution et la manifestation des hétérogénéités à micro-échelle dans l'argile compactée, ainsi que sur les comportements hydromécaniques à macro-échelle lors de conditions de chargement anisotropes dans les mélanges argile-poudre compactés.

INTRODUCTION

1.1 Overview

Secure, sustainable, and robust nuclear waste disposal systems are paramount to expanding and maintaining the existing nuclear based electricity grid. Despite a strong collaborative international experimental and numerical effort to develop these disposal technologies, there remain a host of unknown variables associated with the long-term containment efficacy of clay-based vertical seals. In the context of the present thesis, two novel conceptual models are developed and discretized numerically to elucidate the microscopic and macroscopic hydraulic processes in nuclear disposal facility clay seals.

1.2 Broad motivation

As of 2021, nuclear energy provided 10% of the world's electricity. The growing demand for clean nuclear based electricity will follow the growing world population and the pressing need to limit carbon emissions. Although nuclear power is rarely grouped with renewable energy sources, it still has the ability to reduce carbon emissions more than any other energy source on the planet [1]. However, nuclear energy does produce a toxic by-product and as nuclear energy continues to grow in France and the rest of the world, so do the volumes of nuclear waste. According to [2], 90% of the nuclear waste is stored safely in near surface repositories because it is classified as low level waste. Meanwhile, the remaining 10% is considered "intermediate level waste" and "high level waste" (HLW), which both require special disposal considerations due to the possible negative humanitarian impacts associated with contact to high levels of radioactivity. Currently, scientific consensus points to deep geological disposal as the safest configuration for long term disposal of HLW [3]. In fact, many countries, including France, Sweden, Finland, and the USA, have active deep geological disposal projects. One project, called "SEALEX" (SEALing performance EXperiments), was launched by the Institut de Radioprotection et de Sûreté Nucléaire (IRSN) at the Underground Research Laboratory (URL) in Tournemire, France, and is focused on understanding and improving the hydraulic performance of various sealing systems [4]. One particular seal of interest, shown in Fig. 1.1, is called the vertical shaft seal (VSS). These seals are undoubtedly one of the key elements for safe long-term containment since they constitute the main potential pathway between nuclear wastes and the biosphere. SEALEX is already generating important research associated with the long-term behavior of these VSS [4, 5]. In fact, the principal goals of SEALEX include:

- ▶ comparing the hydraulic performance of various seal configurations
- ▶ evaluating the effect of swelling on construction related voids
- ▶ investigating the evolution of structural heterogeneities

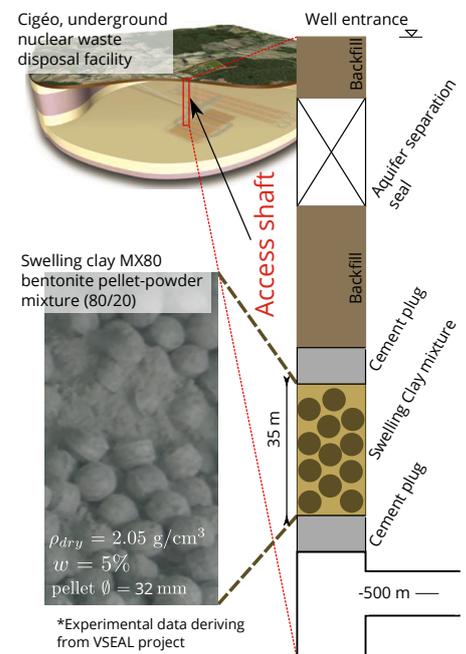


Figure 1.1: Placement of a clay pellet-powder mixture engineered barrier in a future underground disposal facility, Cigéo.

[1]: IEA (2021), *World Energy Outlook 2021*

[2]: ANDRA (2020), *National Inventory of Radioactive Materials and Waste 2020 - The Essentials*

[4]: Mokni et al. (2016), 'Hydro-mechanical analysis of SEALEX in-situ tests - Impact of technological gaps on long term performance of repository seals'

[6]: Pusch (1979), 'Highly Compacted Sodium Bentonite for Isolating Rock-Deposited Radioactive Waste Products.'

[13]: Molinero-Guerra et al. (2017), 'In-depth characterisation of a mixture composed of powder / pellets MX80 bentonite'

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[15]: Mokni et al. (2020), 'Modelling the long-term hydro-mechanical behaviour of a bentonite pellet/powder mixture with consideration of initial structural heterogeneities'

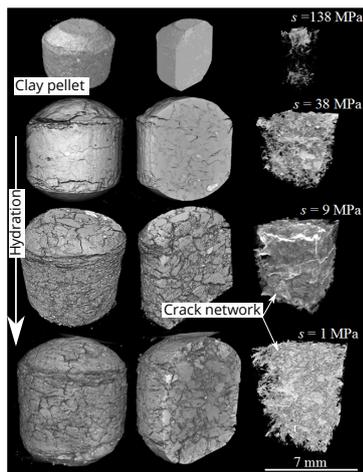
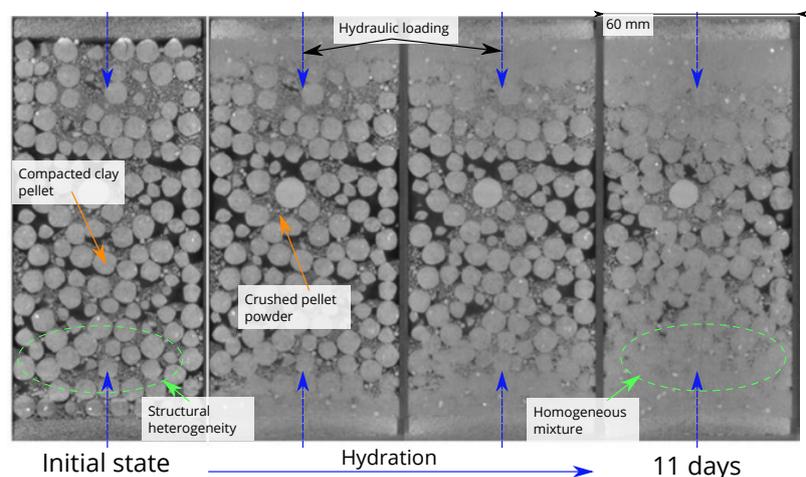


Figure 1.2: MX80 bentonite clay pellets at various stages of hydration, with crack network [14]

Many projects, including SEALEX, are focused on clay based VSS because some clays have high swelling potential and low permeability [6]. The literature boasts a strong experimental and theoretical record focused solely on the swelling properties of compacted clay [7–12]. However, there remains uncertainty associated with the practical hydraulic performance of various clay mixtures, especially with respect to various installation configurations.

One of the candidate configurations for these vertical seals consists of a mixture of swelling clays (Na-montmorillonite) in the form of a poly-disperse assembly of highly compacted pellets and crushed pellets in a strongly desaturated initial state (greater than 100MPa suction). Since the hydro-mechanical behavior of these pellet-powder mixtures plays an important role in limiting water-gas migration processes, the IRSN initiated a set of in-situ and laboratory experiments within the VSEAL (Vertical SEALing) project. These multi-scale experiments are aimed at understanding structural evolutions during hydration [13] and the hydro-mechanical effects of anisotropic imbibition under simultaneous hydraulic-gas loadings. At the pellet scale level, X-ray microtomography and CT observations revealed the development of intricate crack patterns due to vapor transport in free swelling (Fig. 1.2 [14]). Meanwhile, at the pellet-powder mixture scale, other laboratory experiments demonstrated the effect of initial structural heterogeneities on swelling pressures Fig. 1.3 [15]. Results from these studies demonstrated the importance of considering the evolution of initial heterogeneity when making assessments on the long-term sealing efficacy of pellet-powder mixtures. However, experiments are limited by boundary conditions, non-geological time-spans, and an inability to perform statistical parametric analyses across spatial scales. Thus, the goal of the present thesis is to provide a set of numerical models that can alleviate some of these limitations.

Figure 1.3: Lab-scale (1/10th) engineered barrier hydration experiment (MX80 bentonite pellet-powder mixture) 20/80 in dry mass [14]



1.3 Objective

In the context of this thesis, the IRSN is focused on improving the understanding of long-term hydraulic performance of these clay pellet-powder mixture vertical shaft seals. Considering the complexities of modeling hydro-mechanical behaviors of swelling clays for engineered barriers (Chap. 2), this thesis aims to provide a deeper conceptual understanding of the micro- and macro-scopic processes occurring in clay pellet-powder mixtures. These scales are shown in Fig. 1.4 In particular, the major goals of the thesis include the:

- ▶ construction of a conceptual model of hydro-mechanical processes at the *macro-scale* for partially saturated pellet-powder mixtures which accounts for the evolution of initial *structural* heterogeneities (Chap. 4).
- ▶ construction of a conceptual model of hydro-mechanical processes at the *micro-scale* for the evolution of *material* heterogeneities in compacted swelling clay and quantify the effect of micro-scopic crack networks on water and gas permeability in partially saturated conditions (Chap. 3).
- ▶ discretization and combination of the macro- and micro-scale conceptual models geometrically and temporally (Chap. 4).
- ▶ formulation and validation of the practical implementation of the conceptual models such that other researchers can use, modify, and reproduce results (Chapters 4 and 3).
- ▶ application of the practical implementations to relevant initial/boundary conditions to gain insights into long-term permeability evolutions in engineered barriers(Chapters 4 and 3).
- ▶ analysis and acceleration of the underlying algorithm and numerical solution methodology (Chap. 5).

Thesis Objective

Model the fundamental processes responsible for the complex behavior of heterogeneous partially-saturated swelling-clay material subjected to anisotropic hydraulic and gas loadings, with particular attention paid to understanding the persistence of initial heterogeneities on swelling pressures and gas migration.

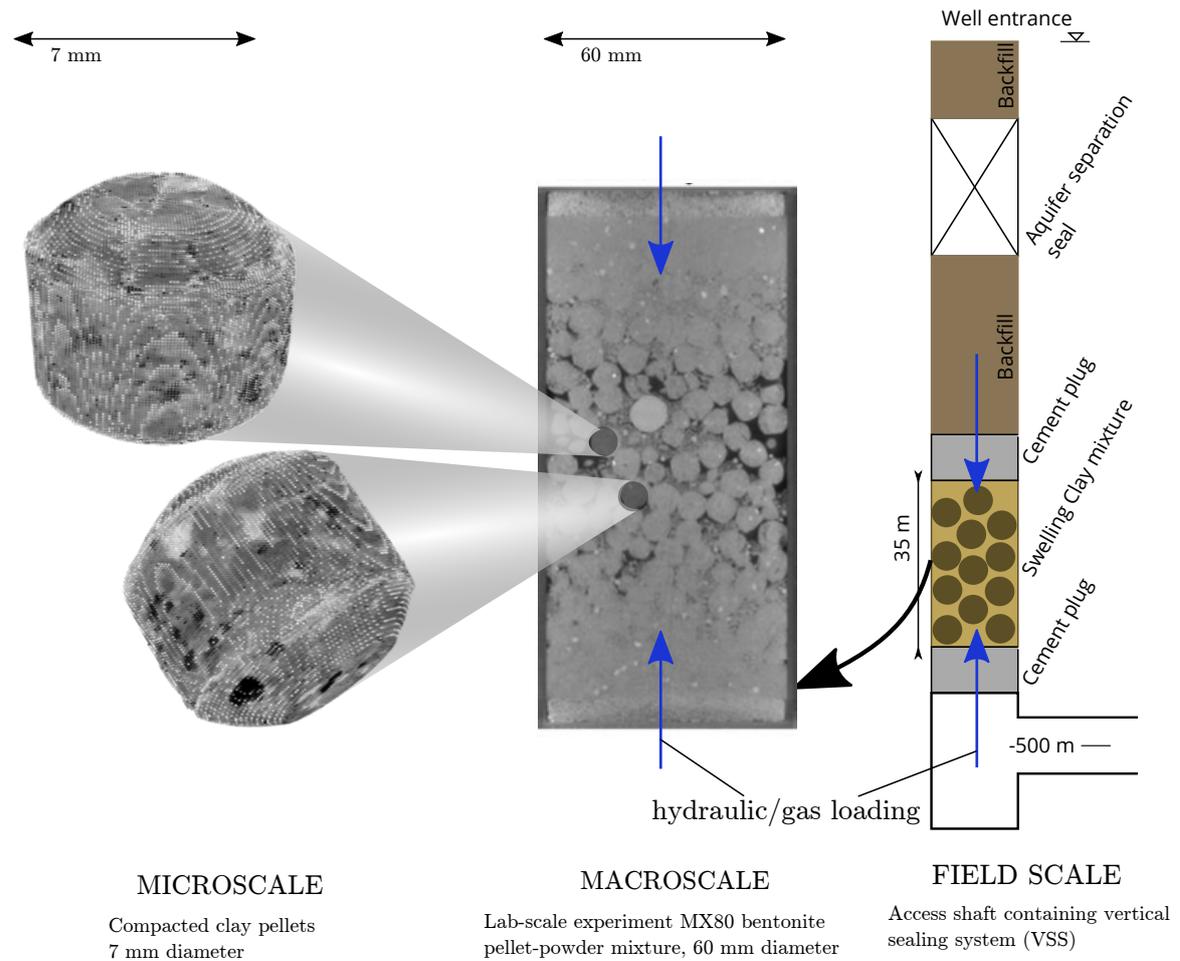


Figure 1.4: Overview of scales of interest, micro and macro scales modeled in Chapter 3 and 4

1.4 Outline

1: The full citation list is not contained in the margins, instead the reader can find the full list of references at the end of the thesis.

The thesis is written using a double column format where one column contains the text body and the other column contains margin information. The margin aids the text by containing *important* citation expansions¹, as well as notes and helper figures.

The thesis content is organized into seven chapters. Chapters 1 and Chapters 2 both introduce the reader to the thesis and provide necessary background to inform the reader of the underlying motivations as well as the historical work performed on the present topic.

Chapter 3 introduces another conceptual model of the micro-scale heterogeneities in compacted clay pellets. The fundamental model is again governed by a conservation of mass and local pressure gradients. However, the local permeability field is hypothesized to depend strongly on the development of cracks and the evolution of local porosities. The novel, physically based, crack model considers micro-scale cracks developing in compacted clay material due to material deformations and the associated required water entry pressures governed by surface tension of water. Experimental imagery is used to inform the initial heterogeneity field, and the domain is discretized using the Discrete Element Method. An

in-depth hydration analysis is performed on a numerical clay specimen which has initial conditions based on experimental imagery. The evolution of porosity is evaluated in addition to scale effects, mesh resolution requirements, crack model efficacy, and the evolution of crack fabric.

Chapter 4 introduces a conceptual model of the macro-scale geometry for a pellet powder mixture. The fundamental model is governed by a conservation of mass exchange between the compacted clay pellet domain and the crushed powder domain. These mass fluxes are modeled using a partially saturated flow model which depends on experimentally collected water retention curves from [14]. Meanwhile, the volumetric swelling is coupled to the partially saturated flow model using additional experimental data from [14, 16]. The mechanical coupling is based on a Hertz contact law, but with a saturation dependent stiffness parameter derived from laboratory experiments performed by [17]. This conceptual model is discretized using a Discrete Element formulation, which is implemented into Yade open-source software. A validation is performed on the discretized implementation and to gain insights into how structural heterogeneities impact swelling pressures in engineered clay barriers comprised of pellet-powder mixtures.

Chapter 5 discusses and implements a variety of acceleration techniques for improving the time-dependent implicit solution of fluid pressures in the poromechanical model introduced in Chapter 3. In summary, the conductivity matrix is factorized using Graphical Processing Units, it is refactored in parallel to the active simulation, it is reused for multiple time-steps, and the entire flow scheme is solved parallel to the mechanical interaction detection in DEM.

Chapter 6 reviews the objectives posed at the start of the thesis, draws general conclusions in support of these objectives, and shares perspectives for future work. Meanwhile, Chapter 7 summarizes the entire thesis in French, but is in itself a unique document which pulls together all the presented methods into one self-contained story.

1.5 Record of support

The research presented in this thesis depends strongly on a wide variety of other academics in the open-source communities. The developers of Yade created and maintained an incredible tool which counts over 150k lines of code. This robust code base enabled the present author to contribute over 6800 lines of code to the Yade source code during the course of this thesis (Yade dev team). In addition to technical assistance, the Yade community at Launchpad also played an important role in this thesis. The author received support from other academics, and reciprocated the support to other users throughout the duration of the thesis. In total, the present author answered over 500 questions in support of other users (rcaulk on launchpad). With respect to the GRICAD cluster usage in Chap. 3, the author also gained support from other members of the Yade MPI hackathon team. This special team, comprised of the present

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[16]: Darde et al. (2020), 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'

[17]: Darde et al. (2018), 'Comportment hydromecanique de pellets de bentonite: caracterisation au laboratoire et simulations DEM swelling'

author and four other members, earned 1st place for code optimization ([GENCI competition results](#)). Additional knowledge was gained and employed in Chap. 4 and 3 when the author assisted the SPAM team in the implementation of their triangulation interface [96].

[96]: Stamati et al. (2020), ‘Spam: software for practical analysis of materials’

BACKGROUND

Within this chapter, the reader is presented with important background information that supports the motivation of the work, the development of the conceptual models, and the numerical implementations in the body of the thesis (Chap. 4, Chap. 3, Chap. 5).

2.1 Origins of flow in porous media

The basis for modeling flow through porous media stems back to Henry Darcy's work as director of public works in Dijon, France, during 1856 [18]. He studied the flow of water through saturated vertical sand filters, which preceded the formal Darcy's law [19]:

$$q = -\frac{k}{\mu}(\nabla p + \rho g \nabla z) \quad (2.1)$$

where he shows that the volumetric flux, \mathbf{q} , is proportional to the fluid pressure gradient, ∇p , via a material property called the intrinsic permeability, k , and the fluid viscosity, μ . For low Reynolds numbers, the gravitational term is dropped due to negligible inertial effects. In the context of the present thesis, all fluxes are derived from this fundamental relationship. However, in the partially saturated cases under investigation, the flux problem is more complicated. The permeability is saturation dependent and the flux depends on local water-retention curves, which are two-way coupled with local characteristics such as volume changes due to swelling. Details of these variations are elaborated in Chapters 4, 3, and 5.

[18]: Darcy (1856), 'Les fontaines Publiques de la ville de Dijon, Paris: Dalmont, 1856'

[19]: Muskat (1938), 'The flow of homogeneous fluids through porous media'

2.2 Swelling clay, the fundamentals

The hydro-mechanical behavior of partially saturated clay is highly scale dependent, both temporally and spatially [20]. The reason for this behavioral discrepancy can be attributed to the fundamental processes which control the general hydro-mechanical behavior of clay at the particulate level (nano-scale). Since the present thesis contains two conceptual models of clay at two different spatial scales, the current section is presented in an effort to provide the reader with a basic understanding of fundamental processes, as well as the common methods employed by researchers to model behaviors at various scales.

[20]: Alonso et al. (1999), 'Modelling the mechanical behaviour of expansive clays'

2.2.1 Hydro-mechanical modeling

The hydro-mechanical behavior of clay is complex due to a variety of factors including load history, particle alignment, particle charge, clay type, saturation level, chemical interactions, particle size distribution, pore size distribution, etc [21]. At a macroscopic scale, these complex non-

[21]: Lambe (1958), 'The engineering behavior of compacted clay'

[7]: Gens et al. (1992), ‘A framework for the behaviour of unsaturated expansive clays’

[22]: Horpibulsuk et al. (2011), ‘Compressibility and permeability of Bangkok clay compared with kaolinite and bentonite’

[23]: Pusch (1980), ‘Permeability of highly compacted bentonite’

[24]: Norrish (1954), ‘The Swelling of Montmorillonite’

[9]: Yong (1999), ‘Soil suction and soil-water potentials in swelling clays in engineered clay barriers’

[25]: Anandarajah et al. (2013), ‘Discrete-element study of the swelling behaviour of Na-montmorillonite’

[26]: Yao et al. (2003), ‘Three-Dimensional Discrete Element Method of Analysis of Clays’

[27]: Katti et al. (2009), ‘Multiscale modeling of swelling clays: A computational and experimental approach’

[28]: Amarasinghe et al. (2012), ‘Behavior of Swelling Clays: A Molecular Dynamic Study’

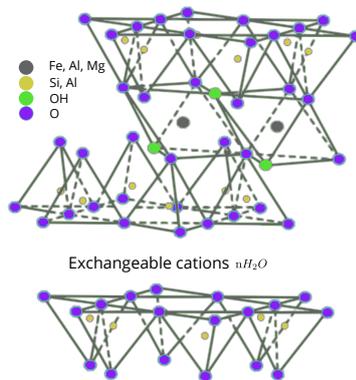


Figure 2.1: Layers of montmorillonite with interlayer cations and water molecules.

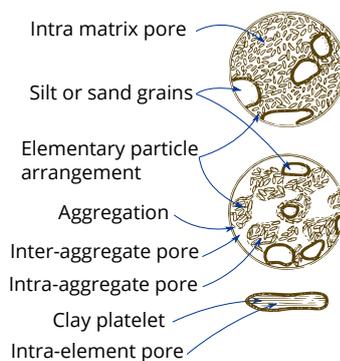


Figure 2.2: Diagram of clay structure and porosity scales [7]

linear microscopic processes contribute to non-linear compressibility [7, 22], and the non-linear evolution of permeability [23]. Typically, engineers and researchers consider their scale of interest and build empirically based models to simplify their prediction tasks. In the context of this thesis, we introduce some of these microscopic processes to help the reader understand the physical underpinning of various empirical models presented herein.

Some clay types have the propensity to volumetrically swell when presented with water. At a particulate level this is referred to as “crystalline” and “osmotic swelling”. [24] and [9] discuss how the smallest scale of swelling, called “crystalline swelling”, occurs in the inter-layer porosity and is a result of the adsorption of water (Fig. 2.1). On the other hand, the intra- and inter-aggregate porosities (Fig. 2.2) enable “osmotic swelling” leading to double-layer repulsion caused by electrostatic interactions between the negative surface charge and the ions in the electrolyte. For a sense of particulate scale, these pores measure between $0.002 \mu\text{m}$ and $0.0035 \mu\text{m}$ [9]. These nano-scale physics are well studied at the particulate and aggregate level. For example, [25], [26], and [27] used discrete elements to study the swelling behavior in clay particle packings by computing the inter layer forces including mechanical forces, attractive forces (van der Waals), and repulsive forces (as a function of electrolyte concentrations, cation valence, and surface potential). Meanwhile, [28] employed molecular dynamics to obtain pressure layer displacement curves (i.e. swelling potentials). Clearly, the particulate level understanding is mature. Despite this fundamental understanding, it is still not computationally feasible to account for billions and trillions of non-spherical particles. Thus, it is necessary to build models that abstract away from the nano-scale, while maintaining the peculiar non-linear hydro-mechanical behaviors that derive from these nano-scale chemical and mechanical swelling processes.

The Barcelona Basic Model (BBM) was the first self-contained and consistent macro-scale model for expansive partially saturated clay [29]. Simply put, the BBM decomposes volumetric deformation into volumetric change due to net stress and suction. The model is parametrically intensive; it requires the initial stresses, deviatoric stresses, initial specific volumes, all strain hardening parameters, initial position of the yield surfaces, the compressibility coefficients for loading/unloading, and cohesion as a function of suction. This extensive use of state variables and yield curves is not uncommon in unsaturated soil mechanics [30] because it captures the macroscopic behavior well. However, the original BBM does not account for the “double-structure” of the material, which is the idea that macroscopic clay behavior is controlled by inconsistent micro- and macro-structure hydro-mechanical behaviors (an idea well supported by the particulate/aggregate level physics). As described by [7, 11, 20], this “double-structure” advancement of BBM decouples the basic swelling of the clay particles from the macro-structural particulate rearrangements. From a mass transport perspective, the double-structure formulation [31] is relatively straightforward since it simply tracks the flux of water between macro and micro-structural levels using two separate water retention curves to inform a Darcy-flow [32].

2.2.2 Role of suction

The concept of “suction” plays an important role in the present thesis, since it drives mass fluxes and controls empirical models such as volumetric swelling and pellet stiffnesses. In short, suction is a quantification of the tendency for unsaturated clay and soil to absorb water [33]. In many ways, suction is simply a negative fluid pressure which drives fluid fluxes. However, it is much more complicated than this. The origin of suction in clay is particularly important because it is a combination of chemical interactions and particulate structure. For example, the “matric” suction is the suction present in a granular matter due to the attraction of water to climb into small pore throats (capillary rise), due to the surface tension of water [34]. On the other hand, the chemical component of suction, “osmotic” suction, derives from the presence of dissolved solutes in pore water, which interact with the clay particles as shown in Fig. 2.1 [9].

In the context of compacted bentonite clay used for engineered barriers, the total suction can exceed values of 132 MPa (the initial suction for dried compacted clay pellets studied herein). Such high magnitudes can be attributed to the osmotic component, but the nano-scale pore diameter cannot be neglected for the matric suction component. Models presented within depend strongly on the measurement of such suctions, which are typically performed by controlling relative humidity with salt concentrations [14, 35]. These suction values are used to build empirical models for volumetric swelling [17], water retention [14], and mechanical stiffness [36]. The present thesis uses these empirical observations as the foundation of micro-macro scale conceptual models.

2.2.3 Role of heterogeneity

Despite a mature field for modeling the hydro-mechanical behaviors of partially saturated clay, and a fundamental understanding of the particle level physics, there remain a variety of unknown processes that play important roles for understanding the hydro-mechanical behaviors at the macroscopic scale. For example, [37] shows how heterogeneities¹ in partially saturated soils have an important effect on gas permeabilities. Further, [38] concluded that discrete crack paths control permeability in gas migration. From a mechanical perspective, [39] concluded that heterogeneous pellet-powder mixtures behave mechanically in an uncertain way and require more investigation. To complicate matters, microscopic crack network developments in compacted clay pellets may also contribute to macroscopic hydraulic performance, as visualized by [14] (Fig. 1.2).

Take-away

In the consideration of engineered barriers comprised of pellet-powder mixtures, it is clear that heterogeneity *must* be considered for accurate long-term performance assessment.

[29]: Alonso et al. (1990), ‘A constitutive model for partially saturated soils’

[30]: Tarantino (2007), ‘A possible critical state framework for unsaturated compacted soils’

[7]: Gens et al. (1992), ‘A framework for the behaviour of unsaturated expansive clays’

[11]: Gens et al. (2011), ‘Hydromechanical behaviour of a heterogeneous compacted soil: experimental observations and modelling’

[20]: Alonso et al. (1999), ‘Modelling the mechanical behaviour of expansive clays’

[31]: Alonso et al. (2011), ‘Hydromechanical behaviour of compacted granular expansive mixtures: Experimental and constitutive study’

[32]: Mokni (2016), ‘Analysis of hydro-mechanical behaviour of compacted bentonite/sand mixture using a double structure formulation’

[33]: Ridley (2015), ‘Soil suction — what it is and how to successfully measure it’

[34]: Pinder et al. (2008), *Essentials of multiphase flow and transport in porous media*

[9]: Yong (1999), ‘Soil suction and soil-water potentials in swelling clays in engineered clay barriers’

[14]: Molinero-Guerra (2018), ‘Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture’

[35]: Romero et al. (1999), ‘Water permeability, water retention and microstructure of unsaturated compacted Boom clay’

[17]: Darde et al. (2018), ‘Comportment hydromecanique de pellets de bentonite: caracterisation au laboratoire et simulations DEM swelling’

[36]: Darde et al. (2020), ‘Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state’

1: Heterogeneities can refer to any type of discontinuity, including porosity changes, crack networks, and mineral disturbances

[38]: Olivella et al. (2008), ‘Gas flow through clay barriers’

[39]: Imbert et al. (2020), ‘Hydro mechanical behaviour of a heterogeneous swelling clay material’

2: Permeability is a material property which describes the ease of fluid fluxes through a material.

[5]: Barnichon et al. (2011), 'The SEALEX in situ experiments: performance tests of repository seals'

[40]: Martino et al. (2006), *The Tunnel Sealing Experiment : The Construction and Performance of Full Scale Clay and Concrete Bulkheads at Elevated Pressure and Temperature*

[41]: Dixon et al. (2014), 'Enhanced Sealing Project (ESP): evolution of a full-sized bentonite and concrete shaft seal'

[42]: Mokni et al. (2016), 'Effect of technological macro voids on the performance of compacted bentonite/sand seals for deep geological repositories'

[13]: Molinero-Guerra et al. (2017), 'In-depth characterisation of a mixture composed of powder / pellets MX80 bentonite'

[43]: Wan et al. (2018), 'Crack Characteristic and Permeability Change of Compacted Clay Liners with Different Liquid Limits under Dry-Wet Cycles'

[44]: De Camillis et al. (2016), 'Hydraulic conductivity and swelling ability of a polymer modified bentonite subjected to wet-dry cycles in seawater'

[45]: Didier et al. (2000), 'Gas permeability of geosynthetic clay liners'

[46]: Liu et al. (2015), 'Gas permeability of a compacted bentonite-sand mixture: Coupled effects of water content, dry density, and confining pressure'

[47]: Wei et al. (2019), 'Influences of degree of saturation and stress cycle on gas permeability of unsaturated compacted Gaomiaozi bentonite'

[48]: Liu et al. (2020), 'Laboratory investigation on gas permeability of compacted GMZ bentonite under a coupled hydraulic-mechanical effect'

[49]: O'Sullivan (2011), *Particulate discrete element modelling: a geomechanics perspective*

[50]: Cundall et al. (1979), 'A discrete numerical model for granular assemblies'

2.2.4 History of permeability evolution in engineered clay barriers

Modeling permeability² and deformation evolution in partially saturated heterogeneous swelling clay remains one of the most challenging topics in geomechanics, due in large part to the complicated non-linear interactions between swelling, porosity, and water retention. These coupled processes produce an unintuitive evolution of macroscopic permeability during hydration, which is of paramount importance in geo-environmental engineering applications such as nuclear waste buffers [6–12].

Despite encouraging research on the efficacy of engineered barriers at the Underground Research Laboratory (URL) [5, 40–42], there remain unanswered questions surrounding the evolution and sustainability of the water and gas permeability. For example, [13] uncovered the development of crack networks during swelling in partially saturated partially confined/unconfined conditions. These discrete crack networks might contribute to gas and water permeability changes during the life-cycle of the buffer, yet they cannot be modeled easily using the traditional Darcy's law. In fact, the development of discrete cracks in highly unsaturated conditions (high suction) may block the migration of water due to low gas entry pressures which favor the migration of gas. Various studies support this theory and conclude that these crack networks depend on many factors including the liquid limit [43] and wetting kinetics [44]. A similar phenomenon was shown by [45] for partially saturated clay; permeability decreased by up to two orders of magnitude between 40% and 70% saturation. Other studies demonstrated the sensitivity of partially saturated clay gas permeability to increasing confining pressure [46–48].

Based on these studies, it is clear that a gap still exists in the literature for understanding how these micro- macro-scopic processes evolve and how they affect the evolution of permeability in engineered barriers.

2.3 Discrete Element Method

The Discrete Element Method (DEM) is a popular method used to discretize the time-integration of Newton's 2nd law in particulate media. Discontinuous in nature, DEM has gained particular popularity in geomechanics due to its efficacy for modeling large deformation in granular materials [49]. In simple terms, this Lagrangian method represents the mechanical behavior of a particulate system as a collection of interacting masses, where interactions between masses follow predefined force-displacement laws [50]. In its most simplistic form:

$$\mathbf{M}\ddot{\mathbf{x}} = \mathbf{f} \quad (2.2)$$

with $\ddot{\mathbf{x}}$, the vector containing each particle acceleration, \mathbf{M} , the diagonal matrix of particle masses, and \mathbf{f} , the vector containing the total forces applied on the particles. The explicit central finite difference time stepping

scheme integrates the particle acceleration from the current step to update the particle position at the next step (see [51] for details of the practical implementation). The inter-particle forces, \mathbf{f}_{ij} , depend on a contact model, F_{ij} , such that:

$$\frac{\partial \mathbf{f}_{ij}}{\partial t} = F_{ij}(\mathbf{x}_i, \mathbf{x}_j, \dot{\mathbf{x}}_i, \dot{\mathbf{x}}_j) \quad (2.3)$$

where this contact model can be as simple and linear as Hooke's law (Fig. 2.3), or as complex and non-linear as Hertz law with variable stiffness parameters (Sec. 4.2.1). This flexibility, coupled with the discontinuous nature of the Lagrangian particulate system, has lent itself well to soil [52], rock [53, 54], and concrete [55].

The success of DEM in geomechanics opened up additional interest augmenting the method to simulate poromechanical behaviors.

Although some studies have effectively embedded capillary forces into the contact laws (Eq. 2.3)[56], others developed pore-scale schemes to model fluid fluxes.

2.4 Numerical methods for poromechanics in granular media

The Discrete Element Method was coupled with a fluid model to generate a sub-field of "pore-scale" poromechanics [57]. Entitled "Pore-Finite Volume" (PFV), the method is based on triangulating the DEM particles to create an interconnected set of pores where the conservation of mass and local pressure gradients govern fluid fluxes³. This hydro-mechanical coupling is highly efficient and capable of simulating accurate poromechanical processes at a fraction of the computational price of traditional Finite Element + CFD couplings [57]. In addition to the computational efficiency, PFV can replicate non-linear poroelastic drainage processes such as those observed in the oedometric conditions [58]. Beyond standard poroelasticity, PFV was extended for other DEM poromechanical applications such as hydraulic fracturing [59], multi-phase flow [60–62], and even thermo-hydro-mechanical applications [63]. These PFV extensions demonstrate the flexibility of the theoretical framework, as well as its robust practical implementation. With an end goal of modeling permeability and deformation evolution in engineered barriers, the present thesis builds upon the existing pore-scale method to discretize two novel conceptual hydro-mechanical models.

From a continuum perspective, some numerical methods leveraged macroscopic permeability and double structure models to successfully model macroscopic clay mechanics and the evolution of relative humidity during hydration of bentonite-sand mixtures ([32]). However, the effect of microscopic discontinuities, such as crack developments, on macroscopic gas/water permeability remains poorly understood [14, 64, 65].

[51]: Šmilauer et al. (2015), 'DEM Formulation, Release Yade documentation'

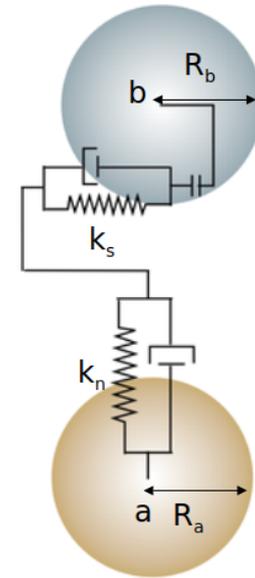


Figure 2.3: A simplistic representation of normal, k_n , and shear, k_s , stiffnesses in Hooke's law used to estimate forces between two particles, a and b

[56]: Scholtès et al. (2009), 'Micromechanics of granular materials with capillary effects (DOI:10.1016/j.jjengsci.2008.07.002)'

[57]: Chareyre et al. (2012), 'Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings'

³: variations of this method are described in detail in Chap. 4, Chap. 3, Chap. 5

[58]: Catalano et al. (2014), 'Pore-scale modeling of fluid-particles interaction and emerging poromechanical effects'

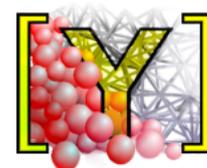


Figure 2.4: Yade open source DEM software

[32]: Mokni (2016), 'Analysis of hydro-mechanical behaviour of compacted bentonite/sand mixture using a double structure formulation'

[14]: Molinero-Guerra (2018), ‘Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture’

[64]: Péron et al. (2007), ‘An improved volume measurement for determining soil water retention curves’

[65]: Liu et al. (2014), ‘Experimental research on water retention and gas permeability of compacted bentonite/sand mixtures’

[66]: Smith et al. (2013), *Programming the finite element method*

[71]: Chen (2005), *Matrix preconditioning techniques and applications*

Instead, the novel conceptual models presented in Chap. 4 and Chap. 3 here address the complexity of these discontinuous micro- and macroscopic processes by:

- ▶ modelling microscopic crack processes directly
- ▶ considering initial heterogeneity distributions
- ▶ defining local water retention and swelling curves
- ▶ connecting the micro-scale to the macro-scale

2.5 Computational limitations in poromechanical modeling

Although there exists a strong body of academic work using the discontinuous pore-scale model for gaining insights into the poromechanical behavior of geomaterials [58–63], the framework is still limited by the time-dependent implicit solution of the pressure field. In fact, the practical implementation presented within this thesis (Chap. 3) suffers from this exact problem. Luckily, there exists a variety of popular methods aimed at alleviating the ailment. For example, the solution of linear systems of equations is usually accelerated depending closely on the sparsity and symmetry of the “system matrix” matrix (i.e. “stiffness matrix” for typical FEM systems). In most linear FEM cases, for instance, the system is sparse and symmetric, but the system matrix requires the solution of a linear problem at each time-step [66]. Typically, parallelizable iterative solvers, such as conjugate gradient, can be employed in a shared memory model with OpenMP [67], a distributed memory model with MPI [68], or on a graphical processing unit (GPU) [69, 70]. These system matrices are generally preconditioned to accelerate the solution [71]. Meanwhile, there exists a more traditional set of solvers called “direct solvers”, which employ matrix factorization methods as an alternative solution. This method, as explained in detail in Chap. 5, enables the reuse of a single factorization for multiple right hand sides if the system is defined by the same conductivity matrix over multiple time iterations [72]. Additionally, if the rank change of the conductivity matrix is low, the solution can be found more rapidly by updating/downdating the factor [73].

This wide variety of acceleration methods are harnessed in Chap. 5 for accelerating the same time-dependent implicit flow problem that presents itself in Chap. 3. From an applied perspective, these acceleration techniques enable larger and longer hydration simulations for compacted clay in engineered barriers.

MODEL DEVELOPMENTS

A conceptual model for micro-scale crack developments in partially saturated compacted clay pellets

3

3.1 Summary

A novel numerical method is presented for modeling hydration, swelling, and crack developments in heterogeneous compacted clay. The proposed method combines the Discrete Element Method (DEM) with the Finite Volume method (FVM) to simulate the mechanical and hydraulic behavior of a partially saturated swelling clay medium such as bentonite. Within the framework, DEM points are triangulated for FVM to solve the transient partial saturation field. The FVM triangulation is initialized with XRay CT scan imagery to form a heterogeneous field of water retention properties. These heterogeneous properties yield the development of cracks during hydration, which increases gas permeability beyond 60% water saturation and blocks water flux until gas entry pressure is reduced. Water permeability evolution concludes past 90% water saturation when the gas crack entry pressure is increased leading to water permeability increases with water saturation. A convergence study is performed on the DEM domain resolution and demonstrates that 150k DEM points produces accurate results at the millimeter scale for a heterogeneous compacted clay specimen. In addition, a multi-scale investigation of the same specimen demonstrates the spatial influence of heterogeneities on permeability and crack patterns at various sub-scales. With application toward engineered barriers, the permeability of the numerical specimen is measured under unconfined and confined conditions to generate an expected range of values for field scale applications. Finally, the crack fabric is analyzed and demonstrates a preferential orientation of cracks orthogonal to the direction of clay compaction.

Some of the contents of this chapter are in review at *Granular Matter*

R. A. Caulk, N. Mokni, and B. Chareyre. 'Modeling the transience of partial saturation and cracks in heterogeneous swelling clays using the Discrete Element and Finite Volume methods'. In: *Granular Matter* ()

3.2 Introduction

The following chapter presents a conceptual model designed to capture the evolution of hydro-mechanical processes resulting from the development of heterogeneities in compacted swelling clay during hydration at the micro-scale. In the context of the present thesis, this model is geared toward understanding the evolution of permeability in MX80 bentonite pellets, which are used in engineered barriers for nuclear waste containment. Beyond the conceptual model, an open-sourced practical implementation is presented with an in-depth analysis of mesh-resolution analysis, scale-dependence, crack developments, and water/gas permeability anisotropy in various confinement conditions. The chapter finishes by drawing conclusions about the evolution of permeability in compacted

swelling clays during hydration due to initial heterogeneities and crack developments.

3.3 Methods

3.3.1 Mechanical Model

The current application of the Lagrangian Discrete Element Method [50] is presented from a different perspective: as a discretization of space into discrete points. This means that the “particles” are no longer representing particles, instead they simply represent the numerical mesh. Each point is characterized by a mass, stiffness, and an *interaction radius*. Similar to Smooth Particle Hydrodynamics (SPH), the velocity of each individual point depends on a neighborhood surrounding it. But different from SPH, there is no “kernel function” accounting for the influence of all points on all other points. Instead, the velocity of a single point only depends on immediate interacting neighbor velocities. The most important reason for this “simplification” is that it enables direct contributions of heterogeneities, such as cracks, as is discussed in Sec. 3.3.6.

Individual point movements are governed by Newton’s second law of motion:

[80]: Smilauer et al. (2021), *Yade documentation 3rd ed.*

$$\ddot{\mathbf{x}} = \frac{\mathbf{f}}{\mathbf{m}} \quad (3.1)$$

where \mathbf{x} , \mathbf{f} , and m are the point position, total force, and mass, respectively. The DEM software employed for the present work, called Yade [80], uses an explicit time stepping scheme to compute each point’s position using its previous velocity and current acceleration. The contact law (Eq. 2.3) for the present model follows an un-damped modified Hertz-Mindlin contact theory [75, 76]:

$$\mathbf{f}_n = \frac{4}{3} E^* R^{1/2} d_n^{3/2} \mathbf{n}_n \quad (3.2)$$

where \mathbf{f}_n is the normal force, $E^* = \frac{E_a E_b}{E_b(1-\nu_a^2) + E_a(1-\nu_b^2)}$ with $E_{a/b}$ being the micro Young’s modulus of the two materials, d_n is the displacement between the two points, relative to their equivalent interacting radii, and \mathbf{n}_n is the unit vector parallel to the interaction. R is the equivalent interaction radius of particle a and particle b , $r_a r_b / (r_a + r_b)$, as shown in Fig. 3.1.

Shear force considerations follow a similar elastic Hertzian approach without a friction threshold that depends on the orientations of both particles in a local coordinate system:

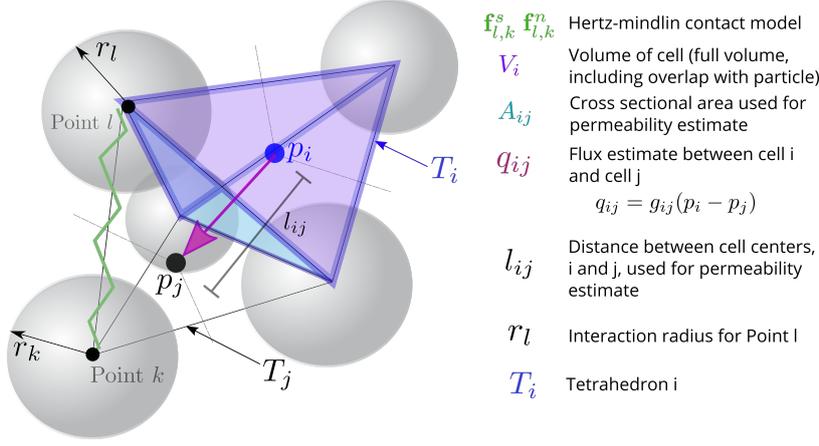


Figure 3.1: Visualization of the flow model and DEM geometries. Points spaced out to aid with labelling. Contact forces only occur for interacting radii.

$$\Delta \mathbf{f}_s = 2\sqrt{4R} \frac{G}{2-\nu} \Delta d_s \cdot \mathbf{n}_s \quad (3.3)$$

$$\mathbf{f}_s = \mathbf{f}_{s,\text{prev}} + \Delta \mathbf{f}_s \quad (3.4)$$

$$(3.5)$$

where G is the average shear modulus of both points (e.g. $G_a = E_a / (2(1 + \nu_a))$), ν is the average Poisson ratio between the two interacting DEM points. \mathbf{n}_s is the unit vector orthogonal to the DEM point interaction. Δd_s is the tangential increment of displacement, which depends on the relative angular velocities, $\dot{\mathbf{x}}_{rel}$, of the two points:

$$\dot{\mathbf{x}}_{rel} = (\dot{\mathbf{x}}_a + \omega_a \times (\mathbf{x}_{ab} - \mathbf{x}_a)) - (\dot{\mathbf{x}}_b + \omega_b \times (\mathbf{x}_{ab} - \mathbf{x}_b)) \quad (3.6)$$

$$\Delta d_s = \dot{\mathbf{x}}_{rel} - (\mathbf{n}_s \cdot \dot{\mathbf{x}}_{rel}) \mathbf{n}_s \quad (3.7)$$

where \mathbf{x}_{ab} is the midpoint between interacting points a and b . ω is the angular velocity.

3.3.2 Partially saturated flow model

Yade's Finite Volume scheme, first introduced by [57], is a Eulerian approach to solving the Stokes equations coupled with the Discrete Element Method. All discrete element point locations are triangulated to form a fluid network which enables the implicit solution of a water/-gas pressure field as well as hydro-mechanical force approximations. Within the partially saturated finite volume model presented here, each tetrahedron of the triangulation (Fig. 3.1) represents a partially saturated volume characterized by a unique porosity, saturation, suction, and water retention parameters. Different from the original implementation by [57], the current partially saturated scheme does not separate the mechanical and fluid domains. Instead, a dual domain approach is adopted where triangulated tetrahedra represent the clay domain (as shown in Fig. 3.1) and the mechanical behavior is controlled by the discrete element point interactions as described in Sec. 3.3.1.

[57]: Chareyre et al. (2012), 'Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings'

The partially saturated flow domain is governed by Stokes equations, assuming small Reynolds and Stokes numbers:

$$\nabla p = \mu \nabla^2 \mathbf{u} - \mathbf{f} \quad (3.8)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (3.9)$$

where \mathbf{u} is the water velocity and p is the piezometric pressure, μ is the water dynamic viscosity, and \mathbf{f} is an applied body force. Using the divergence theorem, the surface integral of Eq. 3.9 can be represented as an integral of the water and contour velocities for full saturation:

$$\int_{\partial\Omega_i} (\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS = 0 \quad (3.10)$$

where Ω_i is the contour of the tetrahedron, \mathbf{u} is the water velocity and \mathbf{v} is the contour velocity. \mathbf{n} is the outward pointing vector.

The contour integral can be reduced to a sum of integrals over the tetrahedron facet areas (A_{ij}) equivalent to the saturation change for the volume, V_i :

$$\sum_{j=1}^4 \int_{A_{ij}} (\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS = -V_i \frac{ds_{w,i}}{dp_i} \frac{dp_i}{dt} \quad (3.11)$$

where V_i is the volume of tetrahedron i and $\frac{ds_{w,i}}{dp_i}$ is the partial derivative of the local water retention curve. The integral on the left-hand side of Eq. 5.5 represents the water flux and can be represented as the sum of fluxes exchanged by each tetrahedron and its four neighbors ($j=1$ to 4):

$$\sum_{j=1}^4 \int_{A_{ij}} (\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS = - \sum_{j=1}^4 q_{ij}. \quad (3.12)$$

Flux (q_{ij}) between tetrahedron, i , and its neighbor tetrahedrons, j , is assumed to be linearly proportional to the local pressure gradient:

$$q_{ij} = g_{ij}(p_j - p_i) \quad (3.13)$$

where g_{ij} is a local conductivity quantity discussed in detail below. Finally, the discretized equation reads:

$$\sum_{j=1}^4 g_{ij}(p_j - p_i) = -V_i \frac{ds_{w,i}}{dp_i} \frac{dp_i}{dt} \quad (3.14)$$

[78]: Van Genuchten (1980), 'A closed form equation for predicting the hydraulic conductivity of unsaturated soils.'

where p_i and p_j are the pressures of neighboring tetrahedrons, and $\frac{ds_{w,i}}{dp_i}$ (Fig. 3.2) follows a Van Genuchten [78] model:

$$s_w(p_i) = \left(1 + \frac{p_i}{\zeta(\phi)^{\frac{1}{1-\lambda(\phi)}}}\right)^{-\lambda(\phi)} \quad (3.15)$$

where the parameters ζ and λ are both functions of porosity as highlighted in Sec. 3.3.4¹. The conductivity, g_{ij} is a Darcy conductivity based on the distance between the two tetrahedron centers, l_{ij} , and the area of the shared facet A_{ij} (Fig. 3.1):

$$g_{ij} = \frac{k_{M,ij} k_{rM,ij} A_{ij}}{\mu l_{ij}} \quad (3.16)$$

where μ is the dynamic viscosity of the fluid and the residual relative permeability factor, $k_{rM,ij}$, is empirically related to the degree of saturation [32]:

$$k_{rM,ij} = \left(\frac{s_{ij,p} - s_{r,p}}{s_{s,p} - s_{r,p}}\right)^n \quad (3.17)$$

where $s_{ij,p}$ is the average phase saturation² between the incident tetrahedrons, $s_{r,p}$ is the residual phase saturation, $s_{s,p}$ is the saturated phase saturation, and n is a material parameter. Meanwhile, $k_{M,ij}$, intrinsic permeability, is an exponential function of the average porosity between incident tetrahedrons i and j at time t , ϕ_{ij}^t [11]:

$$k_{M,ij} = k_o \exp(b_k(\phi_{ij}^t - \phi_{o,ij})) \quad (3.18)$$

where k_o is the reference permeability at the reference porosity ϕ_o and b_k is a calibrated factor.

After the pressures are solved, s_w is explicitly updated similar to [62]:

$$s_w = s_w^{t-1} + \frac{ds_w}{dp_i}(p_i - p_i^{t-1}) \quad (3.19)$$

where the superscript $t - 1$ refers to the previous time step. Since the PFV triangulation represents the saturation field of the entire clay domain, discrete element points adopt the weighted average saturation of incident tetrahedrons. This weighted average saturation is used with localized water retention curves (Fig. 3.2) to determine the suction to be used for the volumetric strain model (Eq. 3.20).

3.3.3 Volumetric Swelling Model

The swelling of the pellets follow exponential models fit to experimental free swelling data by [14, 36]. As shown by Eq. 3.20, ε is estimated using an exponential variant:

1: This approach neglects the hysteresis commonly observed for water-retention in swelling clays.

[32]: Mokni (2016), 'Analysis of hydro-mechanical behaviour of compacted bentonite/sand mixture using a double structure formulation'

2: The gas saturation is used here for estimating the gas permeability in Sec. 3.5

[62]: Sweijen et al. (2018), 'Dynamic pore-scale model of drainage in granular porous media: the pore-unit assembly method'

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[36]: Darde et al. (2020), 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'

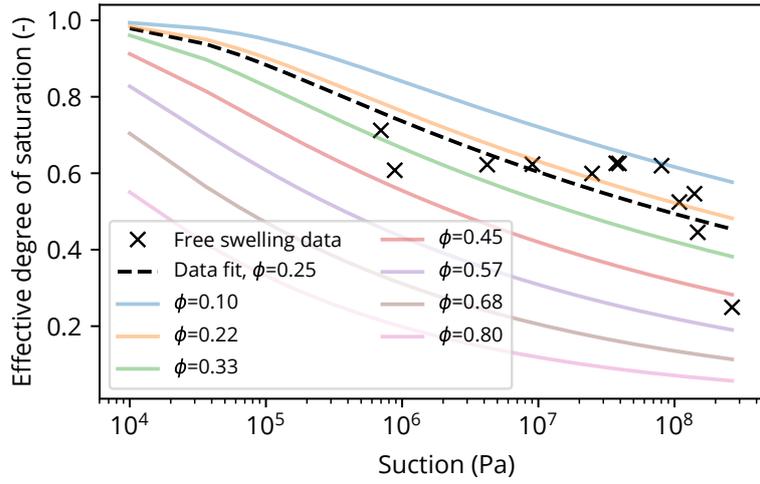


Figure 3.2: Pressure saturation curve for MX80 clay pellet [14] and the effect of changing porosity [32].

$$\varepsilon(s) = \int_{s_0}^s \beta \exp(-\alpha s) ds \tag{3.20}$$

$$\varepsilon(s) = \frac{\beta}{\alpha} [\exp(-\alpha s_0) - \exp(-\alpha s)] \tag{3.21}$$

where β , α are model parameters, s_0 is initial suction, and s is suction based on the weighted average of incident tetrahedra. Therefore, DEM point interaction radii, r , are adjusted as follows:

$$\Delta v = \varepsilon(s)v_o \tag{3.22}$$

$$\Delta r = \Delta v \left(\frac{3v}{4\pi} \right)^{1/3} \tag{3.23}$$

Where v_o is the original volume of each DEM point based on its interaction radius.

3.3.4 Assignment and evolution of porosity

The initial porosity of the tetrahedra can be assigned heterogeneously in various ways such as random assignment or extraction from X-ray CT imagery of a bentonite clay pellet (Fig. 3.3). The present study imaged a pellet at various suctions (Fig. 3.4) to infer densities from gray values by assuming a linear relationship between gray values and density. A linear model was fit by setting the highest gray values to the density of the densest mineral, muscovite (3.9 g/cm^3), and the lowest gray values (black) to the density of air (0.00113 g/cm^3). Next, a 3D grid was overlain on the imagery and volume average density is extracted for each grid point (based on mean DEM interaction radius). Finally, porosity is inferred from the density assuming a linear relationship between density and porosity. For example, grid regions with an average density equal to air, have a porosity of 1, while grid regions with an average density

equivalent to the experimentally measured initial average dry density of the pellet (2.12 kg/m^3) have a porosity equivalent to the experimentally measured average initial porosity of the pellet (0.25). As shown in Fig. 3.3, the initial porosity distribution for the pellet follows closely to imagery based gray values.

As interaction points swell/contract according to Eq. 3.20, the tetrahedra volume, V_i , expand or shrink accordingly which leads to an evolution of porosity:

$$V_{s,i} = (1 - \phi_0)V_{i,0} \quad (3.24)$$

$$\phi_i = 1 - \frac{V_{s,i}}{V_i} \quad (3.25)$$

where $V_{s,i}$ is the volume of solids in tetrahedron i and the subscript 0 refers to the values at the reference state.

Following [32], the water retention curve also evolves with porosity:

$$\zeta(\phi) = \zeta_o \exp(a(\phi_o - \phi)) \quad (3.26)$$

$$\lambda(\phi) = \lambda_o \exp(b(\phi_o - \phi)) \quad (3.27)$$

where ζ_o and λ_o are initial values for the parameters for the Van Genuchten [78] water retention curve. Fig. 3.2 shows the effect of porosity on the water retention curves.

3.3.5 Hydro-mechanical model

The hydro-mechanical coupling follows the original implementation by [57]; pressure and viscous forces acting on some particle, k , are a contour integration of the absolute pressure and viscous stress along the contour shared between the interacting point radius and incident ‘‘pore-throat’’³. A visualization of the ‘‘pore-throat’’ and the surfaces used for integration is shown in Fig. 3.5. This is admittedly *not a physical interpretation of the geometry of the system*, since the system is a discretization of points and the tetrahedrons are fully filled with clay material. However, the following force estimate is estimating relative forces by weighting the space consumed by individual points.

Discretizing the contour to the particular geometry shown in Fig. 3.5, and projecting the pressure onto the intersection of the shared facet with the interacting radius, the pressure force is simply:

$$\mathbf{f}_{ij}^{p,k} = A_{ij}^k (p_j - p_i) \mathbf{n}_{ij} \quad (3.28)$$

where \mathbf{n}_{ij} is the unit vector, A_{ij}^k is the intersection of facet area and discrete element point k (Fig. 3.5). However, different from [57], the

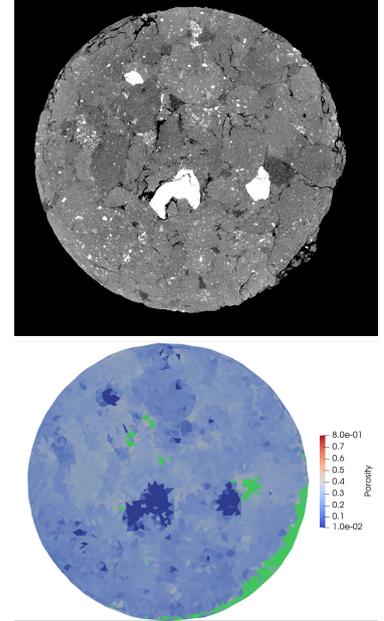


Figure 3.3: Initial porosity distribution comparison top) Xray CT image gray values, 7 mm diameter and pixel size $4.4 \mu\text{m}$ bottom) model porosity values with cracked cells in green.

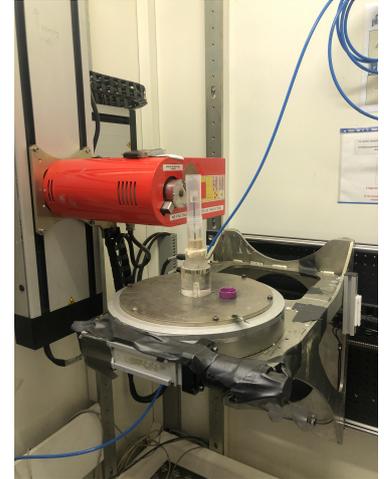


Figure 3.4: Pellet scanning in the tomograph of Laboratoire 3SR.

[57]: Chareyre et al. (2012), ‘Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings’

3: The ‘‘pore-throat’’ here is simply referring to the geometrical reduction of the space to estimate weightings for hydraulic forces. ‘‘pore-throat’’ is *not* referring to an actual pore-throat in the clay

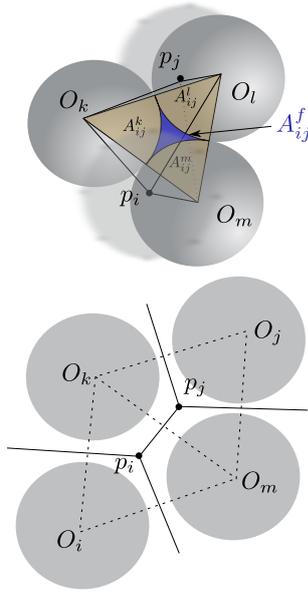


Figure 3.5: Geometrical areas used for pressure and viscous force integration.

present implementation considers the effect of partial saturation (s_w) of each cell so the force is factored to be:

$$\mathbf{f}_{ij}^{p,k} = A_{ij}^k (s_{w,j} p_j - s_{w,i} p_i) \mathbf{n}_{ij} \quad (3.29)$$

where p is the swelling pressure computed with Eq. 4.33. Thus, the swelling powder within the voids adds force to the discrete element particles, which contributes to the force balance on each of the DEM particles.

The viscous shear force is integrated similar to the pressure force above. As [57] points out, the total viscous force is a contour integral of the pressure along the area not consumed by interacting point radii of the shared facet, discretized to A_{ij}^f in Fig. 3.5. For each facet, the total viscous force, \mathbf{f}_{ij}^V , and the individual viscous force applied to each of the participating particles, $\mathbf{f}_{ij}^{v,k}$, is computed:

$$\mathbf{f}_{ij}^V = A_{ij}^f (s_{w,j} p_j - s_{w,i} p_i) \mathbf{n}_{ij} \quad (3.30)$$

$$\mathbf{f}_{ij}^{v,k} = \mathbf{f}_{ij}^V \frac{A_{ij}^k}{\sum_{x=k}^m A_{ij}^x} \quad (3.31)$$

The total force added to the the discrete element contact force summation (Eq. 3.1) becomes:

$$\mathbf{f}^k = \sum_{(ij)incident} \mathbf{f}_{ij}^{v,k} + \mathbf{f}^{p,k} \quad (3.32)$$

3.3.6 Crack model

The development of cracks during the hydration of compacted swelling clay can play an important role in the evolution of water and gas permeabilities [14, 64, 65]. In the present model, the uncracked permeability model follows a Darcy permeability model as discussed in Sec. 3.3.2, where each facet in the triangulation has a unique permeability value based on saturation and porosity. During hydration, a crack may develop if the normal force computed between two DEM points (Eq. 4.1, Fig. 3.6) reaches 0 and the criteria for the water entry pressure is no-longer satisfied in the incident tetrahedra i and j :

$$p_{ij} \geq \frac{\gamma}{u_{ab} - (r_a + r_b)} \quad (3.33)$$

where γ is the gas-water surface tension and u_{ab} is the distance between DEM points a and b . As soon as a crack opens, it is saturated by gas which prevents water from entering. Thus, the local conductivity quantity is reduced:

$$g_{ij} = \frac{k_M k_{rM} (\eta A_{ij})}{l_{ij}} \quad (3.34)$$

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[64]: Péron et al. (2007), 'An improved volume measurement for determining soil water retention curves'

[65]: Liu et al. (2014), 'Experimental research on water retention and gas permeability of compacted bentonite/sand mixtures'

where η is a roughness factor accounting for the fraction of the two crack faces that remain in contact with one another. The capillary forces released by the opened crack are removed from the DEM point summation (Eq. 3.1):

$$\mathbf{f}_{+a/-b} = \mathbf{x}_{ab} \frac{a_c(p_i + p_j)(s_i - s_j)}{2} \quad (3.35)$$

where \mathbf{x}_{ab} is the vector pointing from discrete element point a to point b . Both pressure ($p_{i/j}$) and saturation ($s_{i/j}$) are average values between the two incident tetrahedra i and j . And the crack area (a_c) shown in Fig. 3.6 is computed as:

$$a_c = \|(\mathbf{x}_i - \mathbf{x}) \times (\mathbf{x}_j - \mathbf{x})\| \quad (3.36)$$

where \mathbf{x}_i is the center of tetrahedron i , \mathbf{x}_j is the center of tetrahedron j , and \mathbf{x} is the mutually shared mid point along edge AB (Fig. 3.6). It follows that the crack volume is simply $a_v = a_c(u - (r_a + r_b))$.

Meanwhile, the gas permeability of the opened crack follows a parallel plate approximation (based on the cubic law, [59, 82]), effectively increasing the permeability of the facet:

$$g_{ij} = \frac{(u - (r_a + r_b))^3}{12\mu} \quad (3.37)$$

where μ is the dynamic viscosity of the phase.

The crack remains unsaturated by water as long as the following criteria remains satisfied:

$$p_{ij} < \frac{\gamma}{u - (r_a + r_b)} \quad (3.38)$$

at which point the water permeability of the crack is set according to Eq. 3.37 and the gas permeability is reduced using Eq. 3.34.

Finally, a 2nd order crack fabric tensor ($\mathbf{\Lambda}$) is computed to describe the orientation of the crack network [83, 84]:

$$\mathbf{\Lambda} = \frac{1}{N} \sum_{i=1}^N (\mathbf{n}^i \otimes \mathbf{n}^i) a_c \quad (3.39)$$

where \mathbf{n}^i is the unit vector pointing orthogonal to crack i (determined as described in Eq. 3.36).

[83]: Oda (1982), 'Fabric tensor for discontinuous geological materials'
[84]: Shertzer (2011), 'Fabric tensors and effective properties of granular materials with applications to snow'

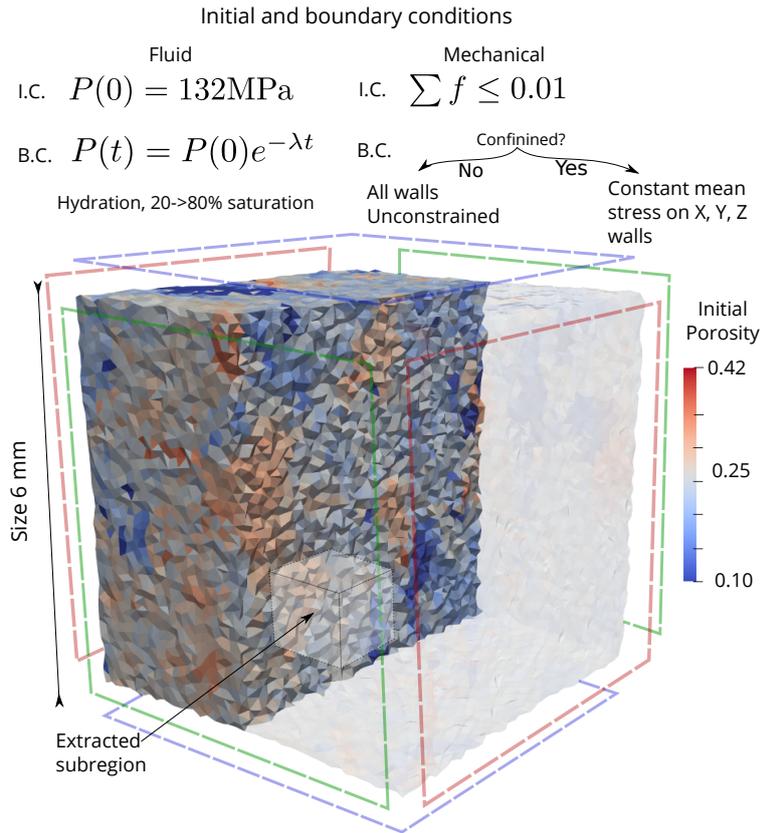


Figure 3.8: Model boundary conditions, initial conditions, and dimensions.

3.5.1 Model configuration

The methods presented in Sec. 3.3 are implemented in Yade open source DEM software and applied here in an example of the hydration of compacted clay powder. As shown in Fig. 3.8,

- ▶ Discrete element points are triangulated to form the clay domain.
- ▶ Boundaries are suction controlled and water/gas-permeable.
- ▶ The initial suction in the domain is set to 132 MPa.
- ▶ Imagery from the compacted clay pellet gathered from [14] is used to set the initial porosity heterogeneity as discussed in Sec. 3.3.4

Hydro-mechanical equilibrium is first achieved by setting an initial suction of 132 MPa to the exterior of the numerical specimen. After equilibrium, boundary suction is decreased at a rate of 10 Pa per simulation time-step ($1e-9$ s). All parameters for the DEM and FVM models are presented in Table 3.1.

3.5.2 Model calibration

The micro-parameters for water-retention, volumetric swelling, material stiffness, and Poisson ratio were calibrated using a variety of simulations.

- ▶ The macroscopic water-retention of the numerical specimen shown in Fig.3.2 was calibrated to macroscopic data collected by [14] for a single compacted clay pellet. This calibration was performed by

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

Table 3.1: Partially saturated flow model parameters

Parameter	Value	Equation, Figure, Reference
β	0.8e-8 Pa	Eq. 3.20, calibrated
α	2.102e-8 Pa	Eq. 3.20, [32]
ζ_o	0.03e6 Pa	Eq. 3.15 Fig. 3.2, calibrated
λ_o	0.08	Eq. 3.15 Fig. 3.2, calibrated
a	6.8	Eq. 3.26 Fig. 3.2, [32]
b	-1.5	Eq. 3.27 Fig. 3.2, [32]
ϕ_0	Xray CT imagery - ρ_d 2.12 Mg/m ³	Sec. 3.3.4, [14]
s_s	1	Eq. 3.17,-,[42]
s_r	0.0	Eq. 3.17,-,[42]
n	2	Eq. 3.17,-,[42]
b_k	2	Eq. 3.18,-,[42]
k_o	1e-20 m ²	Eq.3.18,-,[42]
γ	7.28e-2 N/m	Eq. 3.33, gas-water surface tension
η	0.001	Eq. 3.34, crack roughness factor
μ_g	1.8e-5 Pa-s	Eq. 3.37, gas dynamic viscosity
E	100e8 Pa	Eq. 4.1, mechanical stiffness microparameter, calibrated
ν	0.3	Eq. 4.1, mechanical Poisson ratio microparameter, calibrated
Δt	1e-9 s	time-step

hydrating the specimen in the same conditions as the experimental specimen, and measuring average saturation for each level of average macroscopic suction.

- ▶ Fig. 3.9 shows the macro-scopic volumetric swelling calibration for the microparameters found in Eq. 3.20. The simulation was the same process as the water-retention curve calibration - hydration of the specimen in the same conditions as the experimental data collected by [14].
- ▶ Fig. 3.9 shows the calibration of mechanical parameters, E and ν , to the elasticity parameters measured by [17] for a suction of 132 MPa. However, the elasticity parameters are static during hydration (despite the Hertz contact model which effectively increases stiffness with greater contact). Future advancements should consider the use of a suction dependent Young's modulus, as highlighted in Sec. 4.2.1.

3.5.3 Measuring permeability numerically

The macroscopic hydraulic conductivity of each phase, k_p , is estimated periodically during hydration by:

- ▶ applying a pressure gradient, ∇p , to the entire specimen
- ▶ extracting the water/gas velocity field
- ▶ integrating the volume (Ξ) to obtain a the total flux (q)

$$q = \int_{\Xi} \mathbf{u} dV \quad (3.40)$$

$$k_p = \frac{q \nabla p}{\mu_p} \quad (3.41)$$

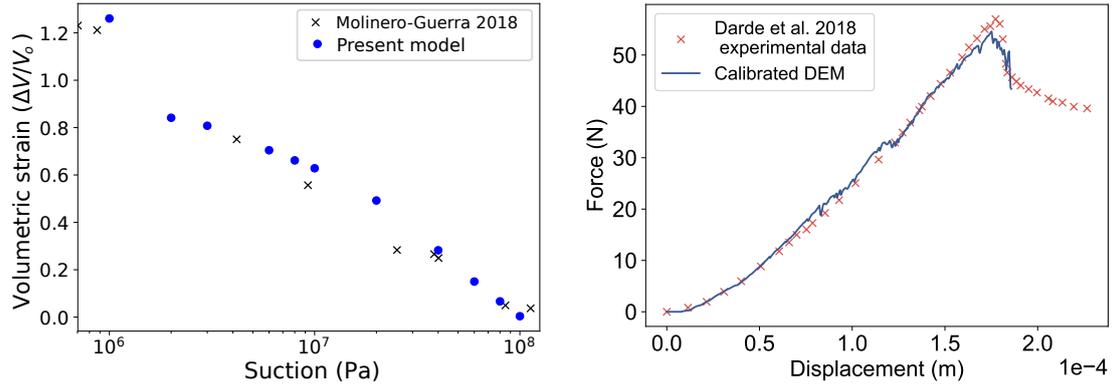


Figure 3.9: Left) Volumetric strain calibration. Black “x” showing data from [14] Right) Strength and stiffness calibration. Red “x” showing data from [17]

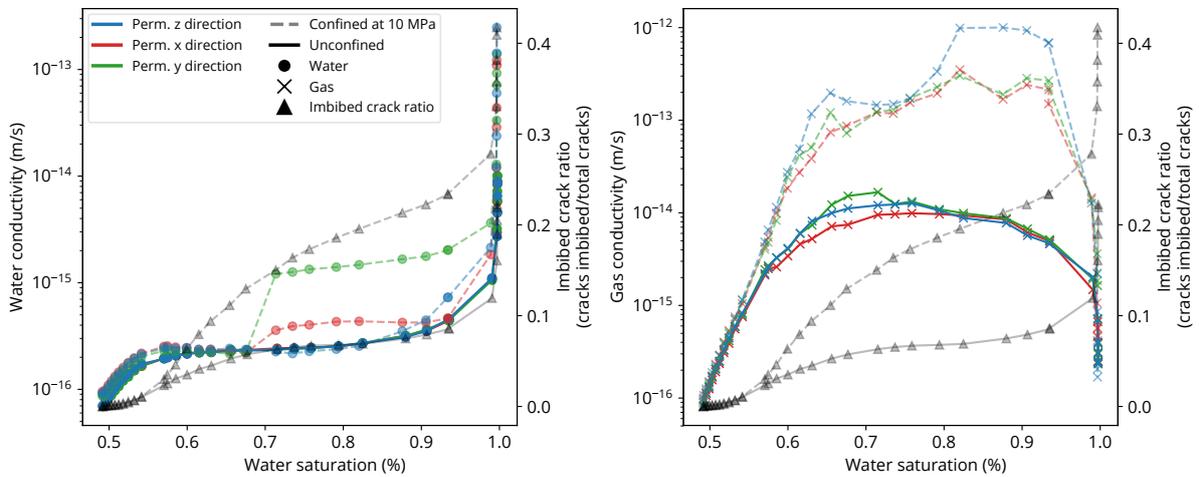


Figure 3.10: Evolution of gas and water conductivities with respect to volumetric saturation

As shown in Fig. 3.11, the water permeability increases with saturation and begins to stabilize at 60% saturation with cracks opening (Eq. 3.33) and blocking water flux through various regions (Fig. 3.14). At the same stage, the gas permeability increases monotonically with saturation due to opening of cracks and porosity increase. Following 90% saturation, the water permeability increases by two orders of magnitude and the gas permeability decreases by more than three orders of magnitude (Eq. 3.38). Further, the permeability anisotropy shows how the permeability in the Y and X axes are higher than the permeability in the Z axes (Z axis is parallel to compaction) by half of one order of magnitude. The permeability and crack fabric tensors are explored further in Sec. 3.5.6.

3.5.4 Effect of confining pressure on permeability evolution

A demonstration of the effect of confining pressure on permeability evolution is performed by applying a confining pressure of 10 MPa to the three axes shown in Fig. 3.8. As shown in Fig. 3.11, the permeabilities in the confined and unconfined cases remain the same until the confin-

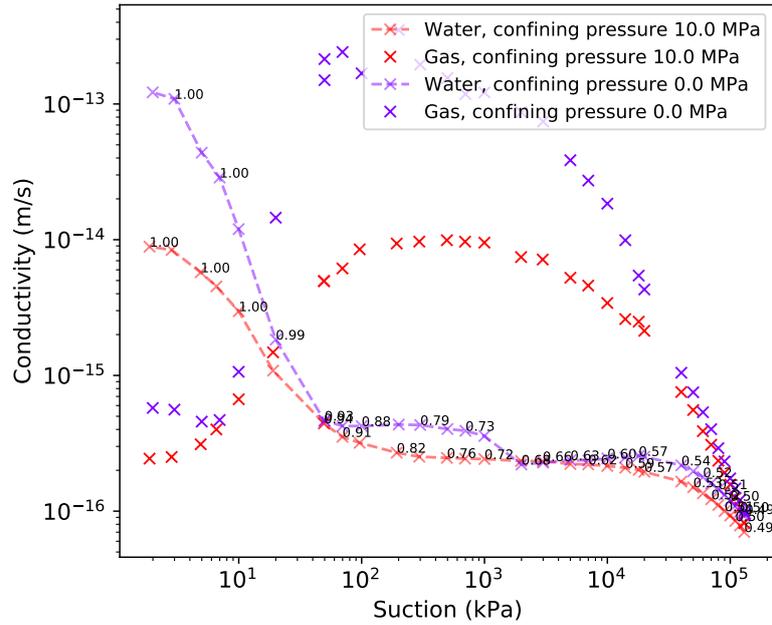


Figure 3.11: Effect of confining pressure on water and gas permeability evolution. Point labels indicate water saturation.

ing pressure is matched by the suction near 10 MPa (Fig. 3.11). Water permeability of the unconfined and confined specimen remains largely the same until a high saturation of 0.99 and low suction of 10 kPa are achieved. At which point, the porosity of the unconfined specimen continues to increase which increases the permeability by another order of magnitude greater than the confined specimen. The gas permeability, on the other hand, shows a larger disparity, up to 1.5 orders of magnitude, during the partially saturated state where saturation is between 0.54 and 1.00.

3.5.5 Effect of heterogeneity on crack developments

Heterogeneity is explored in the clay specimen by performing two hydration simulations on a cubic DEM point mesh: one starting with a heterogeneous porosity field informed using XRay CT scan imagery as discussed in Sec. 3.3.4 and the other starting with a homogeneous porosity field (Fig. 3.12). As shown in Fig. 3.12, the homogeneous specimen does not develop cracks during the entire hydration but the heterogeneous specimen yields a collection of cracks, indicating the necessity of heterogeneity. Thus, the development of cracks during hydration occurs due to the existence of heterogeneities and their influence on the evolution of the saturation and deformation fields.

3.5.6 Effect of crack model on permeability evolution

Fig. 3.13 shows the effect of the crack model by plotting the evolution of macroscopic permeability during imbibition with and without the crack model activated. As shown, without the crack model, the permeability monotonically increases as suction decreases (i.e. saturation increases). The introduction of the crack model reduces the computed permeability by almost 1/2 order of magnitude at a water saturation of 68%. This

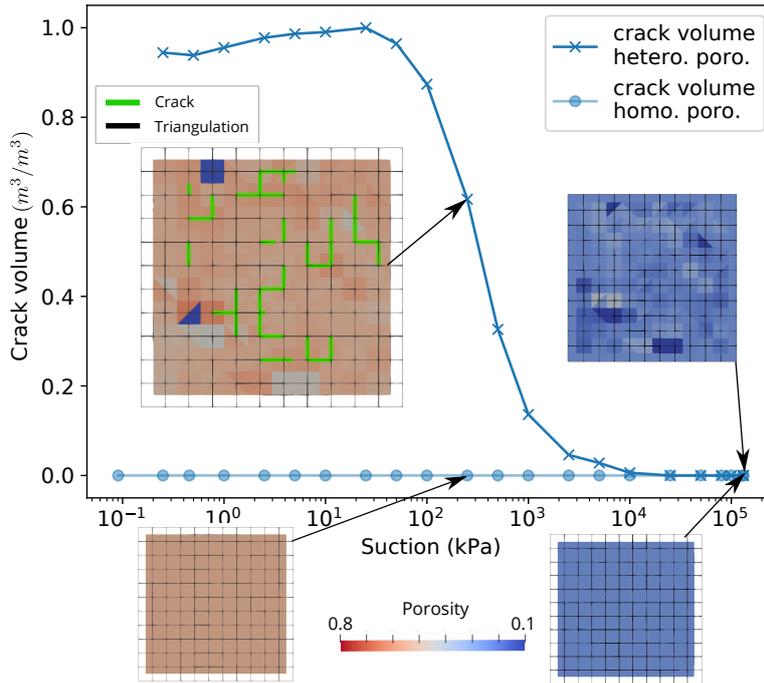


Figure 3.12: Comparison of the effect of homogeneous and heterogeneous initial porosity fields on the development of cracks during volumetric swelling.

phenomenon is due to the development of unimbibed cracks which restrict water fluxes as shown in Fig. 3.14. After 94% water saturation, the imbibed cracks accelerate water flux through the medium, exceeding the permeability computed for the non-crack model by two orders of magnitude at 100% water saturation. This upwards trend of permeability near full saturation also matches the experimentally observed permeability trend at high water content in bentonite-sand mixtures [87].

The evolution of the crack fabric tensor (Eq. 3.39) exhibits the magnitude of anisotropy of the crack network during hydration. As shown in Fig. 3.15, the magnitude of the deviatoric crack fabric tensor is highest at lowest saturation and decreases quickly during the early stage of hydration between 50-60% saturation levels. Beyond 75% saturation, the deviatoric component of the crack fabric remains constant, suggesting the total imbibed+non-imbibed crack network is no longer changing fabric beyond these levels of saturation. Contrary to the crack fabric, the deviatoric components of the gas and water permeability tensor begin as non-existent and then evolve with saturation. Since the cracks first open unimbibed (Sec. 3.3.6), the anisotropy of the gas permeability fabric starts increasing (Fig. 3.15A) until the same cracks start becoming imbibed. Once cracks start becoming imbibed, they stop contributing to the gas permeability tensor, and begin contributing to the water permeability tensor (Fig. 3.15B,C). This crack imbibition leads to a marked increase in the deviatoric component of water permeability at 70% saturation - which is likely due to the opening of preferential water pathways as the opened cracks become imbibed with water. Fig. 3.15B shows that the preferred direction for these imbibed cracks is the Y axis ⁴. Fig. 3.14 agrees as it shows the increasing effect of the imbibed crack volume, but this plot does not weight these imbibed volumes according to their contribution to over-all water permeability. Thus, it does not show the same marked increase in imbibed crack volume. At the same level of saturation, 70%,

4: This abrupt change is not reflected in the crack fabric of Fig. 3.15 because crack fabric here includes imbibed+non-imbibed cracks, further analyses should distinguish crack fabrics between imbibed and non-imbibed to confirm this conclusion

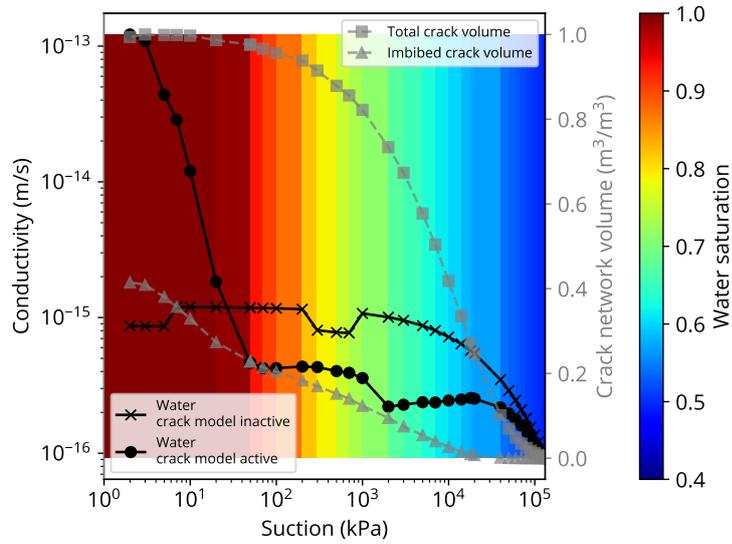


Figure 3.13: Effect of crack model on macroscopic permeability evolution during hydration.

Table 3.2: Comparison of porosity distribution evolution for experimental data and present model.

	Decrease of average porosity	
	132 MPa → 10 MPa	10 MPa → 1 MPa
MIP Data [14]	0.26	0.095
Present model	0.25	0.02

the anisotropy of the gas permeability reaches a minimum value as water is consuming the large pathways previously occupied by gas. Finally, volumetric swelling continues and opens up gaping unimbibed cracks (Fig. 3.15D) which contribute to toward the conclusion of saturation, the saturated domain begins to behave homogeneously as shown by both gas and water anisotropic permeability components trending toward 0.

3.5.7 Evolution of porosity distribution

The evolution of the porosity field during hydration is shown in Fig. 3.16 and reported by Tab. 3.2. In comparison to void ratios determined from MIP data collected by [14], the increase of average porosity between 132 MPa and 10 MPa suction is in agreement, around 0.25. However, the porosity increase near full saturation (between 10 MPa and 1 MPa suctions), is measured more than 4x compared to present model.

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

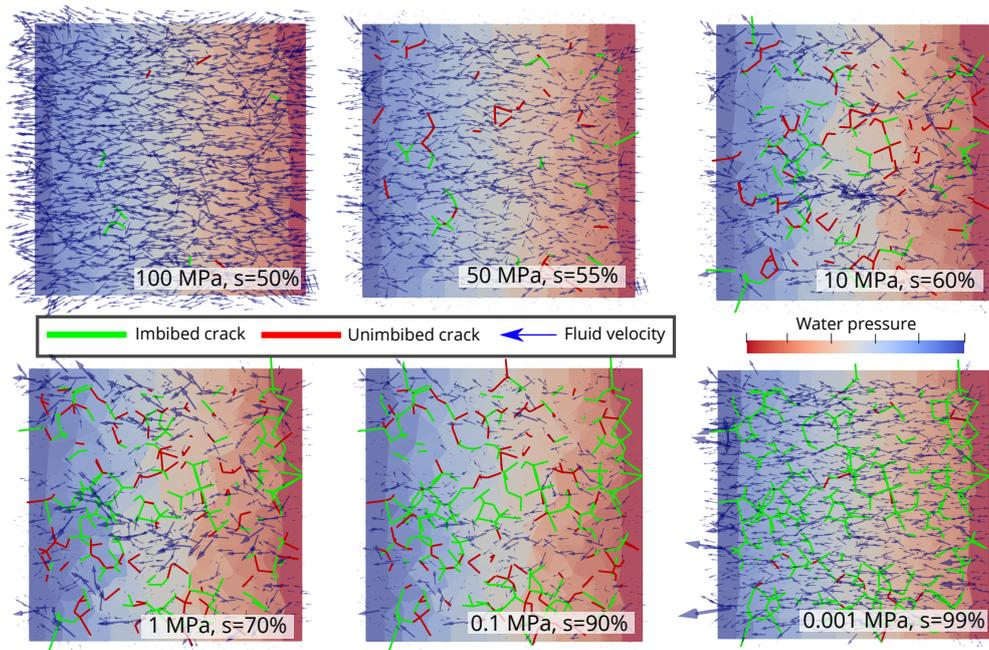


Figure 3.14: Effect of drained and imbibed cracks on water paths and velocities at varying levels of average water saturation.

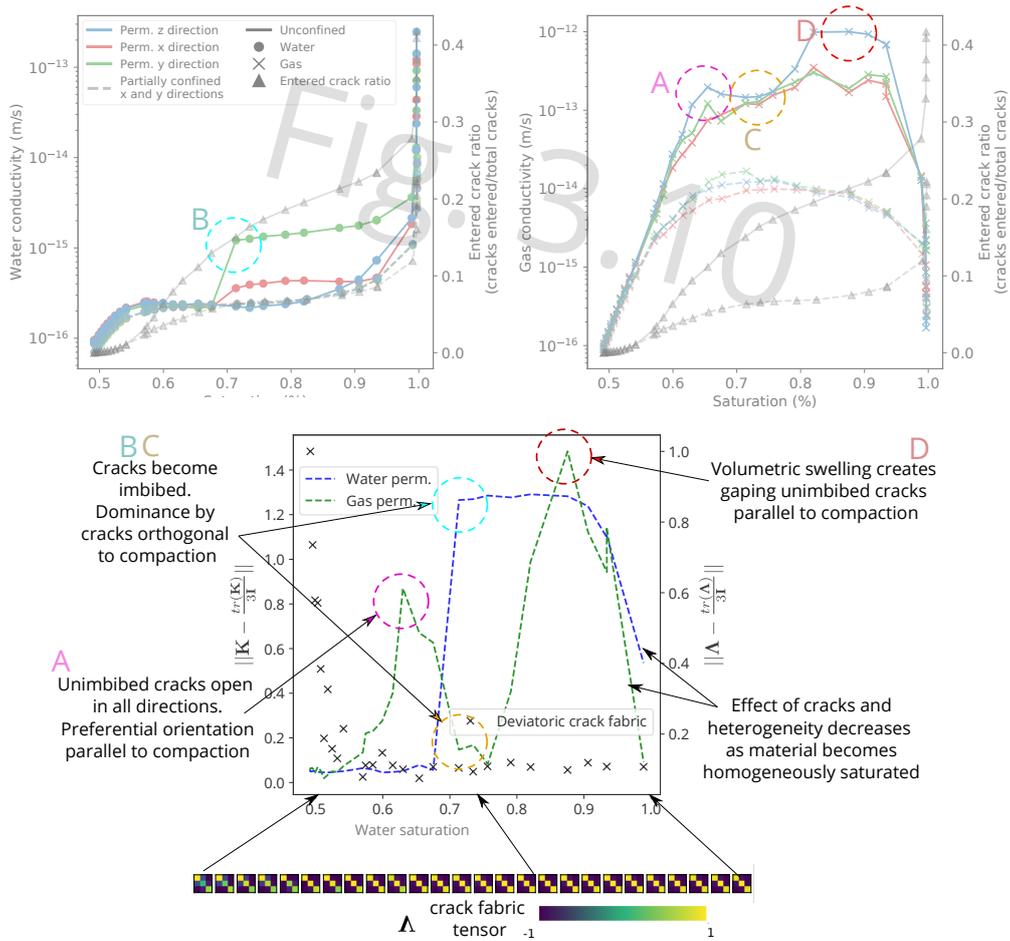


Figure 3.15: Evolution of crack and permeability fabric during hydration. Colored tiles represent the components of the crack fabric tensor.

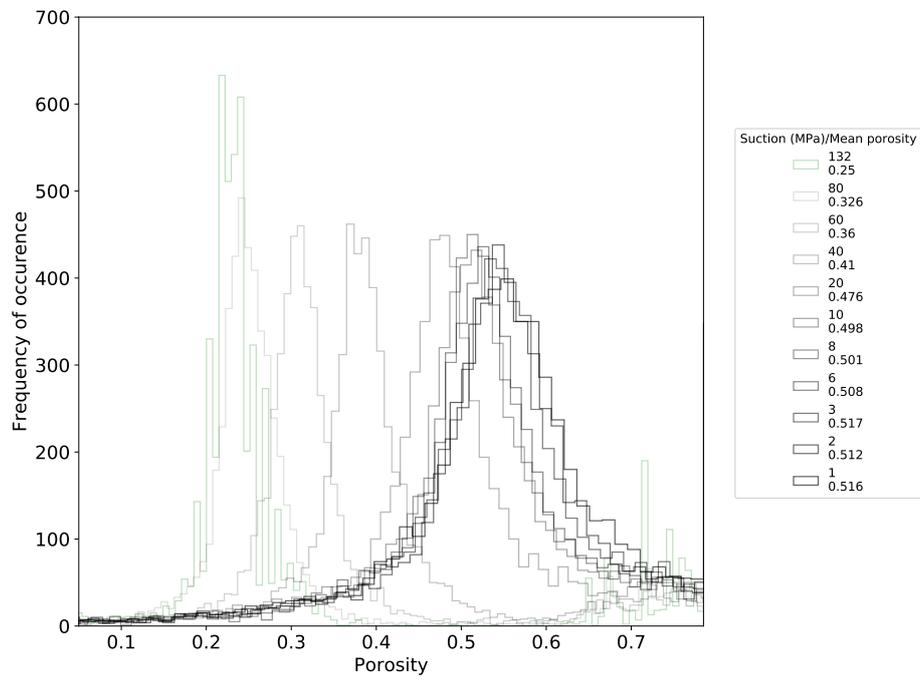


Figure 3.16: Evolution of porosity distribution in pellet.

3.5.8 Convergence study

The heterogeneity associated with the coupled Discrete Element and Partially saturated Finite Volume method presented in Sec. 3.3 introduces an important question of how resolved the coupled domain needs to be in order to capture microscopic heterogeneous behaviors. The present section approaches the problem using a convergence study, which tracks characteristic output parameters based on varying numbers of DEM points (and thus intrinsic finite element triangulation). Three characteristic output quantities were monitored including water permeability, mean water saturation, and volumetric deformation (ϵ_v) during hydration to 130 MPa suction. As shown in Fig. 3.17, the volumetric strain clearly converges as the number of DEM points reaches 150k. However, the average water saturation and average water conductivity do not fully converge. Although they appear to begin converging, we were unable to reach full convergence using our current hardware. We accept this limitation and account for it when drawing conclusions. Future analyses should work on alleviating this limitation.

3.5.9 Spatial scale effects

The spatial scale of various processes plays an important role in understanding the role of heterogeneity during hydration of compacted clay powder. The present section describes a statistical investigation⁵ of heterogeneity by hydrating various sized sub-regions extracted from a principal heterogeneous domain (Fig. 3.8). Practically, a separate simulation is initiated for each sub-region by building a new domain and assigning the heterogeneity accordingly. Boundary conditions and water loadings followed those highlighted in Sec. 3.5.1. These sub-region volumes ranged from 1/100th to 1/4th of the principal domain volume. A set of 40 realizations were run for each sub-region size, with each spatial coordinate randomly generated from the principal heterogeneity domain. After all 40 realizations were complete, the mean and standard deviation were computed for various characteristic model outputs such as water permeability, gas permeability, crack volume, volumetric deformation, and mean water saturation. As shown in Fig 3.18, the mean water permeability stays roughly constant for all investigated scales and for each level of saturation, but the variation of permeability for sub-regions increases dramatically at small domain fractions <1/100th of the principle domain size. A similar trend is observed for the gas permeability, suggesting that the dominant "crack scale" is at 1/100th of the principle domain. The crack volume plot in Fig 3.18 confirms the suspicion, showing that a greater fraction of domain sub-region is consumed by crack volumes at 1/100th of the principle volume. Further, the variation of crack volume per sub-region volume is greatest at a domain fraction of 1/10th and at low saturations, which indicates that the crack network is not evenly distributed at the macroscopic scale during the initial phase of hydration.

Meanwhile, the mean volumetric deformation is greatest at a sub-region approximately equivalent to 1/10th of the principle domain, with the greatest variation of volumetric deformation occurring at 1/16th of the principle domain. Suggesting that the scale between 1/30th and 1/15th

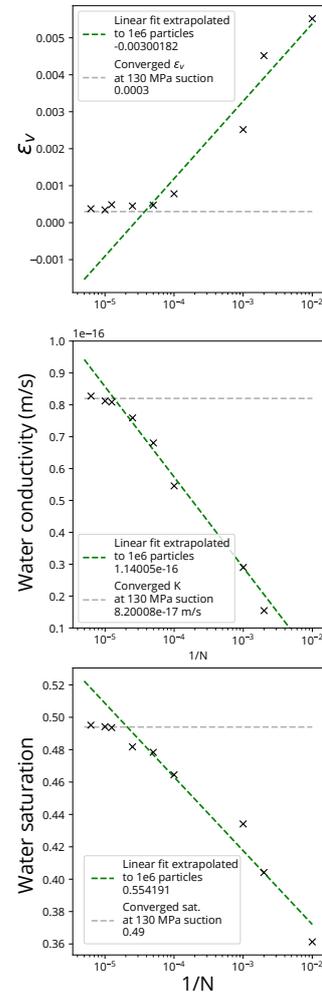


Figure 3.17: Convergence of characteristic output quantities for varying Discrete Element point cloud densities (N =number of DEM points) and associated finite element triangulation resolutions.

5: All (or most of) the computations presented in this section were performed using the UMS GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), which is supported by CNRS and Grenoble research institutes.

of the principle domain is deforming independently to the full domain. Saturation on the other hand remains constant across all scales (Fig. 3.18), despite a strong change of variation with saturation level, in particular at 1/6th of the principle domain. This suggests that saturation heterogeneity is increasing with increasing saturation, in particular at larger scales.

3.5.10 Practical reproduction of results

Readers interested in reproducing the results or simply using the model presented here can access it as part of Yade DEM opensource software [88]. The entire code can be freely installed by typing:

[88]: Yade (2021), *Yade source code*

```
sudo apt-get install yade
```

into any Ubuntu linux terminal. Installation instructions for other linux variants can be found at [89]. Once the code is installed, the user can run the input script provided as supplementary material at <http://u.pc.cd/slYrtalk> or found in Appendix .2 (Chapter4_example_script.py) by executing the following command in a terminal:

```
yade Chapter4_example_script.py
```

Any reader who seeks additional assistance in running or modifying the code provided here, should seek assistance from the author at answers.launchpad.net/yade.

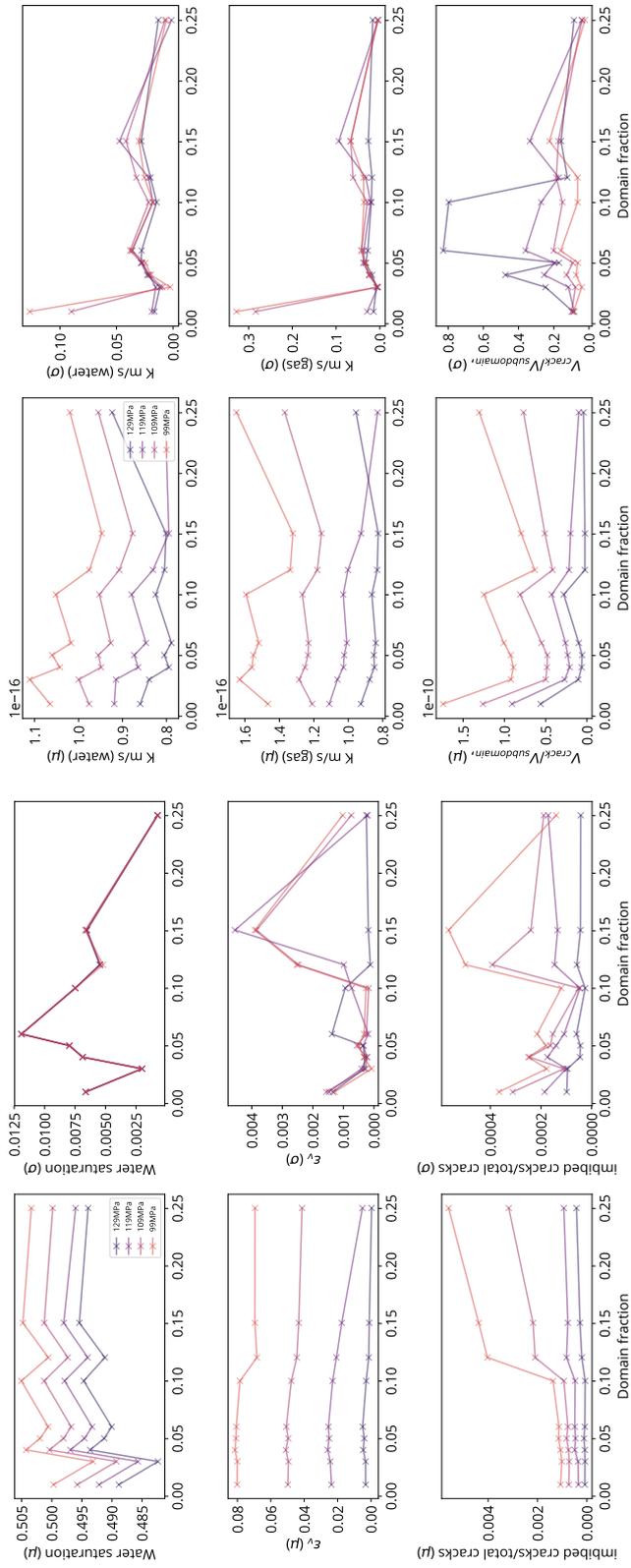


Figure 3.18: Statistical investigation of characteristic model outputs for various domain scales. μ and σ indicate the statistical mean and standard deviation, respectively.

3.6 Conclusion

The novel numerical method presented within enables the quantification of water and gas permeability changes during partial saturation of compacted clay due to the presence of heterogeneities and cracks. By coupling the finite-volume and discrete element methods, the framework accounts for mechanical and hydraulic processes, including heterogeneous saturation and swelling changes. The addition of a discrete crack model enables the complex evolution of gas and water permeability during partially saturated conditions. Results demonstrate how the image based porosity assignment and the crack model both contribute to a non-intuitive evolution of permeability during hydration. In particular, near 60% saturation, the water permeability plateaus and decreases while the gas permeability continues to increase quickly. In confirmation with experimental literature results, beyond 90% saturation, the water permeability increases by several orders of magnitude while the gas permeability decreases by several orders of magnitude. The fully coupled model is also used to investigate mesh resolution, heterogeneity scale effects, crack fabric evolution, and the effect of confining pressure on permeability. The present study draws the following conclusions:

- ▶ Discrete modeling of heterogeneities and cracks is necessary to understand the evolution of permeability in partially saturated compacted swelling clay materials
- ▶ The non-intuitive and anisotropic evolution of water and gas permeabilities demonstrates the complex interactions between swelling, porosity, and water retention in heterogeneous clay materials.
- ▶ Crack networks develop due to the initial presence of porosity heterogeneity.
- ▶ Confining pressure reduces gas permeability by over one order of magnitude in partially saturated conditions compared to unconfined conditions.
- ▶ The DEM domain resolution needs to be 150k points to resolve heterogeneities extracted from Xray CT scan data at 1 mm scale.
- ▶ Permeability variation and crack effects in the numerical model are highest at the smallest scale, 1/100th the size of the compacted clay pellet.
- ▶ The crack fabric changes most during the early phase of hydration before 60% saturation.

These conclusions draw important links to field scale engineered barrier behaviors. For example, the anisotropic evolution of permeability should be considered when estimating the hydraulic or gas permeability of an engineered barrier at various suction values. In particular, the direction orthogonal to clay compaction may reduce gas permeability and increase hydraulic permeability more than the other two directions. Additionally, the hydraulic permeability may vary by an order of magnitude depending on which axis is evaluated. Another important field consideration includes the effect of confining pressure. The present study demonstrated how higher confining pressure has the effect of reducing gas permeability but also decreasing hydraulic permeability (relative to unconfined conditions). This effect is due to the gas entry pressure into cracks and the gas entry pressure increasing with smaller crack apertures as experienced in confined conditions.

The results presented in this paper will also play a significant role in up-scaling clay saturation models from the millimeter scale (presented here) to the meter scale and up to more efficient continuum-like models. An example is presented in Chap. 4, shows an up-scaling of the millimeter scale permeability curve produced in this study for a coarse discretization at the centimeter or meter scales in pellet-powder mixtures.

A framework for modeling hydro-mechanical processes in clay pellet-powder mixtures

4

4.1 Introduction

This chapter presents a conceptual model designed to elucidate the hydro-mechanical processes in a lab-scale hydration test of an engineered barrier comprised of bentonite pellets and powder (Fig. 4.1). Beyond the conceptual model, an open-sourced practical implementation is also discussed and validated using experimental data collected by [74]. The chapter concludes by discussing the strengths and weaknesses of the theoretical model and practical implementation. Finally, a road-map is presented to demonstrate how future advancements will improve the model/implementation.

4.2 Methods

4.2.1 Mechanical Model

Following the standard DEM formulation outlined in Sec. 5.2.1, the contact law (Eq. 2.3) for the present model follows a Hertz-Mindlin law [75, 76]. Hertz law enables a non-linear and saturation-dependent stiffness at the contact despite the fact that a true contact between two compacted clay pellets is likely more complex as outlined by Sec. 2.2. Hence, the contact normal (\mathbf{f}_n) and shear (\mathbf{f}_s) forces are both non-linear and computed using the particle's interaction radius:

$$\mathbf{f}_n = \frac{4}{3} E^* R^{1/2} d_n^{3/2} \mathbf{n}_n \quad (4.1)$$

where d_n is the normal displacement between the two points, and \mathbf{n}_n is the unit vector parallel to the interaction. R is the equivalent interaction radius of particle a and particle b , $r_a r_b / (r_a + r_b)$, as shown in Fig. 2.3. Meanwhile, the Young's modulus is the harmonic mean of the two interacting particles, which accommodates an equal weighting in the case of two drastically different stiffnesses:

$$E^* = \frac{E_a(s_i) E_b(s_i)}{E_b(s_i)(1 - \nu_a^2) + E_a(s_i)(1 - \nu_b^2)} \quad (4.2)$$

and following experimental data collected by [16], the Young's modulus varies with suction (s_i):

$$E(s_i) = 3(1 - 2\nu_i) \frac{1}{\beta} \exp(\alpha s_i) \quad (4.3)$$

where α [Pa] and β [Pa] are the same model parameters used for the volumetric swelling model presented later in the chapter Eq. 4.27. Finally, as demonstrated by [77], the yield strength of the pellet [Pa] is:

[74]: Molinero-Guerra et al. (2018), 'Analysis of the structural changes of a pellet/powder bentonite mixture upon wetting by X-ray computed microtomography'

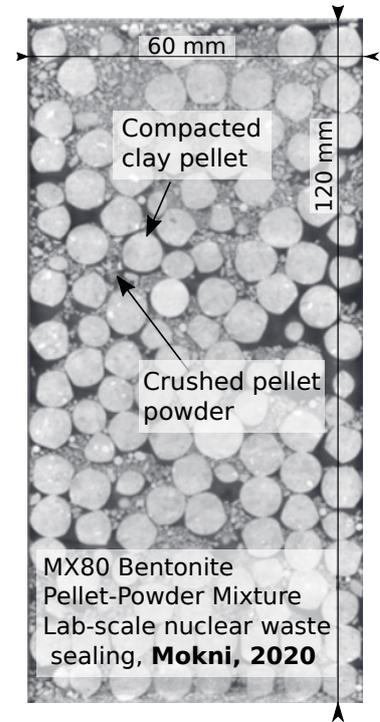


Figure 4.1: CT scan imagery of bentonite clay pellet-powder mixture specimen used for hydration tests [15].

[75]: Hertz (1882), 'Ueber die Berührung fester elastischer Körper (On Contact Between Elastic Bodies)'

[76]: Mindlin (1949), 'Compliance of Elastic Bodies in Contact'

[16]: Darde et al. (2020), 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'

[77]: Darde et al. (2018), 'Hydro-mechanical behaviour of high-density bentonite pellet on partial hydration'

$$R_i = 3(1 - 2\nu)B \frac{1}{\beta} \exp(\alpha s_i) \quad (4.4)$$

where B [-] is a constant fit to experimental data by [77]. In the DEM formulation, the value of R_i is the normal force divided by the contact area. Shear force considerations follow a similar Hertzian approach and depend on the orientations of both particles in a local coordinate system:

$$\Delta \mathbf{f}_s = 4\sqrt{R} \frac{G}{2 - \nu} \Delta \sqrt{d_s} \cdot \mathbf{n}_s \quad (4.5)$$

$$\mathbf{f}_s = \mathbf{f}_{s, \text{prev}} + \Delta \mathbf{f}_s \quad (4.6)$$

$$(4.7)$$

where G is the average shear modulus of both particles (e.g. $G_a = E_a / (2(1 + \nu_a))$), ν is the average Poisson ratio between the two interacting DEM particles. \mathbf{n}_s is the unit vector orthogonal to the DEM point interaction. Δd_s is the tangential increment of displacement, which depends on the relative angular velocities, $\dot{\mathbf{x}}_{rel}$, of the two points:

$$\dot{\mathbf{x}}_{rel} = (\dot{\mathbf{x}}_a + \omega_a \times (\mathbf{x}_{ab} - \mathbf{x}_a)) - (\dot{\mathbf{x}}_b + \omega_b \times (\mathbf{x}_{ab} - \mathbf{x}_b)) \quad (4.8)$$

$$\Delta d_s = \dot{\mathbf{x}}_{rel} - (\mathbf{n}_n \cdot \dot{\mathbf{x}}_{rel}) \mathbf{n}_n \quad (4.9)$$

where \mathbf{x}_{ab} is the midpoint between interacting points a and b . ω is the angular velocity.

4.2.2 Mass Transport Model

The present Eulerian mass transport scheme is governed by the conservation of mass:

$$\frac{dm}{dt} = \frac{d}{dt} \int_{\Theta} \rho dV + q \quad (4.10)$$

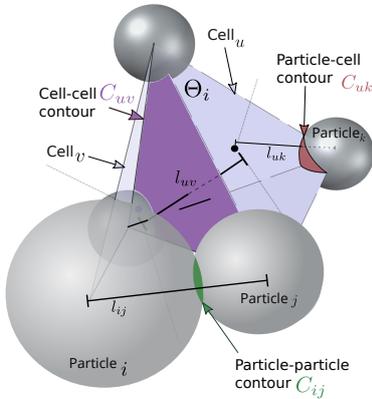


Figure 4.2: Domain discretization and geometric quantities of the mass transport equations for a clay pellet powder mix.

where m is the total mass, ρ is the mass per unit volume, Θ , and q is a material mass flux source or sink. In the present implementation, the global domain is comprised of a collection of spherical compacted clay pellets with clay powder filling the void space (Fig. 4.1). Via the divergence theorem, Eq. 4.10 can be cast to its surface integral, where the contour, $\partial\Theta_i$, is defined by the interface between the pellet and powder domains:

$$\frac{dm_i}{dt} = - \sum_{j=1}^m \int_{\partial\Theta_i} \rho(\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS \quad (4.11)$$

where \mathbf{v} is the contour velocity, and \mathbf{n} is the outward pointing unit vector. Finally, reducing $\partial\Theta_i$ to only the shared contours of all incident local

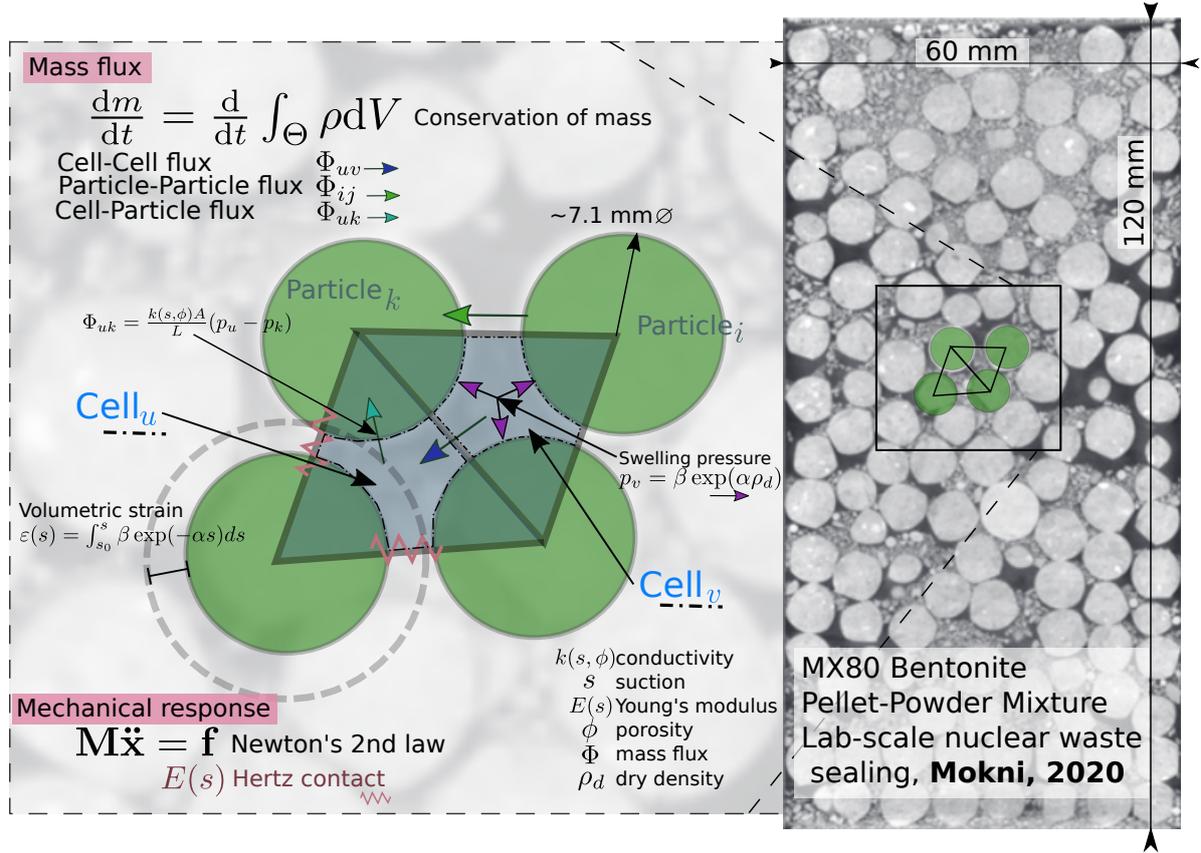


Figure 4.3: Visualization of key quantities for the coupled DEM mass transport model.

domains, C_{ij} , (Fig. 4.2) and assuming small Mach numbers, Eq. 4.11 becomes:

$$\frac{dm_i}{dt} = - \sum_{j=1}^l \int_{C_{ij}} \rho(\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS \quad (4.12)$$

where l is the total number of incident systems. In the present numerical discretization of the problem, the geometry is discretized into a collection of tetrahedra where each vertex represents a single clay pellet which is mechanically represented by a single discrete element (Fig. 4.3). The void space between the discrete elements are considered partially filled by clay powder. These void spaces and discrete elements all behave according to Eq. 4.12. The discretized form of Eq. 4.12 for one vertex (i.e. one discrete element), k , the surface integral is reduced to a summation along all incident vertices and tetrahedra (i.e. void spaces):

$$\frac{dm_k}{dt} = \sum_{w=1}^N \int_{C_{wk}} \Phi_{wk} \cdot \mathbf{n}_{wk} dS + \sum_{u=1}^M \int_{C_{uk}} \Phi_{uk} \cdot \mathbf{n}_{uk} dS \quad (4.13)$$

where C and Φ are the interface contour and average fluid flux between two systems, respectively. M are the incident tetrahedra and N are the incident vertices¹ In the present implementation, the contour between two vertices, C_{ij} is equivalent to the overlapping plane created by the two vertex radii Fig. 4.2. Meanwhile, the shared contact interface between a

1: A cell has a known number of incident systems, eight, since it is comprised of four vertices and four facets. Vertices have an unknown number of incident vertices and cells since this depends on interactions and triangulation geometry.

vertex and a tetrahedron is the spherical triangle, C_{uk} depicted in Fig. 4.2. Finally, the mass flux for a tetrahedron system, u , follows the same logic except replacing the vertex-vertex integral with a tetrahedron-tetrahedron integral:

$$\frac{dm_u}{dt} = \sum_{v=1}^L \int_{C_{uv}} \Phi_{uv} \cdot \mathbf{n}_{uv} dS + \sum_{u=1}^M \int_{C_{uk}} \Phi_{uk} \cdot \mathbf{n}_{uk} dS \quad (4.14)$$

where L are the incident tetrahedra and C_{uv} is the area of the shared facet as depicted in Fig 4.2. In the present implementation, all contour integrals are decomposed into the sum of fluxes between the system, i , and all incident systems, j :

$$\sum_{j=1}^m \int_S \Phi \cdot \mathbf{n} dS = \sum_{j=1}^m \rho \frac{k_{ij} A_{ij}}{\mu} L_{ij} (p_j - p_i) \quad (4.15)$$

where k_{ij} is the permeability between incident systems i and j , where each permeability is based on mean saturation and mean porosity for powder filled void space or empirical data for the discrete elements². For the case of porosity and saturation, the value is defined as:

$$k_{ij} = k_{M,ij} k_{rM,ij} \quad (4.16)$$

where the residual relative permeability factor, k_{rM} , is empirically related to the degree of saturation [32]:

$$k_{rM} = \left(\frac{s_p - s_r}{s_s - s_r} \right)^n \quad (4.17)$$

$$(4.18)$$

where s_p is the mean phase saturation (fluid or gas), s_r is the residual saturation, s_s is the saturated saturation, and n is a material parameter. Meanwhile, k_M , intrinsic permeability, is an exponential function of the mean porosity between incident tetrahedrons i and j at time t , ϕ_{ij}^t , [11]:

$$k_M = k_o \exp(b(\phi_{ij}^t - \phi_{o,ij})) \quad (4.19)$$

where k_o is the reference permeability for porosity ϕ_o and b is a calibrated factor.

Wrapping the various definitions of permeability and discretization geometry into one flux term between arbitrary systems, q_{ij} :

$$q_{ij} = \frac{k_{ij} A_{ij}}{L_{ij}} (p_j - p_i) \quad (4.20)$$

2: Chap. 3 presents a micro-scale model designed to analyze crack network developments and their effect on permeability. Thus, we input these micro-scale permeability predictions directly into k_{ij} for the pellets. Additional details provided in Sec. 4.6

and generalizing the diffusivity term:

$$g_{ij} = \frac{k_{ij}A_{ij}}{L_{ij}} \quad (4.21)$$

the generalized mass transport framework follows:

$$\frac{dm_i}{dt} = \sum_{j=1}^m g_{ij}(p_j - p_i)\rho \quad (4.22)$$

with m representing all incident systems. In the proposed method, the derivative of mass in time is discretized using a forward difference Euler scheme:

$$m^{t+\Delta t} = \Delta t \left(\sum_{j=1}^m \left(\frac{k_{ij}A_{ij}}{L_{ij}} (p_j^t - p_i^t) \rho \right) \right) + m^t \quad (4.23)$$

Once the new mass of water is computed, the new pressures can be *estimated*³ using the water retention curve [78]:

$$P(s) = -\zeta s^{\frac{-1}{\lambda}} (s^{\frac{-1}{\lambda}} - 1)^{-\lambda} (s^{\frac{1}{\lambda}} - 1) \quad (4.24)$$

with s being the saturation and ζ and λ are curve parameters depending on porosity, ϕ [32]:

$$\zeta(\phi) = \zeta_o \exp(a(\phi_0 - \phi)) \quad (4.25)$$

$$\lambda(\phi) = \lambda_o \exp(b(\phi_0 - \phi)) \quad (4.26)$$

where ζ_o and λ_o are constant initial values. The simultaneous solution of p and V is non-linear and thus, requires special treatment as outlined in Sec. 4.3.

4.2.3 Volumetric Swelling Model

The swelling of the pellets follow exponential models fit to experimental data by [14, 36]. As shown by Eq. 4.27, ε is estimated using an exponential variant:

$$\varepsilon(s) = \int_{s_0}^s \beta \exp(-\alpha s) ds \quad (4.27)$$

$$\varepsilon(s) = \frac{\beta}{\alpha} [\exp(-\alpha s_0) - \exp(-\alpha s)] \quad (4.28)$$

where β , α are model parameters, s_0 is initial suction, and s is the current suction. Given the strain, ε , the particle (clay pellet) radii are adjusted as follows:

3: The estimate serves as a starting point for solving the non-linear pressure-volume problem as described in detail in Sec. 4.3

[78]: Van Genuchten (1980), 'A closed form equation for predicting the hydraulic conductivity of unsaturated soils.'

[32]: Mokni (2016), 'Analysis of hydro-mechanical behaviour of compacted bentonite/sand mixture using a double structure formulation'

[14]: Molinero-Guerra (2018), 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'

[36]: Darde et al. (2020), 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'

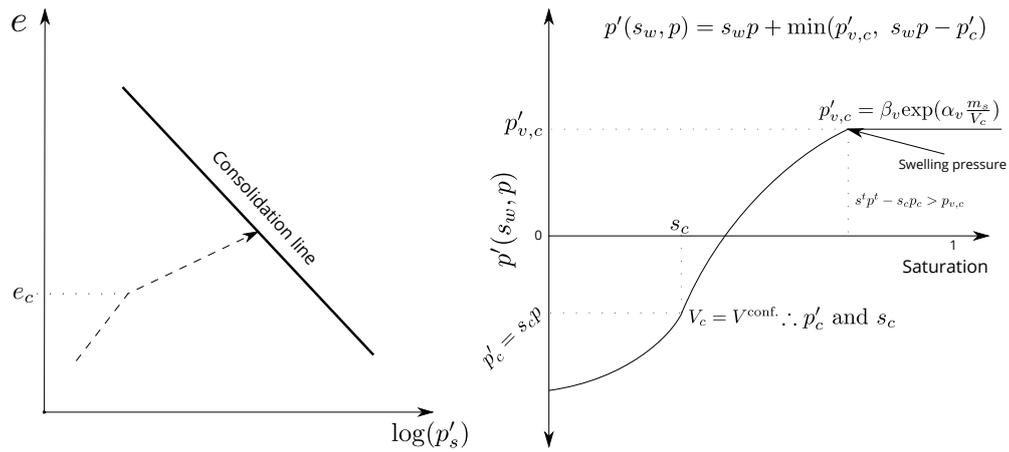


Figure 4.4: Loading path for partially saturated clay transitioning from unconfined to confined conditions (left) and the corresponding effective pressure (right).

$$\Delta v = \varepsilon(s)v \quad (4.29)$$

$$\Delta r = \Delta v \left(\frac{3v}{4\pi} \right)^{1/3} \quad (4.30)$$

Where v is the interaction volume of each DEM point.

Meanwhile, the swelling of powder within the void spaces follows Eq. 4.27, however the effective stress (Fig. 4.4, right), p' , exerted on the void contour depends on the consolidation line (Fig. 4.4, left) for an MX80 bentonite powder [79] the final dry density at full saturation relates to the effective stress as:

$$p'_{v,c} = \beta_v \exp \left(\alpha_v \frac{m_s}{V_c} \right) \quad (4.31)$$

where β_v and α_v are the parameters used to fit the exponential function to experimental data swelling data collected by [79]. These parameters are fit for MX80 bentonite to be $\beta_v = 1.78e - 4$ and $\alpha_v = 6.75$. The dry density of the bentonite, ρ_d is computed as:

$$\rho_d = \frac{m_s}{V} \quad (4.32)$$

In the present model, the clay powder is partially saturated, which implies an onset of swelling stress before reaching the consolidation line as shown in Fig. 4.4. The present model incorporates pressure exerted by the partially saturated powder on the cell contour:

$$p'(s_w, p) = s_w p + \min(p'_{v,c}, s_w p - p'_c) \quad (4.33)$$

where s_w is the water saturation and p'_c is the effective stress on the contour when the powder volume is confined i.e. when the powder volume is equivalent to the cell volume.

[79]: Wang et al. (2012), 'Experimental study on the swelling behaviour of bentonite/claystone mixture'

4.2.4 Hydro-mechanical Model

The hydro-mechanical coupling follows the original implementation by [57]; pressure and viscous forces acting on some particle, k , are a contour integrals of the absolute pressure and viscous stress along the contour shared between solid and incident “pore-throat”. A visualization of the “pore-throat” and the surfaces used for integration is shown in Fig. 4.5. Discretizing the contour to the particular geometry shown in Fig. 4.5, and projecting the pressure onto the intersection of the shared facet with the solid with the shared facet, the pressure force is simply:

$$\mathbf{f}_{ij}^{p,k} = A_{ij}^k (p_j - p_i) \mathbf{n}_{ij} \quad (4.34)$$

where \mathbf{n}_{ij} is the unit vector, A_{ij}^k is the intersection of facet area and discrete element point k (Fig. 4.5). However, different from [57], the present implementation considers the effect of partial saturation (s_w) of each cell and partial volume fraction of powder ($V_i/V_{t,i}$), so the force is factored to be:

$$\mathbf{f}_{ij}^{p,k} = A_{ij}^k \frac{V}{V_t} (s_{w,j} p_j - s_{w,i} p_i) \mathbf{n}_{ij} \quad (4.35)$$

where p is the swelling pressure computed with Eq. 4.33. Thus, the swelling powder within the voids adds force to the discrete element particles, which contributes to the force balance on each of the DEM particles.

The viscous shear force is integrated similar to the pressure force above. As [57] points out, the total viscous force is a contour integral of the pressure along the fluid area of the shared facet, discretized to A_{ij}^f in Fig. 4.5. For each facet, the total viscous force, \mathbf{f}_{ij}^V , and the individual viscous force applied to each of the participating particles, $\mathbf{f}_{ij}^{v,k}$, is computed:

$$\mathbf{f}_{ij}^V = A_{ij}^f \frac{V}{V_t} (s_{w,j} p_j - s_{w,i} p_i) \mathbf{n}_{ij} \quad (4.36)$$

$$\mathbf{f}_{ij}^{v,k} = \mathbf{f}_{ij}^V \frac{A_{ij}^k}{\sum_{x=k}^m A_{ij}^x} \quad (4.37)$$

The total force added to the the discrete element contact force summation (Eq. 3.1) becomes:

$$\mathbf{f}^k = \sum_{(ij)incident} \mathbf{f}_{ij}^{v,k} + \mathbf{f}^{p,k} \quad (4.38)$$

4.3 Solving the non-linear problem

In the model presented above, Sec. 4.2, there is a non-linear relationship between fluid pressures and volumetric deformations. In plain words, the fluid pressure of each local system depends on the local water

[57]: Chareyre et al. (2012), ‘Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings’

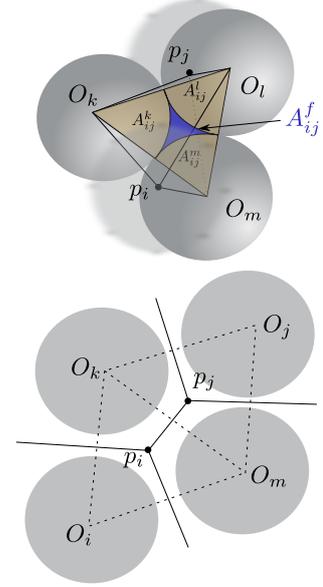


Figure 4.5: Geometrical areas used for pressure and viscous force integration.

retention curve (Eq. 4.24). Meanwhile, the local water retention curve also depends on the local volume, which is dependent on the pressure as shown in Eq. 4.27. Since both the volumetric deformation and the water-retention curves are non-linear, the arises a need to iteratively approach the solution for each local system. In the present framework, the solution is approached using a multi-variate Newton-Raphson with inequality constraints. The two functions, $f_1(V, p)$ and $f_2(V, p)$, containing the two unknowns are first set to zero:

$$f_1(V, p) = p - \left(-\zeta(V)s(V)^{\frac{-1}{\lambda(V)}} \left(s(V)^{\frac{-1}{\lambda(V)}} - 1 \right)^{-\lambda(V)} \left(s(V)^{\frac{1}{\lambda(V)}} - 1 \right) \right) \quad (4.39)$$

$$- \mu \sum_{i=1}^3 \log c_i(V, p) = 0$$

$$f_2(V, p) = V - V_o \left(\frac{\beta}{\alpha} [\exp(-\alpha p_o) - \exp(-\alpha p)] + 1 \right) - \mu \sum_{i=1}^3 \log c_i(V, p) = 0 \quad (4.40)$$

with $s(V)$, $\zeta(V)$ and $\lambda(V)$ short for:

$$s(V) = \frac{m_w}{(V - V_s)\rho} \quad (4.41)$$

$$\zeta(V) = \zeta_o \exp \left(a \left(\phi_0 + \frac{V_s}{V} - 1 \right) \right) \quad (4.42)$$

$$\lambda(V) = \lambda_o \exp \left(b \left(\phi_0 + \frac{V_s}{V} - 1 \right) \right) \quad (4.43)$$

where the volume of solids, V_s , is constant throughout the simulation and so are the initial porosity, ϕ_o , and initial pressure, p_o . However, m_w is only constant during each time-step, which satisfies the continuity equation (Eq. 4.10). Meanwhile, the last term, $\mu \sum_i^3 \log c_i(V)$, enforces the inequality constraints defined to the left.

Inequality constraints:

$$\phi \geq 0, \quad c_1(V) = 1 - \frac{V_s}{V} \geq 0$$

$$\phi \leq 1, \quad c_2(V) = \frac{V_s}{V} \geq 0$$

$$V \leq V_{cell}, \quad c_3(V) = V_{cell} - V \geq 0$$

$$s \geq 0, \quad c_4(V) = 1 - \frac{m_w}{(V - V_s)\rho} \geq 0$$

$$s \leq 1, \quad c_5(V) = \frac{m_w}{(V - V_s)\rho} \geq 0$$

$$p \leq 0Pa, \quad c_6(p) = -p \geq 0$$

$$p \geq -132MPa, \quad c_7(p) = p + 132MPa \geq 0.$$

These constraints enforce phase-relationships and add the applicable pressure boundaries of the water-retention curve. Finally, the factor μ is a relaxation which exponentially decays until a satisfactory error threshold is met. The solution to Newton-Raphson follows:

$$\mathbf{J}^k \delta \mathbf{x}^k = -\mathbf{r}^k \quad (4.44)$$

with k representing the iteration of the Newton-Raphson procedure, \mathbf{J} being the Jacobian matrix, $\delta \mathbf{x}$ is the the vector of updates to solve for, and \mathbf{r} is the vector of residuals. The Jacobian is constructed:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial p} & \frac{\partial f_1}{\partial V} \\ \frac{\partial f_2}{\partial p} & \frac{\partial f_2}{\partial V} \end{bmatrix} \quad (4.45)$$

Considering the difficulties of computing the derivatives for Eq. 4.39, a numerical derivative is used such that:

$$\frac{\partial f_j}{\partial x_i} = \frac{f_j(V, p) - f_j(V, p)}{2h_i} + O(h^2) \quad (4.46)$$

where f_j represents one of the two functions: Eq. 4.39 or 4.40. x_i represents the three variables of interest, V , and p .

Finally, the iterative procedure of Newton-Raphson starts with an initial guess of p based on $m^{t+\Delta t}$ from the present timestep and V^t from the previous time step, as well as an initial guess of V based on p :

$$p = -\zeta(V^t) s(V^t)^{\frac{-1}{\lambda(V^t)}} \left(s(V^t)^{\frac{-1}{\lambda(V^t)}} - 1 \right)^{-\lambda(V^t)} \left(s(V^t)^{\frac{1}{\lambda(V^t)}} - 1 \right) \quad (4.47)$$

$$V = V_o \left(\frac{\beta}{\alpha} [\exp(-\alpha p_o) - \exp(-\alpha p)] + 1 \right) \quad (4.48)$$

with this initial guess of $x = (V, p)$, we compute \mathbf{r} and \mathbf{J} to solve for δx :

$$\delta x = \mathbf{J}^{-1}(-\mathbf{r}) \quad (4.49)$$

with δx , we update x :

$$x = x + \alpha_r \delta x \quad (4.50)$$

where α_r is a relaxation factor that exponentially decays until the root mean square error (RMS) reaches a prescribed tolerance:

$$err^k = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{x_i^k - x_i^{k-1}}{x_i^{k-1}} \right)^2} \quad (4.51)$$

4.4 Numerical stability

The stability of the fluid scheme relies on avoiding numerical oscillations, which is practically ensured by requiring that the pressure change in subsequent steps does not exceed the pressure change of the previous step assuming no additional change to boundary conditions:

$$\left\| \frac{\Delta p^{t+\Delta t}}{\Delta p^t} \right\| < 1 \quad (4.52)$$

with

$$\Delta p = \alpha(p_o - p) \frac{\partial p}{\partial s} \quad (4.53)$$

where α is the diffusivity coefficient, in our case $\alpha_{ij} = \sum_{j=1}^N \frac{k_{ij} A_{ij} \rho}{V l_{ij}}$. It follows that:

$$\left\| \frac{\alpha(p_o - p - \Delta p) \frac{\partial p}{\partial s}}{\alpha(p_o - p) \frac{\partial p}{\partial s}} \right\| < 1 \quad (4.54)$$

$$\left\| 1 + \frac{\Delta t \alpha (p - p_o) \frac{\partial p}{\partial s}}{p_o - p} \right\| < 1 \quad (4.55)$$

$$\left\| 1 - \Delta t \alpha \frac{\partial p}{\partial s} \right\| < 1 \quad (4.56)$$

$$\Delta t < \frac{2}{\alpha} \frac{\partial p}{\partial s} \quad (4.57)$$

Thus, the system within the domain characterized by the maximum diffusivity, α_{max} , controls the maximum allowable timestep. However, this naive approach assumes the change of pressure of a single system in a homogeneous domain, which is not practically the case. Further, it assumes a constant volume associated with $\partial p / \partial s$, which is not truly consistent. This means we factor the estimate by χ :

$$\Delta t < \chi \frac{2}{\alpha_{max}} \frac{\partial p}{\partial s} \quad (4.58)$$

with χ set through trial and error to 0.6.

Additionally, it is important to note that the diffusivity can be scaled to manipulate the fluid fluxes when $\frac{\partial p}{\partial s}$ is close to 0. In this situation the, the system is almost perfectly in equilibrium but the required time-step is extremely low, so the diffusivity is factored to move more quickly through the simulation.

4.5 Defining structural heterogeneity

The model outlined in Sec. 4.2.2 enables a wide variety of choice for structural heterogeneity assignment. First, the packing structure of the compacted clay pellets (i.e. the DEM particles) can be controlled using varying particle size distributions, or varying packing methods such as hexagonal packings, cubic packings, or purely random packings. Next, the initial powder volume fraction field can be assigned randomly or according to a predefined statistical distribution. Another option may be to assign the powder volume fraction field directly from imagery, as is outlined in details in Sec. 3.3.4. Finally, perhaps the most effective choice for large statistical analyses, the powder volume fraction field can be assigned according to a random field, which should account for mesoscale structure. The experimental validation presented in Sec. 4.8, assigned the volume powder fractions randomly according to a predefined Weibull distribution, as shown in Fig. 4.6 and Fig. 4.10. However, we have plans to implement a random field next.

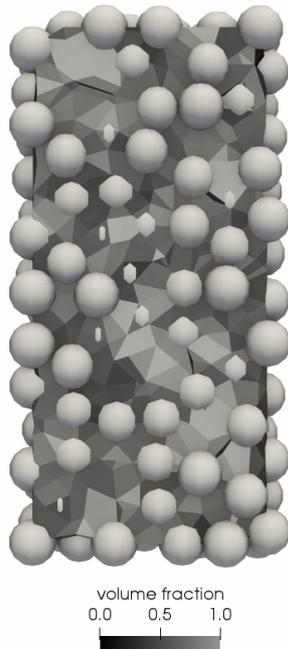


Figure 4.6: Demonstration of initial structural heterogeneity assignment. Weibull distributed initial powder volume fractions (triangulation). Random packing for monodisperse pellet sizes (sphere locations).

4.6 Up-scaling permeability data from micro-scale models

As presented in Chap. 3, the micro-scale hosts a range of discontinuous processes based on heterogeneous porosity distributions and water surface tensions. The present study informs the permeability of the pellets, k , using the permeability curve obtained from the micro-scale model (Fig. 4.7).⁴ The use of this data escapes the need for using expensive computational effort at the micro-scale, yet maintains the effect of crack network developments during hydration.

4.7 Practical implementation

The model presented in Sec. 4.2 is implemented in Yade open-source software. The implementation is geared towards enabling future research and development, while maintaining speed and stability. This is achieved by wrapping the core C++ functionality in high-level Python. Such a wrapping has already proven itself useful in other Yade packages [80], where users can rapidly prototype, debug, and modify core code. Further, the Python wrapper allows users to include and interact with breadth of Python libraries for coupled live Yade analyses, such as SciPy, Pandas, and many other scientific projects. As an example, the Python script associated with reproducing Sec. 4.8 is available in the supplementary material at <http://u.pc.cd/sLYrtalk> or attached in Appendix .1. Additionally, the source code containing the entire practical implementation is available at <https://gitlab.com/robcaulk/transportengine>⁵ A visual representation of this practical implementation, the flow of execution, and the variety of methods used to discretize the model, are highlighted in the following algorithm overview Fig. 4.8.

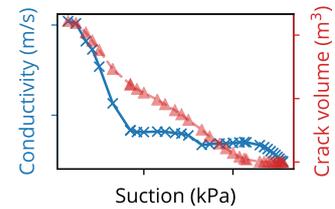


Figure 4.7: Permeability curve, Chap. 3, at the micro-scale for unconfined conditions. This permeability is used to inform the pellet permeability, k , in the present model.

4: An important point to consider is that the microscale data is collected for a free swelling pellet - but here the pellets may be confined by powder and other neighboring pellets. This is a clear weakness in the model despite the geometrical constraints held for the conductivity area A_{ij} and length L_{ij} .

[80]: Smilauer et al. (2021), *Yade documentation 3rd ed.*

5: It will be merged into the code into the general Yade distribution so that it maintains community support moving forward.

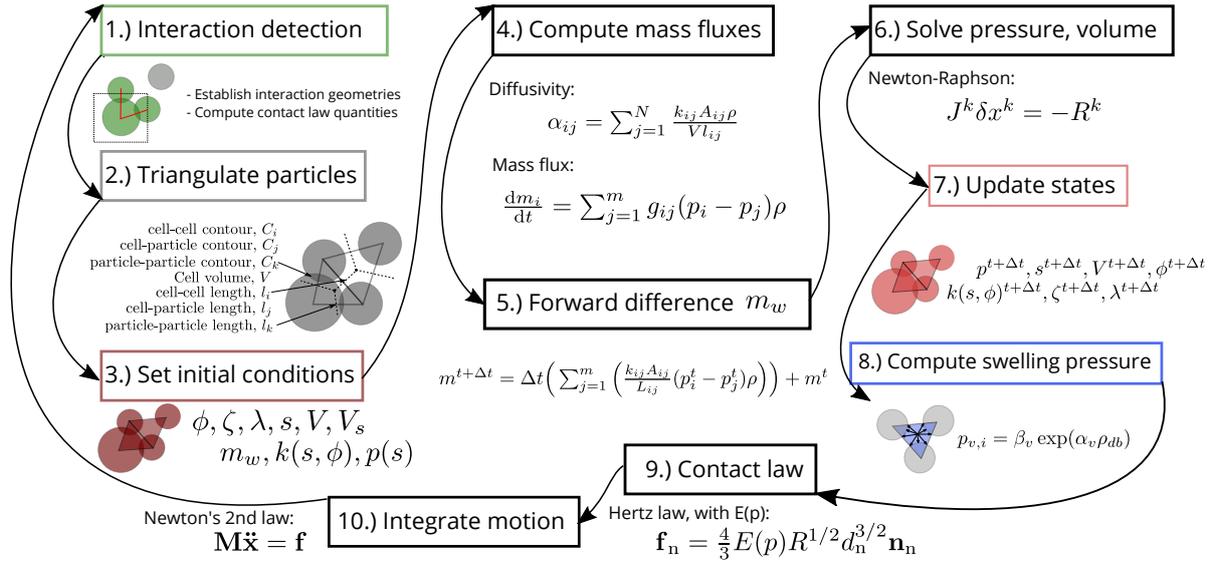


Figure 4.8: Overview of algorithmic implementation.

4.8 Experimental validation and results

The framework presented in Sec. 4.2 and implemented as discussed in Sec. 4.7 is validated in the following section by comparing model results to experimental observations collected by [74]. The hydration experiment hydraulically loads an initially unsaturated mixture of compacted bentonite clay pellets and powder from top and bottom as shown in Fig. 4.9. The numerical representation of the experiment is constructed using Yade open source discrete element software [80]. Model parameters are highlighted in Table 4.1 and Table 4.2.

[80]: Smilauer et al. (2021), *Yade documentation 3rd ed.*

Swelling pressure results shown in Figures 4.10 and 4.11 indicate that the model and associated numerical implementation are capable of capturing experimentally observed trends. For example, the spatial increase of swelling pressure follows the spatial hydration of the specimen, as expected. Starting from the two ends of the specimen, where water is imposed, the Axial and SP20 pressure sensors are first to increase before the interior sensors, SP40 and SP80, follow. As expected, the SP40 and SP80 pressure sensors are equidistant from the hydraulic loading, and therefore follow the same rate of pressure increase. The overall magnitude of the swelling pressures is also similar, which suggests that evolution of pellet stiffness and powder swelling pressure models are properly validated.

The discrepancies between experimental and numerical also shed im-

Table 4.1: Discrete Element Method model parameters for modified Hertz contact.

Parameter	Value	Equation/Reference
β	0.024 MPa ⁻¹	Eq. 4.27, [16]
α	0.016 MPa ⁻¹	Eq. 4.27, [16]
ν	0.3	Eq. 4.3, [16]
B	1.206e-7 m ⁻²	[16]
s_0	132e6 MPa	Eq. 4.27, [16]

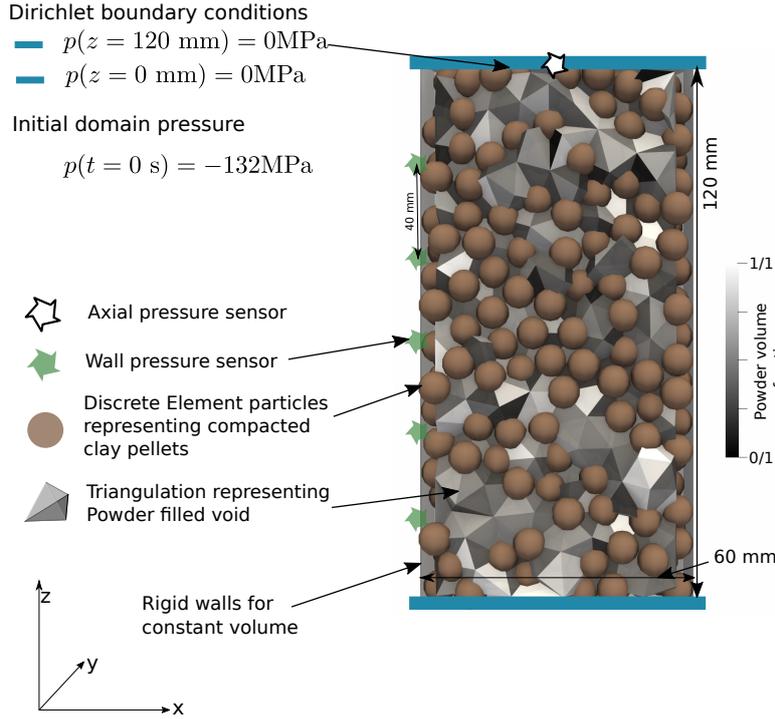


Figure 4.9: Mock-up scale model geometry and boundary conditions

Parameter	Value	Equation/Reference
ζ_o	4.026 MPa	Eq. 4.24
λ_o	0.26	Eq. 4.24
a	6.8	Eq. 4.25 [32]
b	-1.5	Eq. 4.26 [32]
s_w	1	Eq. 4.17,[32]
s_r	0.0	Eq. 4.17,[32]
n	2	Eq. 4.17
b_k	2	Eq. 4.19
k_o	1e-20 m ²	Eq.4.19, [32]
ϕ_o	0.25	Eq.4.25

Table 4.2: Mass transport model parameters.

portant light on the model validation. One obvious discrepancy is the swelling pressure increase during the first 60% of the simulation, which is slower than the swelling pressure increase in the experiment. This is likely caused by the coarse resolution of the powder and pellets in the numerical simulation. In the experiment, there are important kinetics occurring within the powder voids and pellets - but these gradients are not resolved in this model. Another key discrepancy is found in the convergence of the swelling pressures experimentally, but not numerically. This discrepancy may be due to ratcheting in the DEM mechanical model i.e. locking of the force chains beyond the yield. Meanwhile, the experimental material becomes more homogeneous and eventually balances forces along the length of the wall. Another discrepancy is found within the experimental SP60 swelling pressure data, which shows that the interior of the specimen swells before the boundaries. This may derive from complex force chains in the structural heterogeneity which lead to focused forces on the pressure sensor. Finally, the experimental SP40 starts the hydration reading negative swelling pressure - but this is an artifact.

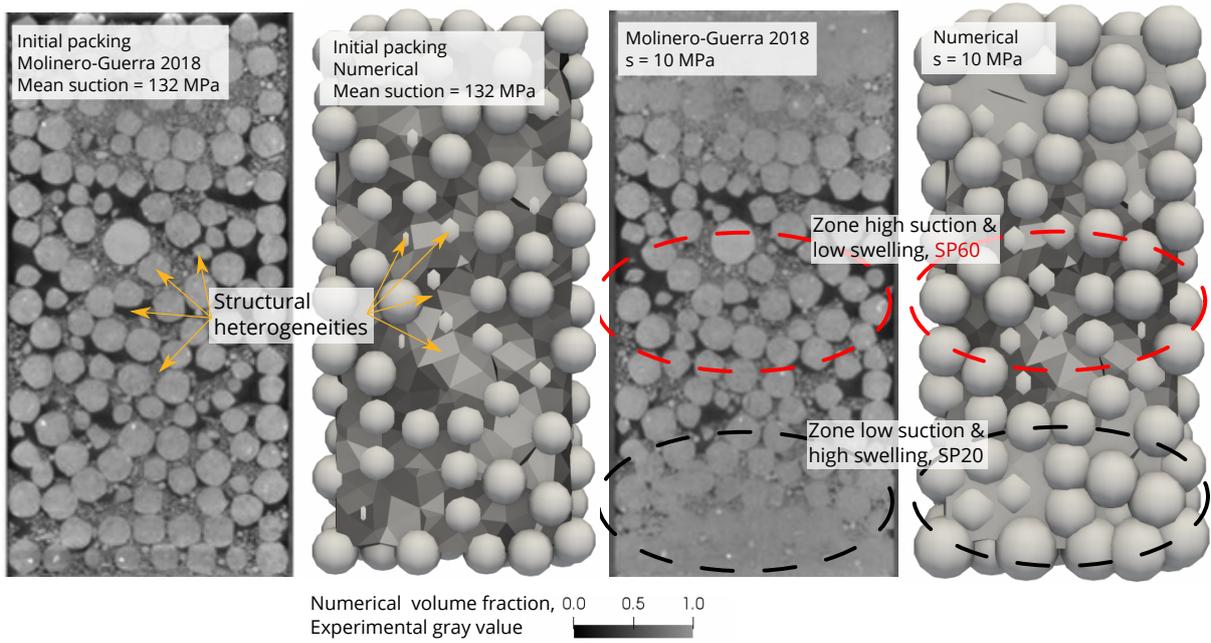


Figure 4.10: Powder and pellet swelling distribution, experimental observation vs numerical model $t = 0\%$ (left) and $t = 20\%$ (right).

Figure 4.10 provides a snapshot comparison of the experimental and numerical models. Qualitatively, the heterogeneous swelling matches with a zone of high suction and low swelling in the center of the specimen and the zones of high swelling and low suction near the ends.

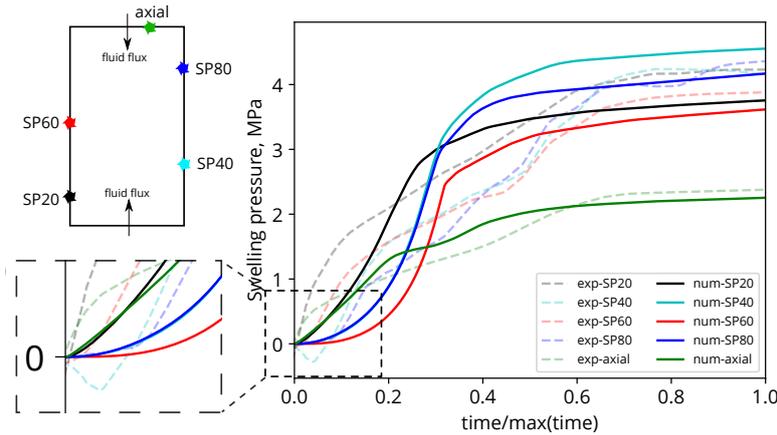


Figure 4.11: Swelling pressure data comparison for numerical and experimental mock-up tests hydration tests.

4.9 Conclusions

In pursuit of a macro-scopic model of the hydro-mechanical processes in heterogeneous partially saturated clay mixtures, the present investigation draws important conclusions for engineered barriers. Most importantly, the investigation proves that a DEM discretization resolved the multi-scale hydro-mechanical behavior of pellet-powder mixtures extremely effectively, without the need for calibration. The investigation also demonstrated the importance of properly treating the local non-linear pressure-volume solution to maintain accuracy. From an applicable point of view, it is clear that the development of swelling pressure depends strongly on initial structural heterogeneous pellet-powder distributions. Thus, results showed that treatment of the structural heterogeneity field is imperative for understanding the evolution of hydro-mechanical processes in engineered barriers. Another important point covered by this investigation is the up-scaling of the micro-scale permeability evolution, which enabled the coarse mesh here to resolve crack-processes. Results show that other micro-scale processes likely play an important role in the development of swelling pressures, however, they do not affect the major trends observed during hydration. Finally, the numerical implementation of the macro-scale conceptual model is flexible and ready for advanced research on engineered barriers.

4.10 Future advancements

The presented model and results are encouraging. Beyond a consistent theoretical foundation, the practical implementation is open-sourced and written to easily support additional modifications. This combination enables a variety of important additions. First, a strict statistical analysis of hydro-mechanical behavior as a function of the initial heterogeneity is important. However, treatment of the heterogeneity must be properly implemented using random fields to maintain meso-scale spatial correlations of powder distributions. Using this statistical framework, another analysis of the the partially saturated permeability field should be performed.

The swelling pressure of the powder currently is not actively applied to the walls of the container, instead the swelling pressure must direct through particles. This needs to be improved by considering appropriate geometric resolutions between the cells and a curved surface. This may be an opportunity for further development of the alpha boundary model described in Sec. 3.4.

Further analysis and data mining should be performed to extract additional insights. For example, a quantitative analysis of the swelling volume along the hydration axis for comparison with experimental imagery.

Computational acceleration of the hydro-mechanical coupling

5

5.1 Summary

This study details the acceleration techniques and associated performance gains in the time integration of coupled poromechanical problems using the Discrete Element Method (DEM) and a Pore scale Finite Volume (PFV) scheme in Yade open DEM software. Specifically, the model is tailored for accuracy by reducing the frequency of costly matrix factorizations (matrix factor reuse), moving the matrix factorizations to background POSIX threads (multithreaded factorization), factorizing the matrix on a GPU (accelerated factorization), and running PFV pressure and force calculations in parallel to the DEM interaction loop using OpenMP threads (parallel task management). Findings show that these four acceleration techniques combine to accelerate the numerical poroelastic oedometer solution by 170x, which enables more frequent triangulation of large scale time-dependent DEM+PFV simulations (356 thousand+ particles, 2.1 million DOFs).

Some of the contents of this chapter were also published in the *Journal of Computer Physics Communications*

R. Caulk, E. Catalano, and B. Chareyre. 'Accelerating Yade's poromechanical coupling with matrix factorization reuse, parallel task management, and GPU computing'. In: *Computer Physics Communications* 248 (2020).

5.2 Methods

5.2.1 Discrete Element Method contact model

Following the standard DEM formulation outlined in Sec. 5.2.1, the contact law (Eq. 2.3) for the present model follows a standard Hook's law:

$$\mathbf{f}_{n,ij} = k_{n,ij} \Delta D_{ij} \cdot \mathbf{n}_{n,ij} \quad (5.1)$$

where $(\mathbf{f}_{n,ij})$ is the normal force between particles i and j , $k_{n/s}$ are the normal and shear stiffnesses, ΔD_{ij} is the displacement between particles, and $\mathbf{n}_{n,ij}$ is the unit vector parallel to the interaction between particles. Since the shear force depends on the orientation of both particles, it is updated incrementally:

$$\Delta \mathbf{f}_{s,ij} = k_{s,ij} \Delta d_{s,ij} \cdot \mathbf{n}_{s,ij} \quad (5.2)$$

$$\mathbf{f}_{s,ij}^t = \mathbf{f}_{s,ij}^{t-\Delta t} + \Delta \mathbf{f}_{s,ij} \quad (5.3)$$

where $\mathbf{f}_{s,ij}^t$ is the shear force between particles i and j at time step t , $\mathbf{n}_{s,ij}$ is the unit vector perpendicular to the particle interaction, $\Delta u_{s,ij}$ is the tangential displacement and $k_{s,ij}$ is simply a fraction of $k_{n,ij}$, (k_s/k_n). Finally, the traction on a particle i interacting with n neighbors becomes:

$$\mathbf{f}_i = \sum_{j=1}^n (\mathbf{f}_{n,ij} + \mathbf{f}_{s,ij}) \quad (5.4)$$

which is used in the time integration of Eq. 3.1.

5.2.2 Pore Finite Volume (PFV) Scheme

[91]: Catalano et al. (2011), 'A pore-scale hydro-mechanical coupled model for geomaterials'

[57]: Chareyre et al. (2012), 'Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings'

[58]: Catalano et al. (2014), 'Pore-scale modeling of fluid-particles interaction and emerging poromechanical effects'

Yade's PFV scheme was introduced by [91], [57], and [58]. Refer to [57] for a thorough description of the poroelastic model, the pore network, and fluid-particle force approximations. In summary, the Discrete Element sphere locations are regular delaunay triangulated to form a tetrahedral mesh. Each tetrahedral is comprised of four discrete elements and represents a single pore comprised of solid and fluid fractions. The total network of tetrahedrals constitutes a pore network, which is used to establish a Stokes-flow. Assuming small Reynolds and large Stokes numbers, the continuity equation can be written as a surface integral:

$$\dot{V}_{p,i} = \int_{\partial\Theta_i} (\mathbf{u} - \mathbf{v}) \cdot \mathbf{n} dS \quad (5.5)$$

where $\dot{V}_{p,i}$ is the pore volume change, $\partial\Theta_i$ is the pore contour, and \mathbf{u} is the fluid velocity relative to the contour velocity \mathbf{v} . Since the solid area of the pore will not change, $\partial\Theta_i$ can be reduced to only the fluid fractions (S_{ij}^f) of the pore contour. Thus, the integral can be represented as the sum of fluid fluxes exchanged by each pore and its four neighbors ($j=1$ to 4):

$$\dot{V}_{p,i} = \sum_{j=1}^4 \int_{S_{ij}^f} (\mathbf{u} - \mathbf{v}) \cdot \mathbf{n} dS = \sum_{j=1}^4 q_{ij}. \quad (5.6)$$

Flux (q_{ij}) through the pore throat connecting pore i and j is approximated by the local pressure gradient:

$$q_{ij} = k_{ij} \frac{p_i - p_j}{l_{ij}} \quad (5.7)$$

where p_i and p_j are the pressures of neighboring pores and l_{ij} is the length of the connecting pore throat. The hydraulic conductance, $g_{ij} = k_{ij}/l_{ij}$, can be approximated using Poiseuille, the details of which can be found in [57].

Finally, a linear system can be constructed based on the pressure at time $t + \Delta t$ as a function of the volume changes at t :

$$\sum_{j=1}^4 g_{ij} \left(p_i^{[t+\Delta t]} - p_j^{[t+\Delta t]} \right) = \dot{V}_{p,i}^{[t]} + Q_i^{[t]} \quad (5.8)$$

where Q_i is a source term for pore i . The matrix representation of the full linear system is simply the known conductivity matrix \mathbf{G} comprising the g_{ij} coefficients from Eq. 5.8 for all i , the unknown pressures listed in a vector \mathbf{p} , and the vector of rate of volume changes \mathbf{V} . \mathbf{V} depends linearly on particles velocities, which can be expressed by an operator \mathbf{E} such that $\mathbf{V} = \mathbf{E}\mathbf{x}$. The instantaneous pressures-velocities relation finally reads:

$$\mathbf{G}\mathbf{p} = \mathbf{E}\mathbf{x} + \mathbf{Q} \quad (5.9)$$

\mathbf{G} is sparse, symmetric, and positive definite (shown in Figure 5.1). Therefore, Cholesky decomposition is employed for the decomposition of \mathbf{G} to a lower triangular matrix multiplied by its transpose ($\mathbf{L}\mathbf{L}^T$). The decomposed matrix, i.e. the factor, can be used to solve for \mathbf{p} by first using forward substitution followed by back substitution:

$$\mathbf{L}\mathbf{y} = \mathbf{x} \quad (5.10)$$

$$\mathbf{L}^T\mathbf{p} = \mathbf{y} \quad (5.11)$$

thus avoiding the prohibitively expensive inversion of \mathbf{G} for the solution of \mathbf{p} . The drag forces on the particles (\mathbf{f}_D) are obtained after multiplication of the pressure vector by a matrix \mathbf{F} whose components reflect projected area:

$$\mathbf{f}_D = \mathbf{F}\mathbf{p} \quad (5.12)$$

As discussed and quantified throughout the remainder of the paper, the computational expense of the poroelastic DEM+PFV coupling is not insignificant. However, the introduction of poroelasticity can compound the computational slowdown by also reducing the maximum stable time step. As demonstrated in Sec.5.7, as soon as typical DEM stiffness effects (the natural period of a spring mass system) become negligible compared to viscous effects (fluid drag forces acting like dampers) the maximum time step depends on the maximum eigenvalue of the viscous system. It is not uncommon for a poroelastic simulation of granular material to operate at a time step equal to one order of magnitude lower than its dry counterpart. Thus, the need for the acceleration techniques highlighted herein is even more pertinent.

5.3 Acceleration techniques

5.3.1 Matrix Factor Reuse

[92] showed how the factorization of $[\mathbf{G}]$ consumes ca. 98% of the total flow solver time. In comparison, the simple process of forward and back substitution into the factor for the solution of $\{\mathbf{P}\}$ is negligible. For this reason, total factorizing is reduced by reusing the costly factor for multiple right-hand solves (refer to Figure 5.3 to see the relationship of matrix factor reuse to the rest of Yade's DEM+PFV algorithms). In other words, as long as the deformation criterion (Eq.5.13) is satisfied, the factor

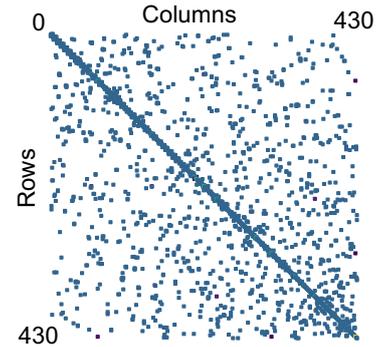


Figure 5.1: Example of a 430 DOFs positive definite, symmetric, banded, sparse conductivity matrix (\mathbf{G})

[92]: Catalano (2012), 'A pore-scale coupled hydromechanical model for biphasic granular media. Application to granular sediment hydrodynamics'

is reused for the duration of a remesh interval, λ_{rm} . This factor-reuse reduces the cost of determining $\{\mathbf{P}\}$ by an order of magnitude since the expensive factorization is not repeated. The negligible effect of remesh interval during a quasi-static geomechanical oedometer test is confirmed by comparing pressure at the same location and time (Sec.5.4.2) for nine different remesh intervals (Fig. 5.2a). Pressure differences are negligible and random, owing to the effect of force summation order in parallel environments for DEM, as shown by the replicate rests run for Fig. 5.2b. Both analyses demonstrate how matrix factor reuse does not significantly impact the solution of the quasi-static oedometer simulation used for performance bench-marking throughout the remainder of this paper.

In dynamic simulations associated with large deformations, the remesh interval depends on deformation criteria. For instance, the criterion

$$\max(\varepsilon_{v,i}^{t_0 \rightarrow t}) < 0.01 \quad (5.13)$$

can be used where $\varepsilon_{v,i}^{t_0 \rightarrow t}$ is the volume change of pore i since last remesh. Remeshing would be triggered when that condition is not satisfied. Auxiliary analyses compared this remesh criterion to remeshing at each interval and concluded that geometrical and mechanical variables are sufficiently representative of the state of the medium during deformation, to yield accurate results.

5.3.2 Multithreaded factorization

Despite accelerating the solution, the matrix factor reuse scheme described in Sec. 5.3 still requires the DEM simulation to stop at the end of a remesh interval and perform both the retriangulation of the pore network and the factorization of $[\mathbf{G}]$. During this interim step, the DEM+PFV simulation cannot continue stepping through time since it needs to wait for the new conductivity matrix before it can obtain pore pressures and the associated viscous and pressure forces. To address this weakness, a multithreaded scheme was added to Yade's PFV with the objective of retriangulating the pore network and factorizing $[\mathbf{G}]$ on background POSIX threads while the DEM+PFV simulation steps forward with a previous pore network and prefactorized $[\mathbf{G}]$ on foreground OpenMP threads (refer to Algorithm 1 and Figure 5.3). This multithreaded configuration will improve performance for all simulations associated with any λ_{rm} , but there exists an optimal λ_{rm} that will yield *uninterrupted* time stepping

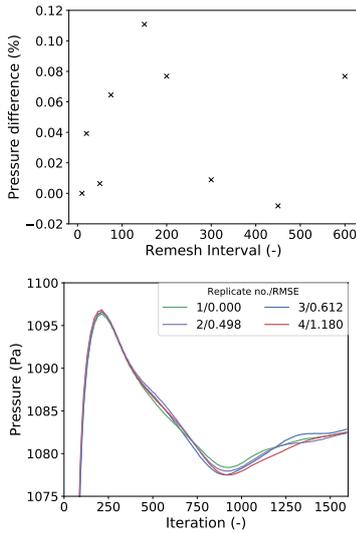


Figure 5.2: a) Pressure difference for various remesh intervals after 1200 iterations and 10 parallel cores for DEM force summations b) Replicate tests using Remesh interval = 20 iterations and 10 parallel cores for DEM force summations

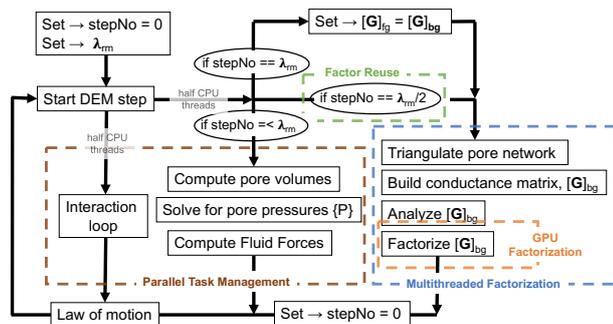


Figure 5.3: Yade DEM+PFV accelerated algorithm overview.

through the coupled DEM+PFV simulation provided the time required to retriangulate the pore network and factorize $[G]$ is less than the time it takes the coupled DEM simulation to step through $\lambda_{rm}/2$ steps. In other words, the optimal λ_{rm} for uninterrupted simulation is dictated by the speed of the simulation (v_{iter} , iter/sec) and the background time (t_{bg} , s):

$$\lambda_{rm} \geq 2t_{bg}v_{iter} \quad (5.14)$$

Algorithm 1 Multithreaded triangulation and factorization

```

simulationRunning  $\leftarrow$  simulation activity boolean
stepNo  $\leftarrow$  number of steps since last remesh
 $\lambda_{rm}$   $\leftarrow$  remesh interval
Foreground simulation ( $f_g$ )
while simulationRunning = True do
  foreground OpenMP threads solve for pore pressure at each time
  step by reusing:
    triangfg  $\leftarrow$  foreground pore network
    factorfg  $\leftarrow$  foreground factorization
  Background factorization ( $b_g$ )
  if stepNo =  $\lambda_{rm}/2$  then
    background POSIX threads retriangulate pores and build/factor
    conductivity matrix:
      triangbg  $\leftarrow$  retriangulate pore network
      factorbg  $\leftarrow$  factorize conductivity matrix
    end if
    if stepNo =  $\lambda_{rm}$  then set new bg solver to fg:
      triangfg = triangbg
      factorfg = factorbg
      stepNo = 1
    end if
  end while

```

5.3.3 GPU Accelerated Factorization

The present study aims to reduce the heavy cost of $[G]$ factorization in Eq. 5.9 by leveraging GPU computing. In particular, the PFV scheme presented here employs ‘CHOLMOD’, a GPU accelerated sparse matrix solver part of the open source SuiteSparse C library [93]. CHOLMOD provides Cholesky decomposition, it builds an elimination tree of the matrix based on a METIS partitioning, and sends subtrees directly to the GPU for factorization [94]. The subtree algorithm is highly optimized to reduce the volume of data exchange between the GPU and the CPU.

5.3.4 Parallel Task Management

The final acceleration technique, called Parallel Task Management (Figure 5.3), exploits the highly parallel nature of DEM’s interaction detection and force collection methodologies. Since the time integration of particle movement depends solely on the traction from the current time step

(Eq. 3.1), fluid forces can be collected in parallel just like the particle-particle forces are collected in parallel. As shown in Figure 5.3, the fluid force algorithm is initiated on a separate set of OpenMP threads from the contact detection threads. DEM forces and fluid forces are combined before the final integration step.

5.4 Test setup

5.4.1 Computer Details

All simulations presented in this study were performed on a scientific workstation containing the following hardware:

- ▶ **CPU** Xeon 2680 v2 E5 2.8 GHz 10 core processor, 448 GFLOPS double precision
- ▶ **GPU1** GeForce 1050 Ti, 4 GB RAM, 1392 MHz, 32 cuda cores, 61.9 GLOPS double precision
- ▶ **GPU2** Tesla K20, 5 GB RAM, 2496 MHz, 706 cuda cores, 1175 GFLOPS double precision, ECC=ON
- ▶ **RAM** 64 GB 1866 MHz
- ▶ **Storage** 500 gb SSD 600 MB/s read/write

and the software presented to the left:

Software used for study and analysis:

- ▶ Linux Ubuntu 18.04
- ▶ Yade git-28917a9
- ▶ OpenMP parallelization
- ▶ SuiteSparse 4.6.0-beta
- ▶ CUDA 9.0
- ▶ Nvidia 384.11 GPU drivers

[58]: Catalano et al. (2014), 'Pore-scale modeling of fluid-particles interaction and emerging poromechanical effects'

5.4.2 Model details

The DEM+PFV performances of multi-core CPU, GeForce 1050 Ti GPU, and Tesla K20 GPU conductivity matrix factorizations (Eq.5.11) were evaluated using a pre-validated [58] consolidation test of a saturated soil packing (example script *). The DEM sphere packing is cubically sized from $8e-6$ to $3.4e-3$ m³ (Figure 5.4) with microparameters as shown in Table 5.1. The fluid and mechanical boundary conditions follow traditional oedometer boundary conditions as shown in Figure 5.4: enclosing walls impose a deviatoric stress of 1 kPa (Neumann) in the Y direction and maintain fixed displacement (Dirichlet) in the X and Z directions. Meanwhile, fluid boundary conditions include drained (Dirichlet - imposed pressure of 0 Pa) at the top Y cube face and impermeable (Neumann - no flux) on the remaining cube faces. All flow is calculated using the dynamic viscosity of water $\mu=1$ cP. Both mechanical and fluid time steps are set constant to $1e-6$ s and the simulation proceeds for 600 time steps with $\lambda_{rm}=200$. These time-steps and duration are irrelevant, they simply maintain stability so that the performance of various aspects of the algorithm can be measured.

5.4.3 Data description

A parametric sweep was performed for three device types and six problem sizes, resulting in 18 total simulations. For each parametric combination, six distinct timings were collected and averaged for each of the following seven algorithms:

* [GitHub: yade/trunk/examples/oedometer.py](https://github.com/yade/trunk/examples/oedometer.py)

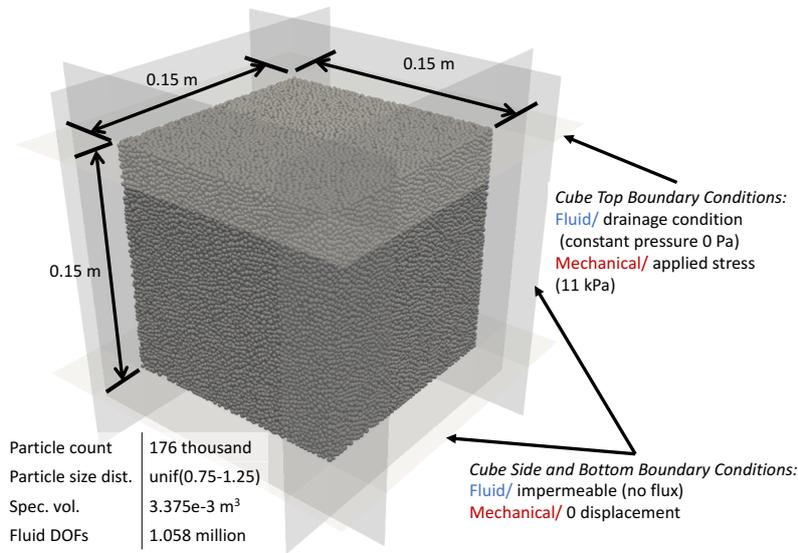


Figure 5.4: Example of one of the cubical DEM+PFV 1-D consolidation models used to test performance of GPU accelerated factorization.

- (1) Build the system of linear equations
- (2) Allocate the system to memory
- (3) Analyze the system (identify non-zero pattern and build elimination tree)
- (4) Factorize the system ($[G]$ matrix decomposition)
- (5) Solve the system (forward/backward substitution into factor)
- (6) Compute pore volumes
- (7) Compute fluid forces (pressure and viscous forces)

Additionally, the simulation speed and total time to step 600 iterations of each parametric combination was collected. In total, 972 data points were used to generate the parametric sweeps presented in Sec. 5.5.

Micro parameter	Value (DEM)
E_i	1 MPa
k_s/k_n	0.5
ϕ_b	30°
γ_{int}	1.329
Sphere radius	unif(0.75 mm, 1.25 mm)
Sphere density	2600 kg/m ³

Table 5.1: Numerical specimen DEM microproperties

5.5 Results and Discussion

Results show how the combined acceleration techniques of matrix factor reuse, multithreaded factorization, GPU accelerated factorization, and parallel task management improve performance by 170x (Figure 5.5), enabling continuous simulation of poroelastic problems reaching 2.1 million DOFs on an office workstation. The first acceleration technique, **matrix factor reuse**, has the greatest impact on performance by reducing the frequency of rebuilding, reanalyzing, and refactorizing the conductivity matrix (5.9) according to the selected remesh interval. Results show that these operations consume up to 140 seconds for a system with 2.1 million DOFs (Figures 5.6 and 5.7). Without matrix factor reuse, these expensive operations are performed every iteration despite only being necessary

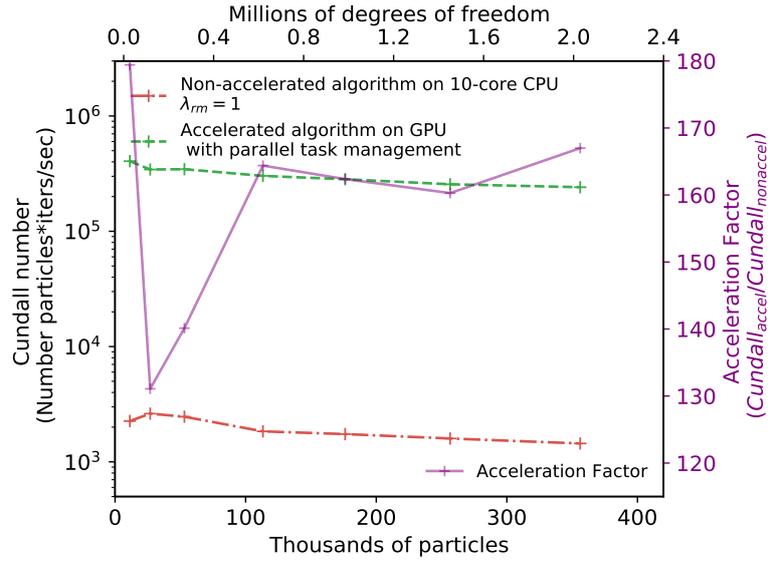


Figure 5.5: Performance comparison for non-accelerated and fully accelerated algorithms.

after large deformations (Eq. 5.13) (i.e. matrix factor reuse acceleration is proportional to the selected remesh interval and the dynamics of the system).

The second acceleration technique, **GPU accelerated factorization**, decreases the conductivity matrix factorization (Eq. 5.11) time by 75% compared to a 10-core CPU for 2.1 million DOFs (Figure 5.6). However, the total t_{bg} is only decreased by 50% due to the single-threaded analyze step comprised of matrix graph partitioning and preconditioning. Although most simulations require the costly analyze step, certain stiff poroelastic simulations benefit from its elimination since it simply reorders and prepares the matrix for factorization. For example, the Discrete Fracture Network model in Yade benefits from reusing the matrix reordering for subsequent factorizations since the non-zero pattern remains constant. Both matrix factor reuse and GPU accelerated factorization contribute to significant gains in performance for the poroelastic oedometer simulation, while the third technique, **multithreaded factorization**, removes the computational time associated with the conductivity matrix factorization by parallelizing the operations with the primary DEM simulation. Therefore, multithreaded factorization increases the optimal remesh frequency associated with an uninterrupted simulation (i.e. conductivity matrix factorization occurs in less time than the time required for the primary simulation to step through one remesh interval Eq. 5.14). As shown in Figure 5.8 the time spent per iteration is almost identical for all three devices, which means that the factorization is fully backgrounded in these oedometer simulations. However, it is worth noting that the poroelastic simulation runs 10% faster when the GPU participates, suggesting CPU resources are less strained when the burden of factorization is taken by the GPU. In any case, the optimal remesh interval and Cundall numbers, show how the GPU is only beneficial for cubical packings ≥ 30 thousand particles. Larger cubical packings comprised of $\geq 30k$ particles allow the Tesla K20 to improve λ_{rm} by up to 42%, which means the Tesla K20 enables the update of $\mathbf{[G]}$ almost two times more frequently than the 1050 Ti for cubical packings comprised of 356 thousand particles. Fig. 5.8 also shows how the GPU increases the Cundall number by up to 12% for 180k

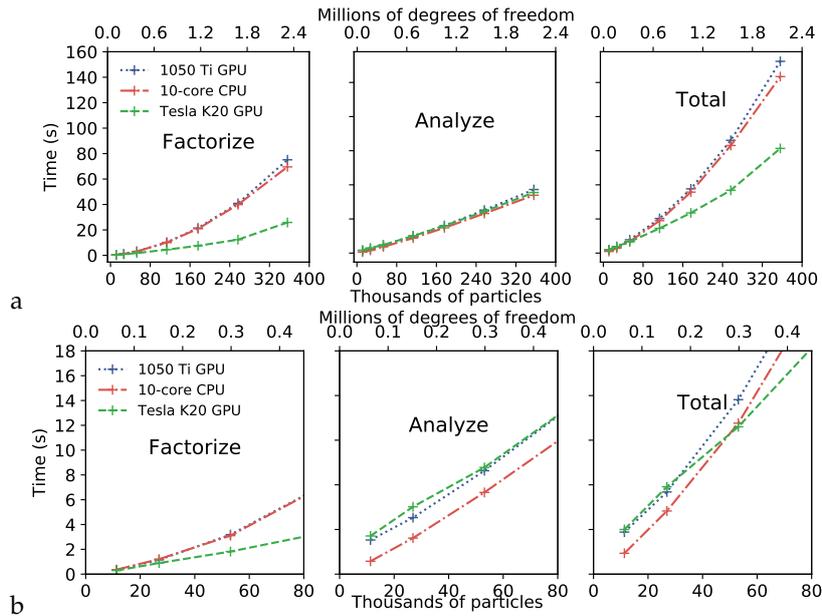


Figure 5.6: a) Time required to factorize and analyze the conductivity matrix (Eq. 5.9). $t_{bg} = t_{factor} + t_{analyze}$ b) Zoomed in to show devices timings for small packings (bottom)

particle packing. Meanwhile, for small cubical packings comprised of ≤ 30 thousand particles, the time spent moving information to and from the GPU outweighs the time saved by the accelerated GPU factorization, resulting in less favorable λ_{rm} and Cundall numbers compared to the 10 core CPU.

The final acceleration technique, **parallel task management**, accelerates the coupled solution by ca. 1.2x (Fig. 5.9). A closer look shows how the time spent on these parallelized algorithms (solving for pore pressures, computing pore volumes, and computing fluid forces) is nearly equivalent to the time spent running the full DEM interaction loop for large particle packings (Figure 5.8 and 5.10). The result is misleading since it implies that a coupled simulation should run at exactly the same speed as an uncoupled simulation (provided equivalent core counts). In fact, auxiliary tests show that uncoupled DEM tests run approximately 1 order of magnitude faster than the parallelized coupled DEM+PFV simulation. Ultimately, the CPU L2 and L3 cache sizes in addition to the RAM speed likely limit the linear scalability of increased instruction requests for the coupled simulation in the highly parallelized environment. The point is supported by Figure 5.10, showing that requesting CPU resources for factorization in addition to foreground FlowEngine algorithms slows the simulation down by 17%. Instead, the GPU factorization technique frees up CPU resources for computing foreground FlowEngine algorithms. Finally, the acceleration benefit for parallel task management decreases as the the system sizes increase and the time spent factorization the matrix dominates the total simulation time (Fig. 5.9).

5.5.1 Increasing mesh resolution for partially saturated problems

The acceleration techniques outlined in the present Chapter enable the increase of mesh resolution by a full order of magnitude as shown in Fig. 5.11. This advancement enabled the mesh resolution investigation

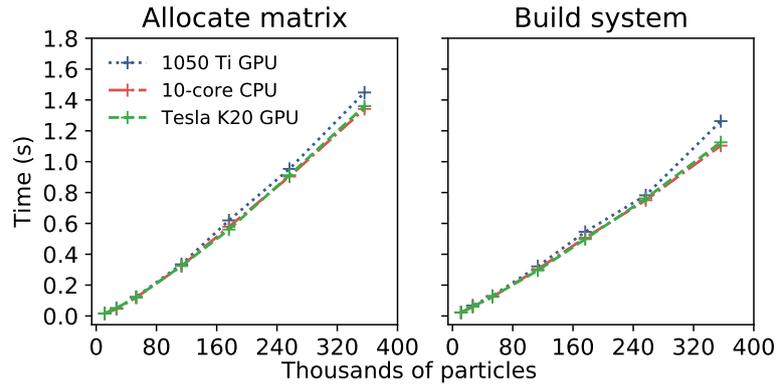


Figure 5.7: Time required to allocate conductivity matrix (Eq. 5.9) to memory (left) and build the system of equations (Eq. 5.9) (right)

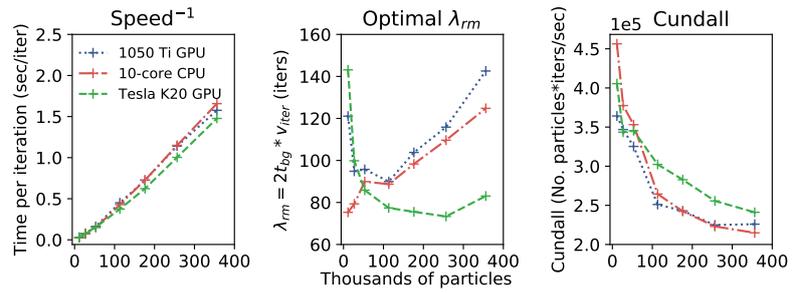


Figure 5.8: Time per iteration, optimal remesh interval (λ_{rm}) associated with v_{iter} and t_{pg} , and Cundall number for various conductivity matrix (Eq. 5.9) sizes

presented in Sec. 3.5.8 and well as the statistical investigation of scale in Sec. 3.5.9.

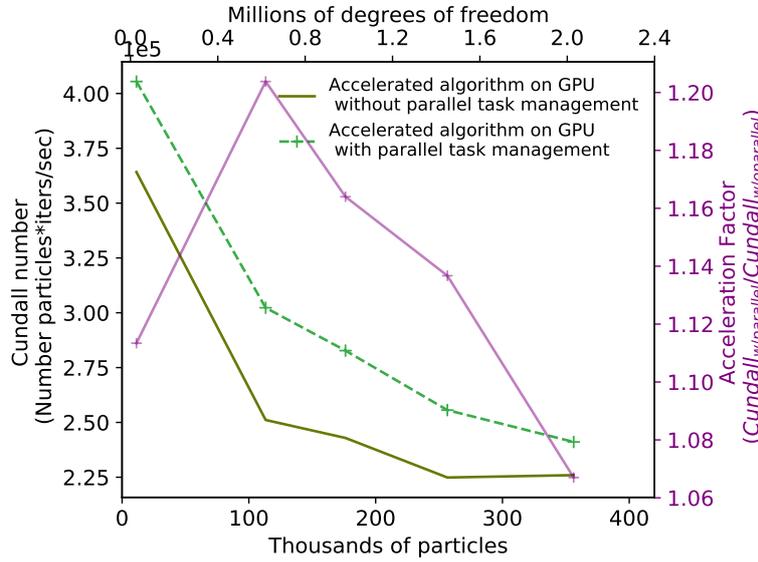


Figure 5.9: Performance gain from implementing OpenMP parallel task management (Fig. 5.3).

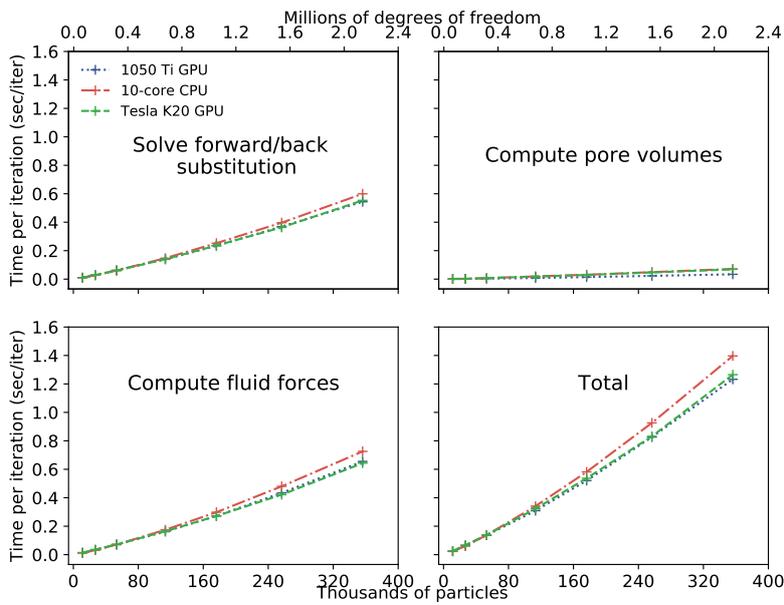


Figure 5.10: Wall time spent solving for pressures (Eq. 5.11), computing pore volumes, and computing forces (Eq.5.12) (parallel task management in Fig. 5.3).

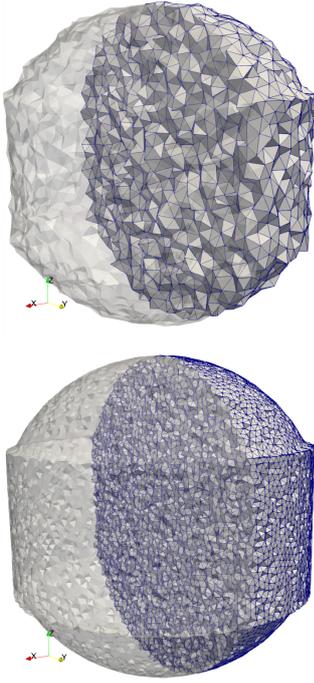


Figure 5.11: Mesh resolution limit before, $7e4$ degrees of freedom (top) and after acceleration techniques, $7e5$ degrees of freedom (bottom).

5.6 Conclusions

Yade's poromechanically coupled DEM+PFV scheme was accelerated by 170x by combining four techniques: matrix factor reuse, multithreaded factorization, GPU accelerated factorization, and parallel task management. Each technique ameliorated different weaknesses associated with the time-dependent implicit pore finite volume scheme. First, **matrix factor reuse** has the largest impact of on performance by reducing the frequency of the costly conductivity matrix factorization. Second, **multithreaded factorization** parallelizes the costly factorization in the background while the coupled simulation steps through time, thus reducing computational cost and enabling an increase of factorization frequency without additional computational time. Third, **GPU accelerated factorization** moves the computational cost of matrix factorization to a Tesla K20 GPU where it factorizes the matrix 75% faster for large poroelastic problems, thus further increasing the factorization frequency by 42% without adding computational time. Finally, **parallel task management** accelerates the solution by 30% by parallelizing auxiliary PFV algorithms (e.g. volume calculations and force calculations) with the DEM interaction loop. All techniques combined enable the simulation of poroelastic systems comprised of of 2.1 million DOFs (356 thousand particles) on an office workstation. After reducing the cost of factorization, the new limitation lies in CHOLMOD's analysis step, which orders the matrix and builds the elimination tree on a single thread. However, specialized stiff fracture network simulations can avoid this step entirely by reusing the matrix ordering for subsequent factorizations. Future improvements will focus on the parallelization of the analysis step, MPI solutions for systems with 10s of millions of DOFs, and updating/downdating matrix factorizations depending on the magnitude of rank change.

5.7 Stability of the coupled algorithm

In coupled simulations, the fluid surrounding particles acts as a viscous damper, which results in a force to be added to the contact forces. After substituting the drag forces with equations 5.9 and 5.12 the Newton's second law of motion can be written, formally:

$$\mathbf{M}\ddot{\mathbf{x}}^{[t]} + \mathbf{V}^{[t]}\dot{\mathbf{x}}^{[t]} + \mathbf{K}^{[t]}\mathbf{x}^{[t]} = 0 \quad (5.15)$$

where \mathbf{x} is the generalized particle position, \mathbf{M} and \mathbf{K} express the global mass and stiffness matrices, and the viscous matrix is comprised of the inverted conductivity matrix, the global force matrix and the global volume rate matrix, $\mathbf{V} = \mathbf{F}\mathbf{G}^{-1}\mathbf{E}$. The stability of the explicit time integration scheme for this equation is now discussed by considering two limit cases: stiffness dominated regimes and viscosity dominated regimes.

In stiffness dominated systems, the stability of the oscillating spring-mass system is simply a function of the natural period of the system (see the appendix of [95] for a detailed derivation):

$$\Delta t_{M-K} = \min \left(\sqrt{m_k / K_{k,i}} \right) \quad (5.16)$$

where m_k and $K_{k,i}$ are particle k mass and equivalent stiffness (considering all particle k contacts and degrees of freedom, i).

If stiffness effects are negligible compared to viscous effects, we derive the stability criteria as follows:

$$\ddot{\mathbf{x}}^{[t]} + \mathbf{V}^{[t]}\mathbf{M}^{-1}\dot{\mathbf{x}}^{[t]} = 0 \quad (5.17)$$

$$\frac{\mathbf{x}^{[t+\Delta t]} - \mathbf{x}^{[t]}}{\Delta t} + \mathbf{V}^{[t]}\mathbf{M}^{-1}\mathbf{x}^{[t]} = 0 \quad (5.18)$$

$$\mathbf{x}^{[t+\Delta t]} = \left(\mathbf{I} - \mathbf{V}^{[t]}\mathbf{M}^{-1}\Delta t \right) \mathbf{x}^{[t]} \quad (5.19)$$

let $\mathbf{VM}^{-1} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$ be the eigenvalue decomposition of \mathbf{VM}^{-1} , where $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues and \mathbf{U} is the matrix of eigenvectors. Plugging $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$ into Eq. 5.19 yields:

$$\mathbf{U}^{-1}\mathbf{x}^{[t+\Delta t]} = \mathbf{U}^{-1}\mathbf{x}^{[t]} - \Delta t\mathbf{\Lambda}\mathbf{U}^{-1}\mathbf{x}^{[t]} \quad (5.20)$$

and the transformation between between coordinates is denoted as $\mathbf{y}^{[t]} := \mathbf{U}^{-1}\mathbf{x}^{[t]}$:

$$\mathbf{y}^{[t+\Delta t]} = (\mathbf{I} - \Delta t\mathbf{\Lambda})\mathbf{y}^{[t]} \quad (5.21)$$

which is a set of scalar equations since $\mathbf{\Lambda}$ is diagonal. Thus, stability is ensured by imposing:

$$|1 - \Delta t\lambda_{max}| < 1 \quad (5.22)$$

where λ_{max} is the maximum eigenvalue of the matrix $\mathbf{V}^{[t]}\mathbf{M}^{-1}$. Finally, the viscous dominated timestep is computed as:

$$\Delta t_{M-V} = \Delta t < 2 \cdot \lambda_{max}^{-1} \quad (5.23)$$

In an attempt to relieve the computational expense associated with determining the eigenvalues of $\mathbf{V}^{[t]}\mathbf{M}^{-1}$ (which would need to invert \mathbf{G}), a parametric analysis was performed to investigate the distribution of viscous coefficients for polydispersed granular packings. In brief, a non-zero velocity was imposed on each particle and the resulting viscous force was measured. An empirical upper bound of $m_k/\mathbf{v}_k^{[t]}$ in dense packings was found as:

$$\frac{m_k}{\mathbf{v}_k^{[t]}} < \frac{1}{8000} \frac{\pi\rho_k\phi_k^2}{\mu} \quad (5.24)$$

where ρ_k , ϕ_k , and μ are particle k density, particle k diameter, and fluid viscosity. Considering $\min(m_k/\mathbf{v}_k^{[t]}) \approx \lambda_{max}^{-1}$, the empirical estimate for

$m_k/\mathbf{v}_k^{[t]}$ is inserted into Eq. 5.23 to yield a fast estimate of the maximum viscous timestep as:

$$\Delta t_{M-V} < 2 \cdot \min \left(\frac{1}{8000} \frac{\pi \rho_k \phi_k^2}{\mu} \right) \quad (5.25)$$

The final maximum allowed timestep for the coupled scheme in viscous or stiffness dominated regimes is as follows, where the 0.8 pre-factor is enough to ensure stability even in mixed elastic-viscous regimes:

$$\Delta t = 0.8 \min(\Delta t_{M-V}, \Delta t_{M-K}) \quad (5.26)$$

5.8 Practical reproduction of results

Readers interested in reproducing the results or simply using the model presented here can access it as part of Yade DEM opensource software [88]. The entire code can be freely installed by typing:

[88]: Yade (2021), *Yade source code*

```
sudo apt-get install yade
```

into any Ubuntu linux terminal. Installation instructions for other linux variants can be found at [89]. Once the code is installed, the user can run the input script provided as supplementary material at <http://u.pc.cd/slYrtalk> or found in Appendix .2 (Chapter5_example_script.py) by executing the following command in a terminal:

```
yade Chapter5_example_script.py
```

Any reader who seeks additional assistance in running or modifying the code provided here, should seek assistance from the author at answers.launchpad.net/yade.

CONCLUSION

Conclusions and Perspectives

The objective of the present thesis was to model the fundamental processes responsible for the complex behavior of heterogeneous partially-saturated swelling-clay material subjected to anisotropic hydraulic and gas loadings. In this context, particular attention was paid to understanding the persistence of initial heterogeneities on swelling pressures and gas migration. In pursuit of this objective, two conceptual models were developed, which were designed to elucidate the hydro-mechanical behaviors of swelling clay at two separate scales. These conceptual models were implemented numerically using a variety of methods including the Discrete Element Method (DEM), the Pore-Finite Volume (PFV) method, and finite difference methods. The practical implementation of these methods required a set of mature tools including standard workstation hardware, cluster compute power (GRICAD), and GPU compute power. Further, the implementation includes the use of an advanced set of features by C++ and Python programming languages as well as state-of-the-art third party libraries¹.

In the end, this thesis project was a unique approach to understanding the *most important* aspects of partially saturated swelling clays in engineered barriers. That is to say, we explicitly admitted that we would not study *all* aspects of partially saturated clay. Such a simple admission to simplicity (if one could call the study of hydro-mechanical behaviors in heterogeneous partially saturated clay, *simple*) unleashed an incredibly fruitful multi-scale investigation. It means that the thesis project was never bogged down by extra complexities such as yield curves, strain hardening, compressibility coefficients, etc. This admission to simplicity means that the thesis project is focused, principally, on other hydro-mechanical features such as solving the pressure-volume equations in pellet-powder mixtures (Chap. 3), modeling crack developments from heterogeneities (Chap. 3), and understanding/improving the practical limitations of our tools (Chap. 5). In the end, the research outlined in this thesis produced applicable conclusions to nuclear waste seal design as well as reusable/modifiable scientific methods/tools. To be precise, the first conceptual model/tool focused on understanding the complex interactions between initial heterogeneity, water-retention, volumetric swelling, and swelling pressures in pellet-powder mixtures. The second conceptual model focused on understanding the development of crack networks within individual compacted clay pellets and their effect on permeability evolution during hydration. Additionally, the method itself was analyzed and accelerated to improve the domain size constraints.

6.1 Hypothesis

The current state of knowledge regarding the hydro-mechanical behavior of partially saturated engineered barriers remains insufficient to make long-term predictions with high levels of confidence (Chap. 2). One of the main reasons for this deficiency is the complex non-linear interactions

1: It is important to note that *all* software used for this thesis and *all* tools developed during the thesis are open sourced. This includes a variety of key softwares including, but not limited to:

- ▶ Linux
- ▶ Paraview
- ▶ Yade
- ▶ Python
- ▶ TexStudio
- ▶ Matplotlib
- ▶ NumPy

between swelling, porosity, and water retention. These complexities are magnified with the inevitable presence of material and structural heterogeneities at multiple spatial scales.

Thus, we hypothesized that these multi-scale heterogeneities control the evolution of permeability in anisotropic loading conditions.

Complimentary to the main hypothesis, we postulated that the Discrete Element Method would provide a deeper understanding of these processes due to its discontinuous nature and striking geometrical resemblance to pellet-powder mixtures.

6.2 Major conclusions

The contents of this thesis support the hypotheses by drawing the following conclusions:

Pellet-powder mixture scale (macro-scale):

- ▶ A DEM discretization resolves the hydro-mechanical behavior of pellet-powder mixtures extremely effectively, with minimal needs for calibration.
- ▶ Treatment of the local pressure-volume solution is imperative for accurate results.
- ▶ The development of swelling pressure depends strongly on initial structural heterogeneous pellet-powder distributions.
- ▶ The processes occurring below the resolved mesh can be resolved using an up-scaled permeability curve, while any neglected processes do not impact the accuracy of final results, but play an important role.
- ▶ The numerical implementation of the macro-scale conceptual model is flexible and ready for advanced research on engineered barriers.

Pellet scale (micro-scale):

- ▶ Crack networks play an important role in gas and water permeability evolution in confined and unconfined conditions.
- ▶ Unimbibed cracks block and redirect water fluxes into channels.
- ▶ Crack networks develop during hydration due to the initial presence of heterogeneities.
- ▶ Confining pressure reduces gas permeability by over one order of magnitude during hydration.
- ▶ Crack fabric homogenizes during hydration.
- ▶ Deviatoric components of permeability tensors elucidate the important role of swelling for anisotropic permeability evolution.

Acceleration:

- ▶ Conductivity matrix factor reuse contributes the largest improvement to performance.

- ▶ Factorizing the next conductivity matrix in parallel with the active simulation enables an optimization of computational power.
- ▶ GPU acceleration makes the conductivity matrix factorization 75% faster for large poroelastic problems.
- ▶ Combining all acceleration techniques leads to a speed up of 170x, which enables larger partially saturated clay hydration simulations.

6.3 Perspectives

These conclusions demonstrate a thorough numerical analysis of the hydro-mechanics in swelling clays. However, the conceptual models and tools presented here open additional questions and research opportunities. From a general perspective, the pellet-powder macro-scale model outlined in Chap. 4 still hosts a wide range of data to be mined. For example, the evolution of force chains between pellets during hydration can provide important information about the role of initial structural heterogeneities (initial packings). Additionally, the swelling powder may migrate throughout the specimen, which would modify the expected behavior and interact with the fluid fluxes. This should be considered in any model advancements. There are also many important details that should also be investigated with respect to the up-scaling mechanism (using the micro-scale permeability curve to inform the macro-scale pellet permeability evolution, Sec. 4.6). For example, Chap. 3 also contains the dynamics of the permeability tensor during hydration - this too can be added to the macro-scale model with little modification. Another important advancement should be a consideration of the material “phase-change” from granular material to a homogeneous slurry at high saturations.

Another important aspect of the pellet-powder macro-scale model is the clay powder distribution within the voids. This powder distribution likely has an important impact on the heterogeneity evolution and thus, on the hydro-mechanical behaviors. This can undoubtedly be resolved using a random field, which will also enable statistical analyses and improved confidence in the effect of heterogeneity on long-term hydration processes.

Regarding the micro-scale model outlined in Chap. 3, there remains a host of additional work to be done. For example, the contact model should be updated to accommodate the decreasing material stiffness with increasing saturation. Another important improvement should be made with respect to hysteresis in the water-retention curve. With respect to the crack model, the simplified geometrical representation of parallel plates can be improved to accommodate tortuosity. Additionally, the present model can accommodate investigations into self-healing cracks for hydration-dehydration cycles.

Finally, in terms of the algorithmic acceleration - the present implementation can be further accelerated by computing new geometric quantities based on DEM particle location, instead of waiting for the full remeshing of vertices. Other acceleration possibilities exist in consideration of rank updates/downdates. Another advancement would be to re-build the pore-finite volume scheme to be MPI compatible.

6.4 Acknowledgments

Work presented here is supported by the Institut de Radioprotection et de Sûreté Nucléaire (IRSN) as well as the Commission canadienne de sûreté nucléaire (CCSN) and Laboratoire 3SR of Univ. Grenoble Alpes. All (or most of) the computations presented in this paper were performed using the UMS GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), which is supported by CNRS Grenoble research institutes.

RÉSUMÉ EN FRANÇAIS

7.1 Introduction Générale

Des systèmes de stockage de déchets nucléaires sûrs, durables et robustes sont primordiaux pour étendre et maintenir le réseau électrique nucléaire existant. Malgré un solide effort de collaboration internationale expérimentale et numérique pour développer ces technologies de stockage, il reste une multitude de variables inconnues associées à l'efficacité de confinement à long terme des joints à base d'argile dans ces installations de stockage. Le problème est aggravé par les volumes croissants de déchets nucléaires de haute activité nécessitant un stockage. Un projet lancé par l'Institut de Radioprotection et de Sécurité Nucléaire (IRSN) au Laboratoire de Recherche Souterrain (URL) à Tournemire, France, intitulé SEALEX (SEALing performance EXperiments), se concentre sur la compréhension et l'amélioration des performances hydrauliques de divers systèmes d'étanchéité. Un scellement d'intérêt particulier dans la configuration française de stockage est appelé scellement vertical (VSS). Ces scellements sont sans aucun doute l'un des éléments clés d'un confinement sûr à long terme puisqu'ils constituent la principale barrière entre les déchets nucléaires et la biosphère, comme le montre la Fig. 7.1.

Une des configurations candidates pour ces scellements verticaux consiste en un mélange d'argiles gonflantes (Na-montmorillonite) sous la forme d'un assemblage polydispersé de pastilles très compactées et de pastilles broyées dans un état initial fortement désaturé (suscion supérieure à 100MPa). Le comportement hydromécanique de ces mélanges pastilles-poudre jouant un rôle important dans la limitation des processus de migration eau-gaz, l'IRSN a initié un ensemble d'expérimentations in-situ et en laboratoire pour mieux comprendre les évolutions structurales [13]. L'un des principaux objectifs de ces expériences était de découvrir les effets hydromécaniques de l'imbibition anisotrope sous des chargements simultanés eau-gaz. À l'échelle des pastilles, la microtomographie aux rayons X et les observations CT ont révélé le développement de motifs

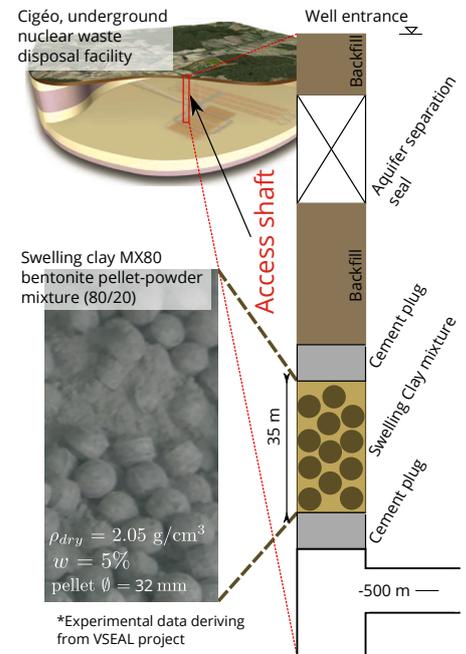


Figure 7.1: Placement of a clay pellet-powder mixture engineered barrier in an underground disposal facility, Cigéo.

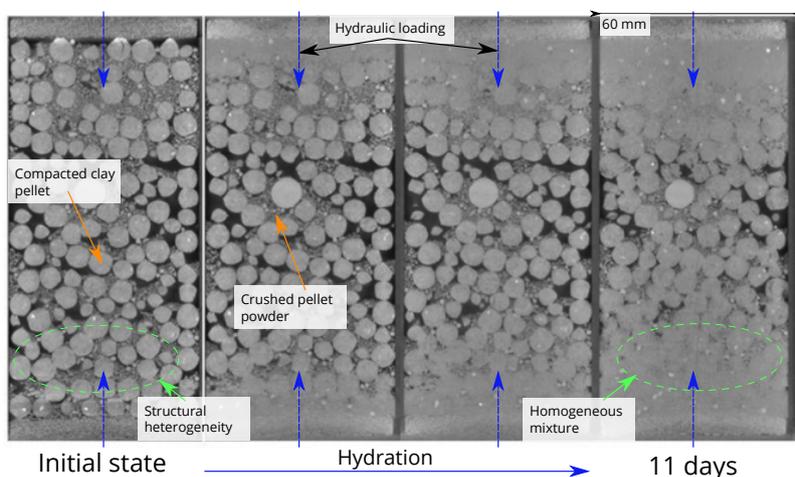


Figure 7.3: Lab-scale engineered barrier (MX80 bentonite pellet-powder mixture) [14]

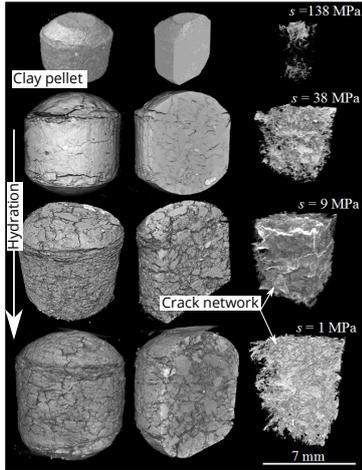


Figure 7.2: Hydration of an MX80 bentonite clay pellet, with crack network [14]

Objectif de la thèse

Modéliser les processus fondamentaux responsables du comportement complexe d'un matériau hétérogène d'argile gonflante partiellement saturé soumis à des charges hydrauliques et gazeuses anisotropes, avec une attention particulière à la compréhension de la persistance des hétérogénéités initiales sur les pressions de gonflement et la migration des gaz.

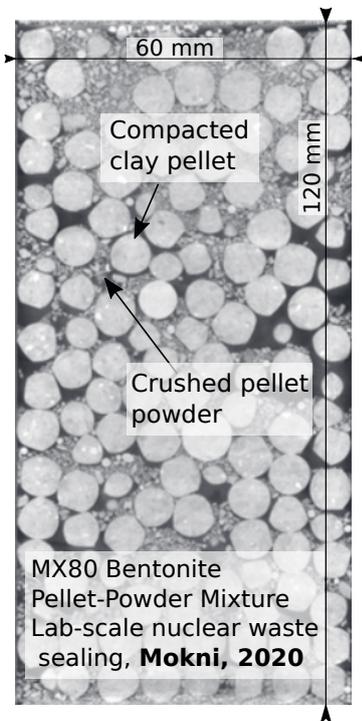


Figure 7.4: Imagerie tomodensitométrique d'un échantillon de mélange de pastilles et de poudre d'argile utilisé pour les tests d'hydratation [15].

de fissures complexes dus au transport de vapeur dans le gonflement libre Fig. 7.2 [14]. Pendant ce temps, à l'échelle du mélange pellet-poudre, d'autres expériences de laboratoire ont démontré l'effet des hétérogénéités de tassement initial sur les pressions de gonflement Fig. 7.3 [15]. Ces études ont démontré l'importance de considérer l'évolution de l'hétérogénéité initiale lors de l'évaluation de l'efficacité de scellement à long terme des mélanges granulés-poudre. Cependant, les expériences manquent de flexibilité des conditions aux limites, des limitations de durée et des analyses paramétriques statistiques à travers les échelles spatiales. Ainsi, l'objectif de la présente thèse est de fournir un ensemble de modèles numériques pouvant pallier à certaines de ces limitations.

7.2 Objectif de la thèse

Dans le cadre de cette thèse, l'IRSN et la Commission canadienne de sûreté nucléaire collaborent pour améliorer la compréhension des performances hydrauliques à long terme de ces joints de puits verticaux en mélange pastilles-poudre d'argile. Compte tenu de la complexité de la modélisation des comportements hydromécaniques des argiles gonflantes pour les barrières ouvragées, cette thèse vise à fournir une compréhension conceptuelle plus approfondie des processus micro et macroscopiques se produisant dans les mélanges argile-poudre. En particulier, les principaux objectifs de la thèse comprennent :

- ▶ Construire un modèle des processus hydromécaniques à l'*macro-échelle* pour des mélanges granulés-poudre partiellement saturés qui rend compte de l'évolution des hétérogénéités initiales *structurelles*.
- ▶ Construire un modèle des processus hydromécaniques à l'*micro-échelle* pour l'évolution des hétérogénéités *matériau* dans l'argile gonflante compactée et quantifier l'effet des réseaux de fissures microscopiques sur la perméabilité à l'eau et au gaz dans conditions partiellement saturées.
- ▶ Discrétiser et combiner les modèles à l'échelle macro et micro géométriquement et temporellement.
- ▶ Formuler et valider la mise en œuvre pratique des modèles afin que d'autres chercheurs puissent utiliser, modifier et reproduire les résultats.
- ▶ Appliquer les mises en œuvre pratiques aux conditions intital/limites pertinentes pour avoir un aperçu des évolutions de la perméabilité à long terme dans les barrières ouvragées.

7.3 Modèles conceptuels

7.3.1 Modèle hydromécanique à l'échelle macro pour les mélanges granulés-poudre

Le premier modèle (Chap. 4) est conçu pour élucider les processus hydromécaniques dans un test d'hydratation à l'échelle du laboratoire d'une barrière ouvragée composée de pastilles et de poudre (Fig. 7.4).

Le modèle est basé sur une combinaison de la 2e loi de Newton pour les interactions mécaniques entre les pastilles d'argile et la conservation de la masse pour le mouvement de l'eau entre les domaines des pastilles et de la poudre. Dans sa forme la plus simpliste :

$$\mathbf{M}\ddot{\mathbf{x}} = \mathbf{f} \quad (7.1)$$

avec $\ddot{\mathbf{x}}$, le vecteur contenant chaque accélération de particule, \mathbf{M} , la matrice diagonale des masses des particules, et \mathbf{f} , le vecteur contenant les forces totales appliquées sur les particules. Le schéma explicite aux différences finies intègre l'accélération des particules pour mettre à jour la position des particules à chaque pas de temps (voir [51] pour les détails de la mise en œuvre pratique). Les forces inter-particules, \mathbf{f}_{ij} , dépendent d'un modèle de contact, F_{ij} , tel que :

$$\frac{\partial \mathbf{f}_{ij}}{\partial t} = F_{ij}(\mathbf{x}_i, \mathbf{x}_j, \dot{\mathbf{x}}_i, \dot{\mathbf{x}}_j) \quad (7.2)$$

Comme indiqué dans la Sec. 4.2.1, la loi de contact est basée sur une loi de Hertz-Mindlin modifiée, qui tient compte de la réduction de la rigidité à saturation croissante.

Simultanément, la conservation de la masse de fluide impose que la variables de masse dans tout sous-domaine résulte d'un bilan des fluxes (q):

$$\frac{dm}{dt} = q \quad (7.3)$$

où m est la masse totale, ρ est la masse par unité de volume, Θ et q est une source matérielle ou un puits. Dans la présente implémentation, le domaine global est composé d'une collection de pastilles d'argile compactées sphériques avec de la poudre d'argile remplissant l'espace vide (Fig. 7.4). Dans tout sous-domaine (correspondant à une pastille ou un pore), l'équation 7.3 peut être convertie en son intégrale de surface, où le contour, $\partial\Theta_i$, est défini par l'interface entre les domaines de la pastille et de la poudre :

$$\frac{dm_i}{dt} = - \sum_{j=1}^m \int_{\partial\Theta_i} \rho(\mathbf{v} - \mathbf{u}) \cdot \mathbf{n} dS \quad (7.4)$$

où \mathbf{v} est la vitesse de contour, \mathbf{u} est la vitesse relative fluide-solide, et \mathbf{n} est le vecteur unité de pointage vers l'extérieur. Comme indiqué dans la Sec. 4.2.2, et montré ici dans la Fig. 7.6, ces équations sont réduites à des formes géométriques fondamentales pour estimer plus facilement les flux entre les pastilles, au sein de la poudre et entre pastilles et poudre.

D'autres considérations importantes de ce modèle, qui sont décrites dans le Chap. 4, comprennent :

- ▶ une déformation volumétrique définie empiriquement (Sec. 4.2.3)
- ▶ une nouvelle approche de la pression de gonflement partiellement saturée (Fig. 4.4)

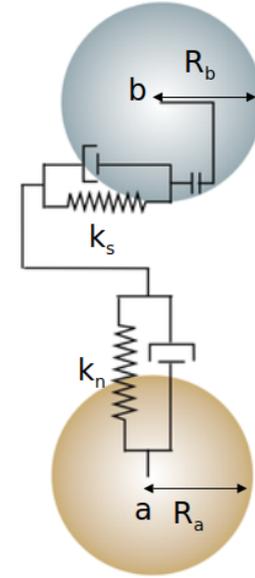


Figure 7.5: Une représentation simpliste des rigidités normales, k_n , et de cisaillement, k_s , dans la loi de Hooke utilisée pour estimer les forces entre deux particules, a et b

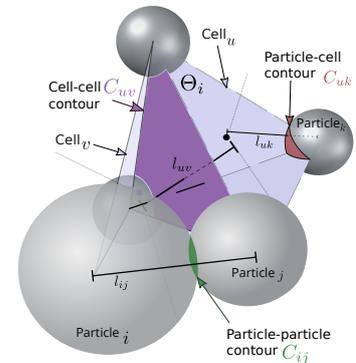


Figure 7.6: Domain discretization and geometric quantities of the mass transport equations for a clay pellet powder mix.

► la solution au problème pression-volume non linéaire (Sec. 4.3)

Un aperçu du modèle conceptuel est présenté dans la figure 7.7.

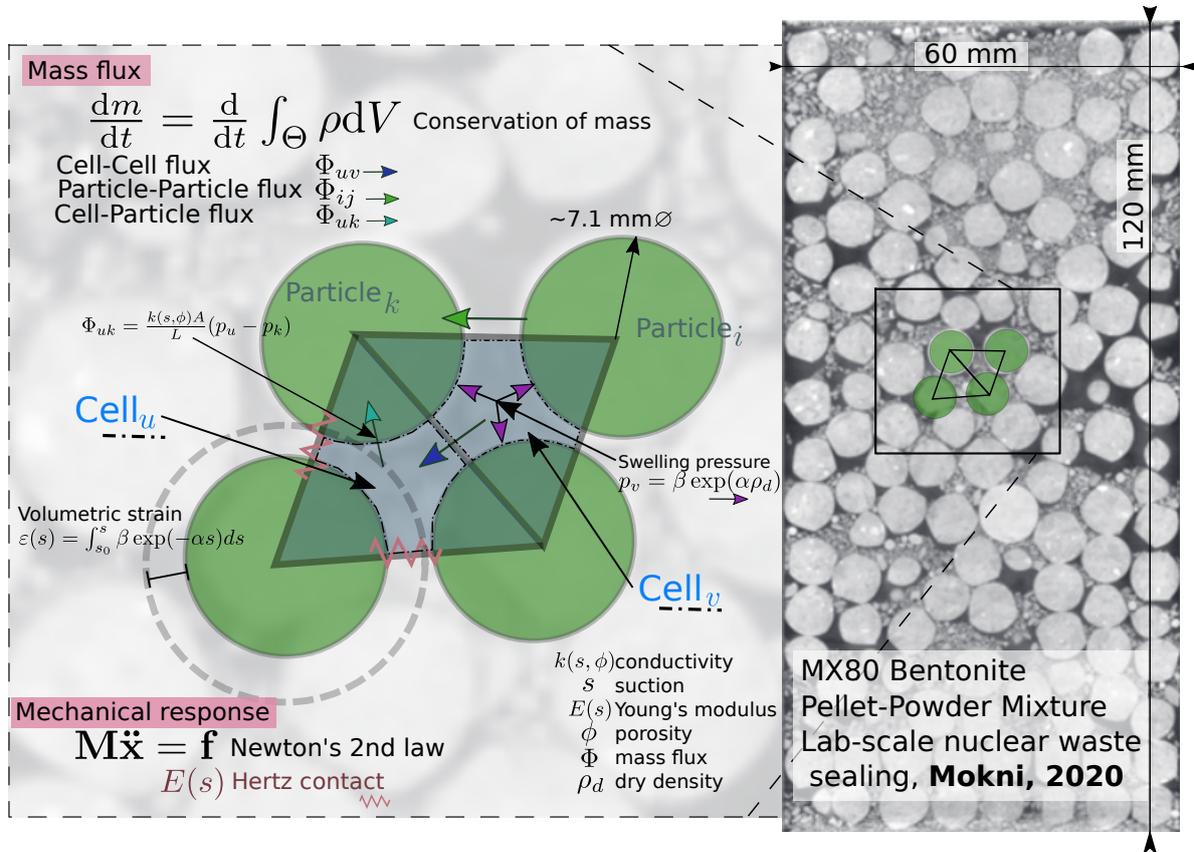


Figure 7.7: Visualisation des grandeurs clés pour le modèle de transport de masse DEM couplé.

Un élément important de ce modèle conceptuel est la connexion entre les micro et macro-échelles. Cette connexion est faite en intégrant la courbe de perméabilité calculée à l'aide du modèle à micro-échelle présenté au Chap. 3.

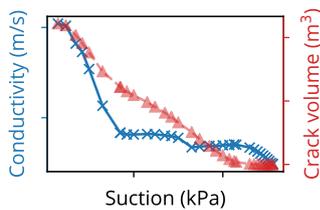


Figure 7.8: Permeability curve, Chap. 3, at the millimeter scale. This permeability is used to inform the pellet permeability, k , in the present model.

La validation du modèle a été réalisée à l'aide de la géométrie expérimentale et des conditions de chargement à partir de l'hydratation d'une barrière ouvragée à l'échelle du laboratoire (Fig. 7.9).

Résultats de la Figure 7.10 montrent que l'évolution de la pression de gonflement correspond qualitativement bien aux données expérimentales.

7.3.2 Modèle hydromécanique à l'échelle microscopique pour argile compactée hétérogène

Le modèle conceptuel à micro-échelle présenté dans cette thèse a été conçu pour refléter l'évolution des propriétés hydromécaniques résultant du développement d'hétérogénéités dans l'argile gonflante compactée lors de l'hydratation à la micro-échelle. Une visualisation de ce processus de gonflement est montrée sur la Fig. 7.2, où il est clair que le développement de fissures joue un rôle important dans le comportement hydromécanique de l'éprouvette.

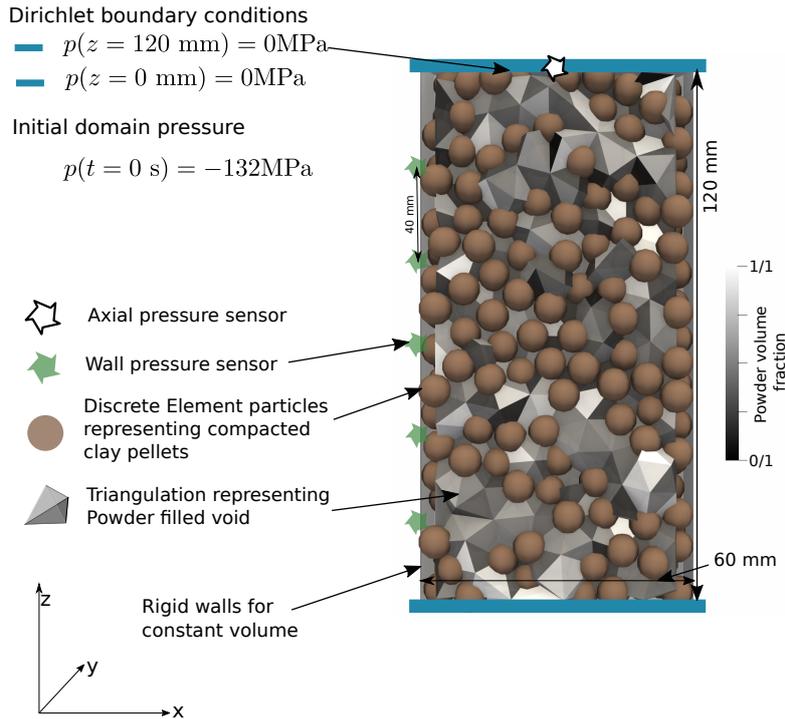


Figure 7.9: Maquette de la géométrie du modèle à l'échelle et des conditions aux limites.

Le modèle conceptuel est basé sur une combinaison de la 2e loi de Newton pour les interactions mécaniques entre les pastilles d'argile et la conservation de la masse pour le mouvement de l'eau entre les domaines des pastilles et de la poudre.

Ce modèle partage de nombreuses similitudes avec celui du Chap. 4, avec quelques différences clés :

- ▶ Une approche à double domaine où les points DEM n'affectent pas les géométries volumiques du domaine fluide
- ▶ La solution pression-volume est une formulation implicite
- ▶ La loi de contact de Hertz-Mindlin ne tient pas compte des changes de la stiffness avec les changes saturation

Sa spécificité reside dans, le modèle de fissure, illustré à la Fig. 7.11, qui est un élément central de l'évolution des perméabilités à l'eau et au gaz dans le matériau. La Section 3.3.6 décrit en détail comment la logique de fissuration est basée sur les pressions d'eau dépassant la pression d'entrée pour une fissure.

Les hétérogénéités matérielles jouent un rôle important dans le développement de ces fissures, et sont donc extraites de l'imagerie XRay CT scan, comme le montre la Fig. 7.12.

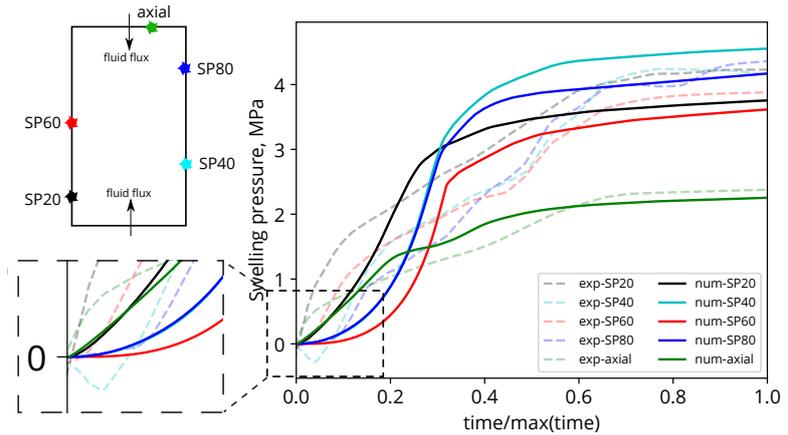


Figure 7.10: Comparaison des données de pression de gonflement pour les essais sur maquette numérique et expérimentale des tests d'hydratation.

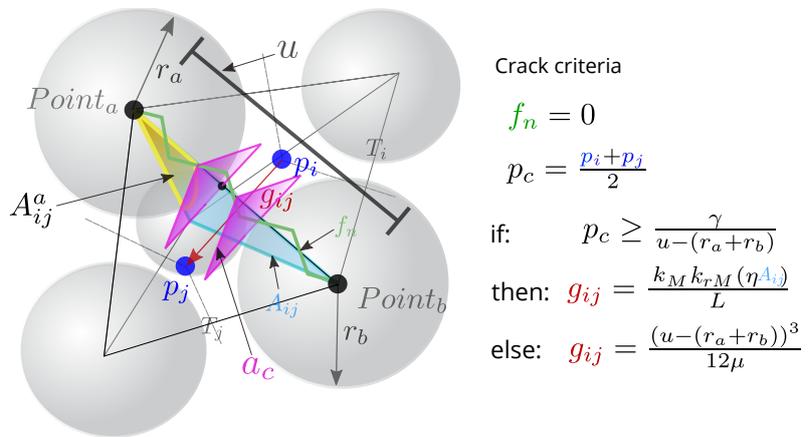


Figure 7.11: Grandeurs géométriques associées au modèle de réseau de fissures.

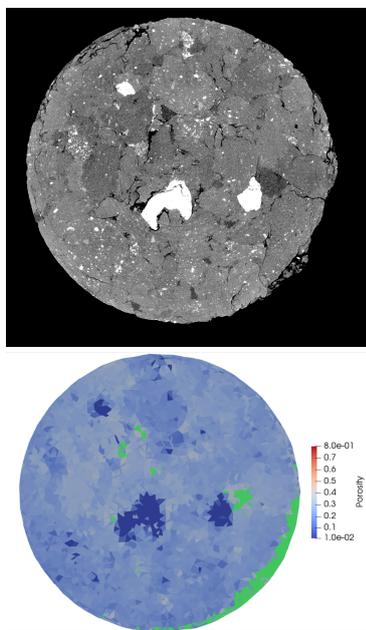


Figure 7.12: Comparaison de la distribution de la porosité initiale en haut) Valeurs de gris de l'image Xray CT, 7 mm de diamètre et taille de pixel $4,4\mu\text{m}$ en bas) valeurs de porosité du modèle avec des cellules fissurées en vert.

En fin de compte, le modèle à micro-échelle est analysé en profondeur en étudiant les éléments suivants :

- ▶ effet de la pression de confinement (Sec. 3.5.4)
- ▶ effet des fissures (Sec. 3.5.6)
- ▶ effet de l'hétérogénéité (Sec. 3.5.5)
- ▶ évolution de la distribution de porosité (Sec. 3.5.7)
- ▶ évolution de la perméabilité (qui est utilisée pour passer la micro-perméabilité au modèle macro-échelle présenté au Chap. 4) (Fig. 7.13)
- ▶ impact de la résolution du maillage (Sec. 3.5.8)
- ▶ modèles statistiques pour diverses échelles spatiales (Sec. 3.5.9)

7.3.3 Accélérer la solution du couplage hydro-mécanique

Le dernier chapitre de la thèse détaille les techniques d'accélération mise en oeuvre et les gains de performances associés pour l'intégration temporelle de problèmes poromécaniques couplés à l'aide des schémas de la méthode des éléments discrets (DEM) et des volumes finis à l'échelle des pores (PFV) dans le logiciel DEM ouvert Yade. Les formulations DEM et PFV suivent de près les méthodes présentées au Chap. 3, sauf que le couplage hydro-mécanique est simplifié en un écoulement incompressible totalement saturé. La solution à ce problème repose sur une inversion

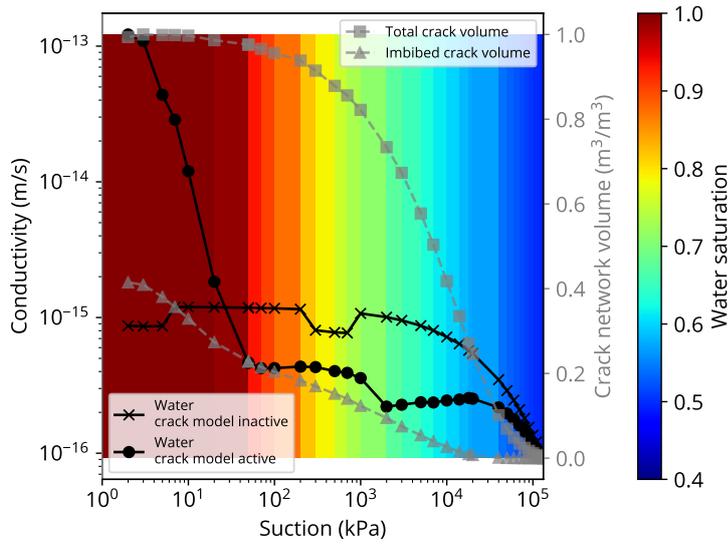


Figure 7.13: Effet du modèle de fissure sur l'évolution de la perméabilité macroscopique au cours de l'hydratation.

d'une matrice de conductivité comme suit :

$$\mathbf{Gp} = \mathbf{Ex} + \mathbf{Q} \tag{7.5}$$

\mathbf{G} clairsemé, symétrique et défini positif (illustré dans la figure 7.14), \mathbf{E} est un opérateur qui dépend linéairement de la vitesse des particules, et \mathbf{Q} est un terme source. La relation pressions-vitesses instantanées s'écrit finalement : Par conséquent, la décomposition de Cholesky est utilisée pour la décomposition de \mathbf{G} en une matrice triangulaire inférieure multipliée par sa transposée (\mathbf{LL}^T). La matrice décomposée, c'est-à-dire le facteur, peut être utilisée pour résoudre \mathbf{p} en utilisant d'abord la substitution avant suivie d'une substitution arrière :

$$\mathbf{Ly} = \mathbf{x} \tag{7.6}$$

$$\mathbf{L}^T \mathbf{p} = \mathbf{y} \tag{7.7}$$

évitant ainsi l'inversion prohibitive de \mathbf{G} pour la solution de \mathbf{p} . Les forces de traînée sur les particules (\mathbf{f}_D) sont obtenues après multiplication du vecteur pression par une matrice \mathbf{F} dont les composantes reflètent la surface projetée :

$$\mathbf{f}_D = \mathbf{Fp} \tag{7.8}$$

Comme discuté et quantifié dans le chapitre, le coût de calcul du couplage poroélastique DEM+PFV n'est pas négligeable. Cependant, l'introduction de la poroélasticité peut aggraver le ralentissement du calcul en réduisant également le pas de temps stable maximal. Comme démontré dans 5.7, dès que les effets de rigidité DEM typiques (la période naturelle d'un système de masse de ressort) deviennent négligeables par rapport aux effets visqueux (forces de traînée fluide agissant comme des amortisseurs),

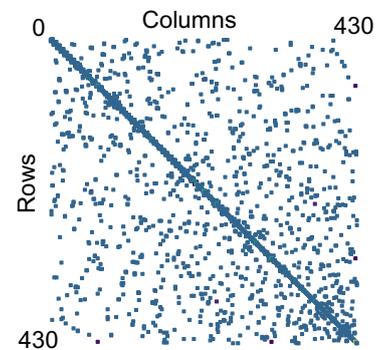


Figure 7.14: Exemple d'une matrice de conductivité définie positive, symétrique, à bandes et clairsemée à 430 degrés de liberté (\mathbf{G})

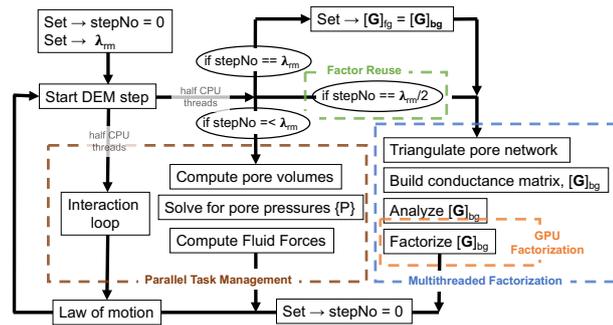


Figure 7.15: Présentation de l'algorithme accéléré Yade DEM+PFV.

le pas de temps maximum dépend du rayon spectral d'une matrice de résistance visqueuse. Il n'est pas rare qu'une simulation poroélastique d'un matériau granulaire fonctionne à un pas de temps inférieur d'un ordre de grandeur à son homologue sec. D'où le besoin de techniques d'accélération mis en évidence ici.

Ainsi, le reste du chapitre traite du traitement de cette limitation à l'aide de quatre techniques d'accélération illustrées dans la figure 7.15 :

- ▶ réutilisation de G pour plusieurs pas de temps et plusieurs solutions de pression
- ▶ factorisation accélérée par GPU de G
- ▶ factorisant G en arrière-plan tout en résolvant activement les pressions
- ▶ calculant les pressions en parallèle avec le reste de l'algorithme DEM

7.4 Conclusions

Au final, ce projet de thèse était une approche unique pour comprendre les aspects supposés **les plus importants** des argiles gonflantes partiellement saturées dans les barrières ouvragées. C'est-à-dire que nous avons explicitement admis que nous n'étudierions pas **tous** les aspects de l'argile partiellement saturée. Un si simple aveu de simplicité (si l'on peut appeler l'étude des comportements hydromécaniques dans des argiles hétérogènes partiellement saturées, **simple**) a conduit à une analyse multi-échelle fructueuse. Cela signifie que le projet de thèse n'a jamais été entravé par des complexités supplémentaires telles que les surfaces de charge, l'érouissage, les coefficients de compressibilité, etc. Cet aveu de simplicité signifie que le projet de thèse se concentre principalement sur d'autres caractéristiques hydromécaniques telles que la résolution des équations pression-volume dans les mélanges pellet-poudre (Chap. 3), la modélisation des développements de fissures à partir d'hétérogénéités (Chap. 3), et la compréhension/amélioration des limites pratiques de nos outils (Chap. 5). En fin de compte, la recherche décrite dans cette thèse a produit des conclusions applicables à la conception de scellement pour déchets nucléaires ainsi que des méthodes/outils scientifiques réutilisables/modifiables. Pour être précis, le premier modèle/outil conceptuel s'est concentré sur la compréhension des interactions complexes entre

l'hétérogénéité initiale, la rétention d'eau, le gonflement volumétrique et les pressions de gonflement dans les mélanges granulés-poudre. Le deuxième modèle conceptuel s'est concentré sur la compréhension du développement des réseaux de fissures au sein des pastilles d'argile compactées individuelles et leur effet sur l'évolution de la perméabilité au cours de l'hydratation. De plus, la méthode elle-même a été analysée et accélérée pour améliorer les contraintes de taille de domaine.

Echelle des granulés :

- ▶ Les réseaux de fissures jouent un rôle important dans l'évolution de la perméabilité au gaz et à l'eau
- ▶ Les réseaux de fissures se développent au cours de l'hydratation en raison de la présence initiale d'hétérogénéités
- ▶ La pression de confinement réduit la perméabilité au gaz de plus d'un ordre de grandeur

Échelle du mélange granulés-poudre :

- ▶ Une discrétisation DEM résout le comportement hydromécanique des mélanges granulés-poudre de manière extrêmement efficace, sans avoir besoin d'étalonnage.
- ▶ La résolution couplée pression-volume est essentielle pour la précision des résultats
- ▶ La pression de gonflement dépend des distributions hétérogènes initiales de granulés-poudre

Accélération:

- ▶ La réutilisation de factorisation $L^T L$ de la matrice de conductivité a le plus grand impact sur les performances
- ▶ La factorisation de la matrice de conductivité mise à jour en parallèle avec la simulation active permet une fréquence accrue de refactorisation sans effort de calcul supplémentaire.
- ▶ L'accélération GPU rend la factorisation de la matrice de conductivité 75% plus rapide pour les gros problèmes.

APPENDIX A: SUPPORTING MATERIALS

.1 Macro-scale pellet-powder mixture python script

```

# -*- encoding=utf-8 -*-
#*****
# Copyright (C) 2019 by Robert Caulk *
# rob.caulk@gmail.com *
# *
# This program is free software; it is licensed under the terms of the *
# GNU General Public License v2 or later. See file LICENSE for details. *
#*****/
#
# Example script demonstrating the use of the TransportEngine with application
# to a clay pellet-powder mixture test designed to track macroscopic
# permeability and swelling pressures

from yade import pack, ymport, plot, utils, export, timing
import numpy as np
import time
import shutil

def loadFacets(fName,**kw):
    with open(fName) as f: # read file
        lines = f.readlines()
        ret = []
        for line in lines: # for each line, reconstruc the facet
            fs = [float(v) for v in line.split()] # read floats
            v1,v2,v3 = [Vector3(fs[i],fs[i+1],fs[i+2]) for i in (0,3,6)] # make 3 Vector3
            f = facet((v1,v2,v3),**kw) # create facet
            ret.append(f)
        return ret

timeStr = time.strftime('%m-%d-%Y')

young=800e6

rMean = 0.00599 #0.00599 #0.00473 #0.00365
iterper = 100
cylHt,cylRd=0.12,0.03
timeStep = 0.1
meshUpdate = 100
dynamic = True
packingName='cylinderSpecimen_'+str(rMean)+'.spheres'
density=2600e10

mn,mx=Vector3(-.035,-.035,-.065),Vector3(0.035,0.035,0.065)
#mn,mx=Vector3(-.05,-.05,-.05),Vector3(0.05,0.05,0.05)

identifier =
↳ '-examplescript_'+"{0:.1e}".format(timeStep)+'_rMean'+"{0:.2e}".format(rMean)

if not os.path.exists('VTK'+timeStr+identifier):

```

```

        os.mkdir('VTK'+timeStr+identifier)
else:
    shutil.rmtree('VTK'+timeStr+identifier)
    os.mkdir('VTK'+timeStr+identifier)

if not os.path.exists('txt'+timeStr+identifier):
    os.mkdir('txt'+timeStr+identifier)
else:
    shutil.rmtree('txt'+timeStr+identifier)
    os.mkdir('txt'+timeStr+identifier)

# copy input script to simulation txt folder for record sake
shutil.copyfile(sys.argv[0], 'txt'+timeStr+identifier+'/' + sys.argv[0])
shutil.copyfile(packingName, 'txt'+timeStr+identifier+'/' + packingName)

0.materials.append(FrictMat(young=young*100,poisson=0.3,frictionAngle=0,density=density,label='walls'))
0.materials.append(FrictMat(young=young,poisson=0.3,frictionAngle=radians(30),density=density,label='sp

walls=aabbWalls([mn,mx],thickness=0,material='walls')
wallIds=0.bodies.append(walls)

#sp=pack.SpherePack()
#sp.makeCloud(mn,mx,rMean=0.0035,rRelFuzz=0.01,num=200,seed=1)
#sp.toSimulation(color=(0.752, 0.752, 0.752),material='spheres')

sp =
↳ 0.bodies.append(ymport.textExt('cylinderSpecimen_noLockedInPressure_'+str(rMean)+' .spheres',
↳ 'x_y_z_r',color=(0.1,0.1,0.9), material='spheres'))

0.materials.append(FrictMat(young=young*100,poisson=0.3,frictionAngle=0,density=density,label='walls2'))

facets = loadFacets('cylinder_'+str(rMean)+' .facets')
containerIds = 0.bodies.append(facets)
axialFacets = []
axialFacets =[i for i in containerIds if 0.bodies[i].state.pos[2] == cylHt/2]

0.engines=[
    ForceResetter(),

    ↳ InsertionSortCollider([Bo1_Sphere_Aabb(aabbEnlargeFactor=1,label='is2aabb'),Bo1_Facet_Aabb()]),
    InteractionLoop(

        ↳ [Ig2_Sphere_Sphere_ScGeom(interactionDetectionFactor=1,label='ss2sc'),Ig2_Facet_Sphere_ScGeom()],
        [Ip2_FrictMat_FrictMat_MindlinPhys()],
        [Law2_ScGeom_MindlinPhys_Mindlin()],label="iloop"
    ),

    ↳ #GlobalStiffnessTimeStepper(active=1,timeStepUpdateInterval=1000,timestepSafetyCoefficient=0.5),
    #triax,
    TransportFlowEngine(dead=1,label="flow",multithread=False),

```

```

↪ VTKRecorder(iterPeriod=iterper, fileName='VTK'+timeStr+identifier+'/spheres-', recorders=['facets',
NewtonIntegrator(damping=0.5)
]

0.step()
ss2sc.interactionDetectionFactor=-1
is2aabb.aabbEnlargeFactor=-1

for b in 0.bodies:
    if isinstance(b.shape, Sphere):
        b.dynamic=dynamic
    if isinstance(b.shape, Facet):
        b.dynamic=False

pZero = -132e6
flow.debug=False
# add flow
flow.permeabilityMap = False
flow.pZero = pZero
flow.meshUpdateInterval=meshUpdate
flow.defTolerance=-1
flow.fluidBulkModulus=2.2e9
flow.useSolver=4
#flow.permeabilityFactor=-4e-8
flow.viscosity= 0.001
flow.decoupleForces = False
flow.cellP0 = pZero
flow.cellBndCondIsDirichlet=[0,0,0,0,1,1]
flow.cellTransferBndCondValue=[0,0,0,0,-1e6,-1e6]

flow.particleTransferBndCondValue=[0,0,0,0,0,0]
flow.thermalEngine=True
flow.particleBndCondIsDirichlet=[0,0,0,0,0,0]
flow.particleP0=pZero
flow.particleTransfer=True
flow.particleCellTransfer=True
flow.cellCellTransfer=True
flow.setParticlePorosity=True
flow.alpham=0.024e-6 #2.1e-8
flow.betam=0.016e-6
# flow.advection= True
conductivityMultiplier = 1 #1e10
flow.particleConductivity = 1e-10 # only used if flow.conductivityTextFile not provided
flow.cellConductivity = 1e-20 * conductivityMultiplier #1e-09
flow.conductivityFactor = conductivityMultiplier
flow.useEquivalentCompressibility = False
flow.boundaryUseMaxMin = [0,0,0,0,0,0]
flow.conductivityTextFile = '../upscale_data/pellet_permeability_data_050621.csv' #
↪ upscaling the permeability data from microscale model

flow.minPoroClamp = 0.1
flow.maxPoroClamp = 0.6

```

```

flow.lambdaWeibullShape = 4 # volume fraction heterogeneity distribution
flow.minimumPorosity=0

flow.homogeneousPorosity = False
flow.meanInitialPorosity = 0.2
flow.freezePorosity = False

flow.swellingPressure=True
flow.cellExpansion=True
flow.particleExpansion=True
flow.homogeneousMixtureVolumes = False # random distribution of powder volumes
flow.youngsModulusChange = True # youngs is function of suction
flow.minYoung, flow.maxYoung = 100e6, 1000e6 # cap the youngs function
flow.letTransportRunFlowForceUpdates = True

flow.useNewtonRaphson = True # use newtonraphson to solve p-v
flow.errorThreshold = 1e-3
flow.lambda_relax = 0.1
flow.relaxFactor = 0.1
flow.hfactor = 0.0001
flow.mu_start = 0.1
flow.constrain_newton = True
flow.newtonUseLdlt = False

flow.debug_precise=False
flow.onlyUpdateFacetSurfaces=True
flow.maxSuctionConstraint = 138e6
flow.minSuctionConstraint = 1e6
flow.maxVoidVolume = 4/3 * np.pi * rMean**3
flow.minVoidVolume = flow.maxVoidVolume * 0.0001
#flow.waitForUnbalanced=True

flow.dead=0

timing.reset()
#0.timingEnabled=1
0.dt=timeStep
0.dynDt=False

def checkTimeStep():
    dt = utils.PWaveTimeStep()
    if dt < 0.dt: 0.dt = dt

## function for collecting swelling pressures on facets
def getPressure():
    numSplits = 6 # split the height of the cell into N equal segments for measuring
    ↪ pressure
    cylHt,cylRd=0.12,0.03#+rMean*2
    htMax, htMin = cylHt/2, -cylHt/2
    splits = np.linspace(htMin, htMax, numSplits)
    totalPressure = np.zeros(len(splits)-1)
    for j in range(1, len(splits)):
        bodyId,facetIs,totalForce = 0,0,0

```

```

    for i in 0.interactions:
        if isinstance(0.bodies[i.id1].shape, Facet) or isinstance
        ↪ (0.bodies[i.id2].shape, Facet):
            if isinstance(0.bodies[i.id1].shape, Sphere): bodyId,
            ↪ facetId = i.id1, i.id2
            if isinstance(0.bodies[i.id2].shape, Sphere): bodyId,
            ↪ facetId = i.id2, i.id1
            if splits[j-1] < i.geom.contactPoint[2] < splits[j]:
                n = 0.bodies[facetId].shape.normal
                f = -i.phys.normalForce - i.phys.shearForce
                fxn = np.dot(f, n)
                totalForce += fxn
    area = (splits[j] - splits[j-1]) * np.pi * cylRd
    totalPressure[j-1] = totalForce/area

axialArea,axialForce = 0,0
for i in axialFacets:
    n = 0.bodies[i].shape.normal
    a = 0.bodies[i].shape.area
    axialForce += np.dot(0.forces.f(i, sync=True), n)
    axialArea += a

axialPressure = axialForce/axialArea

return totalPressure, axialPressure

## function for getting bodies given coordinates
def bodyByPos(x,y,z):
    cBody = 0.bodies[1]
    cDist = Vector3(100,100,100)
    for b in 0.bodies:
        if isinstance(b.shape, Sphere):
            dist = b.state.pos - Vector3(x,y,z)
            if np.linalg.norm(dist) < np.linalg.norm(cDist):
                cDist = dist
                cBody = b
    return cBody

bodyOfInterest = bodyByPos(0.0,0.0,0.0)

from yade import plot

## function for collecting data during simulation
def history():
    sensors, axial = getPressure()
    macro_perm = emulatePermeability()
    plot.addData(
        b_pot = flow.getVertexPotential(bodyOfInterest.id),
        c_pot = flow.getCellPotential((0.0,0.0,0.0)),
        dt = 0.dt,
        ps1 = sensors[0],
        ps2 = sensors[1],
        ps3 = sensors[2],

```

```

        ps4 = sensors[3],
        ps5 = sensors[4],
        axial = axial,
        unbal = unbalancedForce(),
    t=0.time,
    i = 0.iter,
    k = macro_perm
)
plot.plot(noShow=True).savefig('txt'+timeStr+identifier+'/plot.pdf')

↪ plot.saveDataTxt('txt'+timeStr+identifier+'/data.txt',vars=('i','ps1','ps2','ps3','ps4','ps5','axi

0.engines=0.engines+[PyRunner(iterPeriod=iterper,command='history()',label='recorder')]
0.engines=0.engines+[PyRunner(iterPeriod=200,dead=1,command='checkTimeStep()',label='timestep')]

def pressureField():
    flow.saveVertexInfoVTK('VTK'+timeStr+identifier+'/vertex')
    flow.saveCellInfoVTK('VTK'+timeStr+identifier+'/cell', True)

0.engines=0.engines+[PyRunner(iterPeriod=iterper+int(iterper/2),command='pressureField()',label='pressu

plot.plots={'i':(('c_pot','b_pot'),'i ':'unbal',' i':('ps1','ps3','ps5','axial'),' i
↪ ':'k')}}

## emulate permeability check without disturbing the hydration
def emulatePermeability():

    axis = 2
    currentSuction = -flow.getCellPotential((0.0,0.0,0.0))
    file_name = 'suction_"{0:.1e}".format(currentSuction)+'_axis_'+str(axis)
    if not os.path.exists('VTK'+timeStr+identifier+'/' +file_name):
        os.mkdir('VTK'+timeStr+identifier+'/' +file_name)
    else:
        shutil.rmtree('VTK'+timeStr+identifier+'/' +file_name)
        os.mkdir('VTK'+timeStr+identifier+'/' +file_name)
    flow.freezeStates = True
    #flow.requestTriangulation = False
    flow.meshUpdateInterval=-1

    for b in 0.bodies:
        if isinstance(b.shape, Sphere):
            b.dynamic=False
            #b.state.vel=(0,0,0)

    dim=utils.aabbExtrema()
    xinf=dim[0][0]
    xsup=dim[1][0]
    X=xsup-xinf
    yinf=dim[0][1]
    ysup=dim[1][1]
    Y=ysup-yinf
    zinf=dim[0][2]
    zsup=dim[1][2]

```

```

Z=zsuf-zinf
area = [(ysuf-yinf)*(zsuf-zinf), (zsuf-zinf)*(xsuf-xinf),
↪ (ysuf-yinf)*(xsuf-xinf)]
L = [(xsuf-xinf), (ysuf-yinf), (zsuf-zinf)]
kstd = 0

# Setting the pressure gradient along the boundaries
delP = currentSuction*0.1
axis_up = [xsuf,ysuf,zsuf]
down_pressure = -currentSuction-delP/2
up_pressure = -currentSuction+delP/2
slope = (up_pressure - down_pressure)/L[axis]
yintercept = up_pressure-(slope*axis_up[axis])

print("specimen length", L)

for i in range(0,flow.nCells()):
    coords = flow.getCellCenter(i)
    pressure_imposed = flow.getCellFictitious(i)
    if not pressure_imposed: continue
    zcoord = coords[axis]
    pressure = zcoord*slope+yintercept #currentSuction
    flow.imposePressureId(i,pressure) #(coords[0],coords[1],coords[2])
    # print('imposing', pressure, 'on', i)

flow.emulateAction()

# visualization
flow.saveCellInfoVTK('VTK'+timeStr+identifier+'/' +file_name,withBoundaries=True)
#flow.savePermeabilityNetworkVTK(scale_folder+'VTK'+timeStr+identifier+'/' +file_name+'/' +cellConn)
#print("saved the vtk")

## FLUID PERM
size_reduce = 0.1
xlims,ylims,zlims =
↪ [xinf+L[axis]*size_reduce,xsuf-L[axis]*size_reduce],[yinf+L[axis]*size_reduce,ysuf-L[axis]*
velocitySum = 0
saturationSum = 0
poroSum = 0
numPts = 0
nans = 0
cellsHit = []
totalVelocity = np.array([0,0,0])
totalVolume = 0
v = np.array([0,0,0])

cellsInside = []
for i in range(0,flow.nCells()):
    coords = flow.getCellCenter(i)
    if flow.getCellFictitious(i): continue
    if xlims[0]<=coords[0]<=xlims[1] and ylims[0]<=coords[1]<=ylims[1] and
↪ zlims[0]<=coords[2]<=zlims[1]:
        cellsInside.append(i)

```

```

        velocityVector =
        ↪ np.array(flow.getCellVelocity_transport((coords[0],coords[1],coords[2])))
        cellVol =
        ↪ flow.getCellVolume_transport((coords[0],coords[1],coords[2]))
        v = v + cellVol*velocityVector
        totalVolume += cellVol
        #numPts += 1
    print('cells used for volume perm',len(cellsInside))

    q = np.linalg.norm(v)/totalVolume
    kv = q*flow.viscosity/(slope)
    print('perm by volume', kv)

    # rest the bcs and continue on
    flow.clearImposedPressures_transport()
    flow.freezeStates = False
    flow.cellBndCondIsDirichlet=[0,0,0,0,1,1]
    flow.cellTransferBndCondValue=[0,0,0,0,-1e6,-1e6]
    flow.updateTriangulation = False
    flow.requestTriangulation = False
    flow.updateBCs = True
    flow.meshUpdateInterval = meshUpdate

    for b in O.bodies:
        if isinstance(b.shape, Sphere):
            b.dynamic=True
            #b.state.vel=(0,0,0)

    return kv

from yade import qt
yade.qt.Controller(), yade.qt.View()

#O.run(1,1)

## let the initial suctions pull particles together and find stable state before
## initiating simulation. All states frozen during this, and volumes are factored
## down as cell volumes are reduced
def resolveUnbalanced():
    threshold = 0.01
    unb = unbalancedForce()
    if unb > threshold:
        flow.waitForUnbalanced=True
        #flow.requestTriangulation = True
        flow.freezeStates = True
        # flow.meshUpdateInterval=1000
        #O.run(400,1)
        #avgSuction = flow.getAverageSuction()
        print('unbalancedForce', unb)
    if (unb <= threshold): # and centerBody.state.suction!=0:
        flow.freezeStates = False
        flow.waitForUnbalanced=False
        flow.requestTriangulation = False

```

```

        flow.meshUpdateInterval=10000
        print('finished unb')
        resolve.dead=1
    return

0.engines=0.engines+[PyRunner(iterPeriod=1,command='resolveUnbalanced()',label='resolve')]

#0.engines=0.engines+[PyRunner(iterPeriod=iterper,command='emulatePermeability()',label='perm')]

print("avg particle diffusion", flow.avgParticleDiffusion)
print("avg cell diffusion", flow.avgCellDiffusion)

```

.2 Micro-scale compacted clay script

```
# -*- coding: utf-8 -*-
```

```
'''
```

*This script is released in conjunction with Caulk, R., Mokni, N., Chareyre, B. (2021)
↳ Modeling the transience of partial saturation and cracks in heterogeneous swelling
↳ clays using the Discrete Element and Finite Volume Methods. Submitted to Granular
↳ Matter.*

*Install yade on Ubuntu:
sudo apt-get install yade*

*Execute this script with:
yade partiallySaturatedExampleScript.py*

*Ensure you also have downloaded and stored the following files in the same directory:
porosityGrid_150000grid.txt
cube_0.01-to-1500_7mmedge*

Copyright Robert Caulk rob.caulk@gmail.com

```
'''
```

```

from yade import pack, ymport, export
from yade import timing, plot
import numpy as np
import shutil

import sys
import itertools
import importlib.util
tick = time.time()
##### Set general params, built file structure
↳ #####
young=100e8
density = 2600
iterper=400 # output file write frequency
intRadius=1.0
targetSuction = 1000

```

```

pZero = 132e6 # initial suction
captureStdout=False # save terminal output to file
shape = 'cube' # options cube or pellet
batch=True # activate if running sweeps
delP = 1000. # pressure gradient for permeability estimate
dynamics=True
clusterScript = False # set true if running large batches on clusters
if not clusterScript: import psutil

if clusterScript:
    grid_scale = float(os.environ["GRIDSCALE"])
    crackActive = int(os.environ["CRACKACTIVE"])
    confined = int(os.environ["CONFINED"])
    oarjobid = int(os.environ["OAR_JOBID"])
    numCores = int(os.environ["NUMCORES"])
    res = str(os.environ["RESOLUTION"])
    confining_pressure = float(os.environ["CONFININGPRESSURE"])
else:
    grid_scale = 1.5 #
    crackActive = 1 #
    confined = 0 #
    numCores = 4 # must match number of cores on system
    res = '1500' #
    translation_vector = [0,0,0]
    confining_pressure = 0

waterviscosity = 1e-6 * 1e-5
airViscosity = 1e-8 * 1e-5
kfactor = -4e-8
beta_eps = 2.0e-8
random_region = False
partial_confine = False

ortho = False
hydraulic_equilibrium = False
hetero_poro = True

grid_path = 'porosityGrid_150000grid.txt'
timeStep =4e-9

if shape == 'cube':
    if res == '15':
        packingName = 'cube_0.0001-to-15_7mmedge.spheres'
    elif res == '150':
        packingName = 'cube_0.001-to-150_7mmedge.spheres'
    elif res == '1500':
        packingName = 'cube_0.01-to-1500_7mmedge.spheres'
    elif res == '15000':
        packingName = 'cube_0.1-to-15000_7mmedge.spheres'
    elif res == '37500':
        packingName = 'cube_0.25-to-37500_7mmedge.spheres'
    elif res == '75000':

```

```

        packingName = 'cube_0.5-to-75000_7mmedge.spheres'
    elif res == '150000':
        packingName = 'cube_1-to-150000_7mmedge.spheres'

if ortho: packingName = '7cmEdge_ortho_1600.spheres'

# unique identifier, containing some key meta data
scale_folder = 'n_particles_'+res+'/'
if not os.path.exists(scale_folder): os.mkdir(scale_folder)
batchId =
    ↪ "{0:.1e}".format(kfactor)+'beta_'+"{0:.1e}".format(beta_eps)+'crack_'+str(crackActive)
identifier=batchId+'timestep-'+"{0:.1e}".format(timeStep)
if clusterScript: identifier = identifier+'-oarID-'+str(oarjobid)
timeStr = time.strftime('%m-%d-%Y')
# setup the simulation folder structure
if not os.path.exists(scale_folder+'VTK'+timeStr+identifier):
    os.mkdir(scale_folder+'VTK'+timeStr+identifier)
else:
    shutil.rmtree(scale_folder+'VTK'+timeStr+identifier)
    os.mkdir(scale_folder+'VTK'+timeStr+identifier)

if not os.path.exists(scale_folder+'txt'+timeStr+identifier):
    os.mkdir(scale_folder+'txt'+timeStr+identifier)
else:
    shutil.rmtree(scale_folder+'txt'+timeStr+identifier)
    os.mkdir(scale_folder+'txt'+timeStr+identifier)

# copy input script to simulation txt folder for record sake
shutil.copyfile(sys.argv[0],scale_folder+'txt'+timeStr+identifier+'/'+sys.argv[0])
shutil.copyfile(packingName,scale_folder+'txt'+timeStr+identifier+'/'+packingName)

# setting up a saved state file for faster sweeps (avoiding suction equilibration)
savedStateName =
    ↪ 'pellet_'+str(int(pZero/1e6))+'MPa-suction_'+str(int(young/1e6))+'MPastiff_'+packingName+'.yade.bz2'
savedState = False #os.path.exists(savedStateName) # check if the file exists already and
    ↪ use it

# function for finding a body given arbitrary coordinates
def bodyByPos(x,y,z):
    cBody = 0.bodies[1]
    cDist = Vector3(100,100,100)
    for b in 0.bodies:
        if isinstance(b.shape, Sphere):
            dist = b.state.pos - Vector3(x,y,z)
            if np.linalg.norm(dist) < np.linalg.norm(cDist):
                cDist = dist
                cBody = b
    #print 'found closest body ', cBody.id, ' at ', cBody.state.pos
    return cBody

def getMnMx():
    dim=utils.aabbExtrema()

```

```

xinf=dim[0][0]
xsup=dim[1][0]
X=xsup-xinf
yinf=dim[0][1]
ysup=dim[1][1]
Y=ysup-yinf
zinf=dim[0][2]
zsup=dim[1][2]
Z=zsup-zinf
mn,mx=Vector3(xinf, yinf,zinf),Vector3(xsup, ysup, zsup)
return mn,mx

```

```
if captureStdout:
```

```

    stdout_fileno = sys.stdout.fileno()
    stdout_save = os.dup(stdout_fileno)
    stdout_pipe = os.pipe()
    os.dup2(stdout_pipe[1], stdout_fileno)
    os.close(stdout_pipe[1])

```

```
##### Setup engines and material #####
```

```
print('Saved state?',savedState,identifier)
```

```
if not(savedState) or clusterScript: # cluster doesnt like binary access issues
```

```

sp = 0.bodies.append(ympart.textExt(packingName, 'x_y_z_r'))
mn,mx = getMnMx()
centerBody = bodyByPos((mx[0]-mn[0])/2,(mx[1]-mn[1])/2,(mx[2]-mn[2])/2)
body_tracked = bodyByPos((mx[0]-mn[0])/4,(mx[1]-mn[1])/2,(mx[2]-mn[2])/2)
tracked_id = body_tracked.id

```

```
volume_original = (mx[0]-mn[0]) * (mx[1]-mn[1]) * (mx[2]-mn[2])
```

```
Lx, Ly, Lz = mx[0]-mn[0], mx[1]-mn[1], mx[2]-mn[2]
```

```
if random_region:
```

```

    trans_lim_up = np.array([0.003-Lx/2,0.003-Ly/2,0.003-Lz/2])
    trans_lim_down = np.array([-0.003+Lx/2,-0.003+Ly/2,-0.003+Lz/2])

```

```
translation_vector = np.zeros(3)
```

```

translation_vector[0] =
    ↪ np.random.uniform(trans_lim_down[0],trans_lim_up[0])
translation_vector[1] =
    ↪ np.random.uniform(trans_lim_down[1],trans_lim_up[1])
translation_vector[2] =
    ↪ np.random.uniform(trans_lim_down[2],trans_lim_up[2])

```

```
else: translation_vector = np.zeros(3)
```

```
# write a meta data file
```

```
f = open(scale_folder+'txt'+timeStr+identifier+'/meta_data.txt','w')
```

```
f.write('resolution '+res+'\n')
```

```
f.write('grid_scale %g\n' % (grid_scale))
```

```
f.write('time_step %g\n' % (timeStep))
```

```

f.write('translation %g %g %g\n' % (translation_vector[0], translation_vector[1],
↳ translation_vector[2]))
f.write('confiningpress %g\n' % (confining_pressure))
f.write('volume %g\n' % (volume_original))
f.write('crack %g\n' % (crackActive))
f.close()

O.reset()

# materials
O.materials.append(PartialSatMat(density=density,young=young,poisson=.3,frictionAngle=radians(1
O.materials.append(PartialSatMat(density=0,young=8.8e13,poisson=.8,frictionAngle=0.,label='wall

# interaction loop
iLoop = InteractionLoop(
    [Ig2_Sphere_Sphere_ScGeom6D(interactionDetectionFactor=intRadius,
↳ label='SSgeom'),Ig2_Box_Sphere_ScGeom6D()],
    [Ip2_PartialSatMat_PartialSatMat_MindlinPhys(label="hertzIp")],
    [Law2_ScGeom_MindlinPhys_Mindlin(includeAdhesion=False,label='hertzLaw')],label="iloop"
)

walls=aabbWalls([mn,mx],thickness=0,material='walls')
wallIds=O.bodies.append(walls)

sp = O.bodies.append(ympart.textExt(packingName, 'x_y_z_r',color=(0,0.2,0.7),
↳ material='spheres'))

# engine list
O.engines=[
    ForceResetter(),
    InsertionSortCollider([Bo1_Sphere_Aabb(aabbEnlargeFactor=intRadius,
↳ label='Saabb'),Bo1_Box_Aabb()]),
    iLoop,
    PartialSatClayEngine(dead=1,label="flow",multithread=multithread),
    GlobalStiffnessTimeStepper(active=1,timeStepUpdateInterval=100,timestepSafetyCoefficient
    TriaxialStressController(thickness=0,stressMask=7,internalCompaction=False,label='triax
    VTKRecorder(iterPeriod=iterper,fileName=scale_folder+'VTK'+timeStr+identifier+'/spheres
    #CentralGravityEngine(accel=1, label='centralGrav',
↳ centralBody=centerBody.id ,dead=0),
    NewtonIntegrator(gravity=(0,0,0), damping=0.4,label='newton'),
    DomainLimiter(iterPeriod=1,dead=1,label='domainlimit')
]

O.engines=O.engines+[PyRunner(dead=1,iterPeriod=iterper,command='history()',label='recorder')]
↳ # 8
O.engines=O.engines+[PyRunner(iterPeriod=iterper,command='pressureField()',dead=1,label='pressu
↳ # 9
O.engines=O.engines+[PyRunner(dead=1,iterPeriod=iterper,command='stopifDamaged()',label='stopif
↳ # 10

SSgeom.interactionDetectionFactor=1.
Saabb.aabbEnlargeFactor=1.

```

```

mn,mx = getMnMx()

# controlling walls for confining pressure
triax.stressMask = 7
triax.goal2=confining_pressure
triax.goal1=confining_pressure
triax.goal3=confining_pressure

triax.maxVel=0.0001
triax.depth0 = triax.depth

0.timingEnabled=1
0.resetTime()

if savedState:
    print('Using state:',savedStateName)
    0.load(savedStateName)
    flow.dead=1 # stop the old flowengine from operating, it doesnt have tri data
    del 0.engines[3] # delete the old flowengine (doesn't actually delete it within
    ↪ Yade, just the python list here)
    0.engines=0.engines[0:2]+[PartialSatClayEngine(dead=0,label='flow')]+0.engines[2:-1]
    ↪ # add a new flow engine and label it the same name 'flow'
    mn,mx = getMnMx()

### Tools to be used throughout simulation ###

def wallsOnOff(onOff):
    triax.wall_bottom_activated=onOff
    triax.wall_top_activated=onOff
    triax.wall_front_activated=onOff
    triax.wall_back_activated=onOff
    triax.wall_left_activated=onOff
    triax.wall_right_activated=onOff

def history():
    global originalVolume
    plot.addData(t=0.time,p=flow.getPorePressure((xsup/2,ysup/2,zsup/2)),
    centerSat=flow.getCellSaturation((xsup/2,ysup/2,zsup/2)),
    unbal = unbalancedForce(),
    displacement=triax.depth0-triax.depth,
    P = abs(0.forces.f(4)[2]),
    totVol = flow.totalSpecimenVolume,
    crackArea = flow.getCrackArea(),
    crackVolume = flow.getCrackVolume(),
    crackTotal = flow.crackedCellTotal,
    eps_v = (flow.totalSpecimenVolume-originalVolume)/originalVolume,
    s11=-triax.stress(triax.wall_right_id)[0],
    s22=-triax.stress(triax.wall_top_id)[1],
    s33=-triax.stress(triax.wall_front_id)[2]
    )
    plot.saveDataTxt(scale_folder+'txt'+timeStr+identifier+'/saturation'+identifier+'.txt',vars=('t

```

```

def scale_grid(grid_path, translation_vector):
    grid = np.loadtxt(grid_path)
    grid[:, :3] = grid[:, :3]*grid_scale
    grid[:, 0] = grid[:, 0]+translation_vector[0]
    grid[:, 1] = grid[:, 1]+translation_vector[1]
    grid[:, 2] = grid[:, 2]+translation_vector[2]

    ↪ np.savetxt(scale_folder+'txt'+timeStr+identifier+'porosityGrid_scale-'+str(grid_scale)+'.t

##make nice animations:
def pressureField():
    flow.saveUnsatVtk(scale_folder+'VTK'+timeStr+identifier+'/', withBoundaries=True)
    flow.savePermeabilityNetworkVTK(scale_folder+'VTK'+timeStr+identifier+'cellConnect-')

def stopifDamaged():
    P=plot.data['P']
    if 0.iter > 5000:
        if P[-1]>10 and P[-1] < 0.6*max(P):
            print('failure reached')
            #yade.timing.stats()
            0.pause()

def factorDownVelocities(factor):
    for b in 0.bodies:
        if isinstance(b.shape, Sphere):
            #b.dynamic=True
            b.state.vel=b.state.vel*factor

def printParticlePosition(i):
    pos_file =
    ↪ scale_folder+'txt'+timeStr+identifier+'particle_pos_id-'+str(i)+'.txt'
    if os.path.isfile(pos_file):
        f = open(pos_file, 'a')
    else:
        f = open(pos_file, 'w')

    pos = 0.bodies[i].state.pos

    f.write('%g %g %g %g\n' % (0.iter, pos[0], pos[1], pos[2]))
    f.close()

def printCellVertices():
    vertex_file =
    ↪ scale_folder+'txt'+timeStr+identifier+'cell_vertices_'+str(0.iter)+'.txt'
    f = open(vertex_file, 'w')

    for c in range(0, flow.nCells()):
        vs = flow.getVertices(c)
        f.write("%g %g %g %g\n" % (vs[0], vs[1], vs[2], vs[3]))

    f.close()

```

```

##### General flowengine parameters #####

for b in O.bodies:
    if isinstance(b.shape, Sphere):
        b.dynamic=dynamics

flow.dead = 0
flow.debug = 0

flow.numSolveThreads = numCores
flow.numFactorizeThreads = numCores

flow.defTolerance = -1
flow.meshUpdateInterval = -1
flow.useSolver = 4
viscosity_base = 1e-9
flow.viscosity = watervisosity
flow.permeabilityFactor = -kfactor # (-1e-8/flow.viscosity)*1e-3 #e-8 #-1e-16 #1e-12
↳ #-1e-11
flow.airViscosity = airViscosity

if not alpha:
    flow.bndCondIsPressure=[1,1,1,1,1,1] #
    flow.bndCondValue=[-pZero, -pZero, -pZero, -pZero, -pZero, -pZero]
    flow.boundaryUseMaxMin=[1,1,1,1,1,1]
else:
    flow.alphaBound=10000
    flow.alphaBoundValue=-pZero
    flow.fixedAlpha=1
flow.pZero=-pZero

##### Partial Sat param list #####
flow.freeSwelling=True
flow.particleSwelling=True

# swelling model parameters volStrain = betam / alpham * (exp(-alpham * state->suction) -
↳ exp(-alpham * suction0));
flow.betam =beta_eps #1.6e-8 # 1e-8 #for high spec #0.5e-8 for midhigh spec#2.605e-8
↳ #0.015e-6 ## this configuration is slightly exageratted swelling
flow.alpham = 2.102e-8 #0.024e-6
flow.minParticleSwellFactor=0.5

# water retention curve params (van genuchten, these are "avg" params, cell porosities
↳ will adjust individual cell params around these)
flow.lmbda = 0.08 # 0.4 #
flow.Po = 0.03e6 # 0.04e6 #

# permeability params
flow.nUnsatPerm = 1 # 5 # increasing this value decreases the effect of saturation on
↳ perm
flow.clampKValues = False

```

```

flow.bIntrinsicPerm = 2 # 8 decreasing this value decreases effect of porosity on perm
flow.SsM = 1 # saturated saturation
flow.SrM = 0.01 # residual saturation (these also clamp sat values in model)
flow.waterSurfaceTension = 7.28e-2

# porosity params
flow.minPoroClamp = 0.1
flow.maxPoroClamp = 0.8
flow.meanInitialPorosity = 0.25

if not hetero_poro: flow.constantPorosity = True
else:
    scale_grid(grid_path,translation_vector)
    flow.imageryFilePath =
        ↪ scale_folder+'txt'+timeStr+identifier+'/porosityGrid_scale-'+str(grid_scale)+'.txt'

# extraneous params
flow.convertClumps = False #
flow.fracBasedPointSuctionCalc = False
flow.minCellVol = 1e-20 # blocks cells below this volume
flow.getCHOLMODPerfTimings=False

# crack model params
flow.displacementBasedCracks = True # alternative to cohesion break
flow.changeCrackSaturation = crackActive
flow.apertureFactor = 1
flow.permAreaFactor = 0.01
flow.computeFracturePaths=False
flow.useOpeningPressure=True
flow.useForceForCracks=True

# time step parameters
0.dt=timeStep
0.dynDt=False
0.engines[4].active=False # dont need the stiffness timestepper if we are controlling TS
newton.damping=0.4 # playing with high damping

# VTK params
VTKrec.fileName=scale_folder+'VTK'+timeStr+identifier+'/spheres-'

# set walls away from packing

if confining_pressure==0:
    0.bodies[4].state.pos = (0,0,mn[2]-0.001)
    0.bodies[5].state.pos = (0,0,mx[2]+0.001)
    0.bodies[2].state.pos = (0,mn[1]-0.001,0)
    0.bodies[3].state.pos = (0,mx[1]+0.001,0)
    0.bodies[0].state.pos = (mn[0]-0.001,0,0)
    0.bodies[1].state.pos = (mx[0]+0.001,0,0)
##### Reach initial suction equilibrium #####

print("Equilibrating pellet to negative pore pressures before starting saturation")

```

```

flow.meshUpdateInterval=1000 #1000 #1000 # update infrequently, we don't care about the
↳ small changes during this
flow.multithread=multithread
flow.freezePorosity=True # during genesis, we don't alter porosity with volume changes
flow.swelling=False
flow.freezeSaturation=True
threshold = 0.05 #0.06
lastVolume = flow.totalSpecimenVolume

centerBody = bodyByPos((mx[0]-mn[0])/2,(mx[1]-mn[1])/2,(mx[2]-mn[2])/2)

wallsOnOff(False) # if confining_pressure==0: deactivate the walls

# for regular ortho, we need to randomize locations to avoid cgal instabilities
for b in O.bodies:
    if isinstance(b.shape, Sphere):
        rand = np.random.uniform(-1,1) #random()
        rand = (rand*0.00001)*b.shape.radius + b.shape.radius
        b.state.pos =
        ↳ (b.state.pos[0]+rand,b.state.pos[1]+rand,b.state.pos[2]+rand)

while 1:
    O.run(400,1)
    newVolume = flow.totalSpecimenVolume
    deltaVolume = newVolume - lastVolume
    lastVolume = newVolume
    unb = unbalancedForce()
    avgSuction = flow.getAverageSuction()
    print('unbalancedForce', unb,'ncells',flow.nCells(),'volume change',deltaVolume,
    ↳ 'simspeed',O.speed,'avgSuction',avgSuction)
    if not clusterScript: print('rss memory used (mb)',
    ↳ psutil.Process(os.getpid()).memory_info().rss/1e6,'rss memory
    ↳ used',psutil.Process(os.getpid()).memory_full_info().swap/1e6)
    if (not(dynamics) or unb <= threshold): # and centerBody.state.suction!=0:
        flow.freezePorosity=False
        flow.freezeSaturation=False
        flow.resetVolumeSolids=True
        flow.swelling = True
        threshold = 0.02
        print('finished genesis')
        if not savedState:
            O.save(savedStateName)
        break

mn,mx = getMnMx()

#wallsOnOff(True)

if partial_confine or confining_pressure!=0:
    # confine in z direction
    O.bodies[4].state.pos = (0,0,mn[2]) #-0.001)
    O.bodies[5].state.pos = (0,0,mx[2]) #+0.001)
    O.bodies[2].state.pos = (0,mn[1],0)

```

```

O.bodies[3].state.pos = (0,mx[1],0)
O.bodies[0].state.pos = (mn[0],0,0)
O.bodies[1].state.pos = (mx[0],0,0)

if not flow.constantPorosity:
    ↪ os.remove(scale_folder+'txt'+timeStr+identifier+'/porosityGrid_scale-'+str(grid_scale)+'.txt')
    ↪ # this file is too large to make copies. Delete when finished with it

for i in O.interactions:
    i.phys.initD = i.geom.penetrationDepth

##

def decreaseSuction(newTarget):
    if not alpha:
        flow.bndCondIsPressure = [1,1,1,1,1,1]
        flow.bndCondValue=[-newTarget, -newTarget, -newTarget, -newTarget, -newTarget, -newTarget]
    else:
        flow.alphaBoundValue=-newTarget

    flow.updateTriangulation=True
    #flow.resetRHS = True

# freeze everything, alter bound conds, and compute perm
def emulatePermeability(currentSuction,axis):

    file_name = 'suction_'+"{0:.1e}".format(currentSuction)+'_axis_'+str(axis)
    if not os.path.exists(scale_folder+'VTK'+timeStr+identifier+'/'+file_name):
        os.mkdir(scale_folder+'VTK'+timeStr+identifier+'/'+file_name)
    else:
        shutil.rmtree(scale_folder+'VTK'+timeStr+identifier+'/'+file_name)
        os.mkdir(scale_folder+'VTK'+timeStr+identifier+'/'+file_name)
    flow.freezePorosity = True
    flow.freezeSaturation = True
    flow.updateTriangulation = True
    flow.computeFracturePaths = False
    flow.meshUpdateInterval=-1

    for b in O.bodies:
        if isinstance(b.shape, Sphere):
            b.dynamic=False
            #b.state.vel=(0,0,0)

    dim=utils.aabbExtrema()
    xinf=dim[0][0]
    xsup=dim[1][0]
    X=xsup-xinf
    yinf=dim[0][1]
    ysup=dim[1][1]
    Y=ysup-yinf
    zinf=dim[0][2]
    zsup=dim[1][2]

```

```

Z=zsup-zinf
area = [(ysup-yinf)*(zsup-zinf), (zsup-zinf)*(xsup-xinf),
        ↪ (ysup-yinf)*(xsup-xinf)]
L = [(xsup-xinf), (ysup-yinf), (zsup-zinf)]
kstd = 0

# Setting the pressure gradient along the boundaries
delP = currentSuction*0.1
axis_up = [xsup,ysup,zsup]
down_pressure = -currentSuction-delP/2
up_pressure = -currentSuction+delP/2
slope = (up_pressure - down_pressure)/L[axis]
yintercept = up_pressure-(slope*axis_up[axis])

print("specimen length", L)

for i in range(0,flow.nCells()):
    coords = flow.getCellCenter(i)
    pressure_imposed = flow.getCellPImposed(i)
    if not pressure_imposed: continue
    zcoord = coords[axis]
    pressure = zcoord*slope+yintercept #currentSuction
    flow.imposePressureFromId(i,pressure) #(coords[0],coords[1],coords[2])

flow.emulateAction()
flow.emulateAction()
flow.emulateAction()

# visualization
flow.saveUnsatVtk(scale_folder+'VTK'+timeStr+identifier+'/' +file_name,withBoundaries=True)
flow.savePermeabilityNetworkVTK(scale_folder+'VTK'+timeStr+identifier+'/' +file_name+'/cellConne

## FLUID PERM
size_reduce =0.1
xlims,ylims,zlims =
    ↪ [xinf+L[axis]*size_reduce,xsup-L[axis]*size_reduce],[yinf+L[axis]*size_reduce,ysup-L[axis]*
velocitySum = 0
saturationSum = 0
poroSum = 0
numPts = 0
nans = 0
cellsHit = []
totalVelocity = np.array([0,0,0])
totalVolume = 0
v = np.array([0,0,0])

cellsInside = []
for i in range(0,flow.nCells()):
    coords = flow.getCellCenter(i)
    if flow.getCellPImposed(i): continue
    if xlims[0]<=coords[0]<=xlims[1] and ylims[0]<=coords[1]<=ylims[1] and
    ↪ zlims[0]<=coords[2]<=zlims[1]:
        cellsInside.append(i)

```

```

velocityVector =
    ↪ np.array(flow.getCellVelocity((coords[0], coords[1], coords[2])))
cellVol = flow.getCellVolume((coords[0], coords[1], coords[2]))
v = v + cellVol*velocityVector
totalVolume += cellVol
    #numPts += 1
print('cells used for volume perm', len(cellsInside))

q = np.linalg.norm(v)/totalVolume
kv = q*flow.viscosity/(slope)
print('perm by volume', kv)

kgas = 0
#    ## GAS PERM
flow.getGasPerm = True
flow.emulateAction()
flow.emulateAction()
flow.emulateAction()
#xlims, ylims, zlims = [-0.003, 0.003], [-0.003, 0.003], [-0.003, 0.003]
velocitySum = 0
saturationSum = 0
poroSum = 0
numPts = 0
nans = 0
cellsHit = []
totalVelocity = np.array([0, 0, 0])
totalVolume = 0
v = np.array([0, 0, 0])

cellsInside = []
for i in range(0, flow.nGasCells()):
    coords = flow.getCellGasCenter(i)
    if flow.getCellGasPImposed(i): continue
    if xlims[0] <= coords[0] <= xlims[1] and ylims[0] <= coords[1] <= ylims[1] and
        ↪ zlims[0] <= coords[2] <= zlims[1]:
        cellsInside.append(i)
        velocityVector =
            ↪ np.array(flow.getCellGasVelocity((coords[0], coords[1], coords[2])))
        velMag = np.linalg.norm(velocityVector)
        cellVol = flow.getCellGasVolume((coords[0], coords[1], coords[2]))
        v = v + cellVol*velocityVector
        totalVolume += cellVol
print('cells used for volume perm', len(cellsInside))

q = np.linalg.norm(v)/totalVolume
kgas = q*flow.airViscosity/(slope) #use flow.airViscosity
print('gas perm by volume', kgas)

flow.getGasPerm = False

flow.clearImposedPressure()

```

```

Qin=0
Qout=0

fv = flow.getCrackFabricMatrix()
if not isnan(fv[0][0]):
    crack_fabric_tensor = fv #np.outer(fv, fv)
    fileCrackFabric =
        ↪ scale_folder+'txt'+timeStr+identifier+'/fabric_tensor- '+'{0:.2e}'.format(currentSuction)
    np.savetxt(fileCrackFabric, crack_fabric_tensor)

fileName =
    ↪ scale_folder+'txt'+timeStr+identifier+'/collectedPerms_'+str(axis)+'.txt'
if os.path.isfile(fileName):
    f = open(fileName, 'a')
else:
    f = open(fileName, 'w')
    f.write('suction permV permStd flux crackArea crackVol crackPorosity epsV
        ↪ meanSat meanPorosity enteredRatio watervolume numcracks permGas kv
        ↪ avgAp\n')

flow.printPorosity(scale_folder+'txt'+timeStr+identifier+'/porosity- '+'{0:.1e}'.format(currentSuction))
dat =
    ↪ np.loadtxt(scale_folder+'txt'+timeStr+identifier+'/porosity- '+'{0:.1e}'.format(currentSuction))
os.remove(scale_folder+'txt'+timeStr+identifier+'/porosity- '+'{0:.1e}'.format(currentSuction))
    ↪ ## not really needed at the moment

meanPorosity = np.mean(dat[:,1])

f.write('%g %g %g\n' % (currentSuction,
    ↪ kv, -kstd, Qin, flow.getCrackArea(), flow.getCrackVolume(),
    ↪ flow.getCrackVolume()/flow.totalSpecimenVolume,
    ↪ (flow.totalSpecimenVolume-originalVolume)/originalVolume, flow.getAverageSaturation(), meanPorosity,
    ↪ kv, flow.getAverageAperture()))

f.close()

flow.freezePorosity = False
flow.freezeSaturation = False
flow.updateTriangulation = True
flow.meshUpdateInterval=-1 #1000
#flow.computeFracturePaths = True

for b in O.bodies:
    if isinstance(b.shape, Sphere):
        b.dynamic=dynamics
        #b.state.vel=(0,0,0)

```

```
##### Hydrate to target suction #####
```

```
print('Saturating pellet to target suction', targetSuction)
```

```
if confined: flow.forceConfinement=True
```

```
flow.updateTriangulation=True
```

```
hertzLaw.neverErase=True
```

```
flow.crackModelActive=crackActive
```

```
flow.brokenBondsRemoveCapillaryforces=crackActive
```

```
flow.meshUpdateInterval=1000
```

```
flow.defTolerance=-1
```

```
lastZdim = 100
```

```
domainlimit.dead=0
```

```
domainlimit.lo = (mn[0]-0.001,mn[1]-0.001,mn[2]-0.001)
```

```
domainlimit.hi = (mx[1]+0.001,mx[1]+0.001,mx[2]+0.001)
```

```
#pressureRec.dead,presureRec.iterPeriod=False,800
```

```
#VTKrec.dead,VTKrec.iterPeriod=False,800
```

```
VTKrec.recorders=['spheres','intr','partialSat','hertz','boxes','clumpId','velocity']
```

```
suction_list =
```

```
↳ [130e6,120e6,110e6,100e6,90e6,80e6,70e6,60e6,50e6,40e6,20e6,18e6,14e6,10e6,7e6,5e6,3e6,2e6,1e6,7e5,
```

```
tolerance_list = [1.1,1.1,1.1,1.1,1.1,1.1,1.3,2,3] # multiplier for accepted average pressure
```

```
↳ value
```

```
viscosity_list =
```

```
↳ [1e0,1e0,1e0,1e0,1e1,1e1,1e1,1e2,1e2,1e2,1e2,1e3,1e3,1e4,1e4,1e4,1e4,1e4,1e5,1e5,1e5,1e5,1e5,1e5,
```

```
↳ # multiplier for viscosity to speed up stabilization
```

```
suction_n = 0
```

```
lastVolume = flow.totalSpecimenVolume
```

```
global originalVolume
```

```
originalVolume = flow.totalSpecimenVolume
```

```
emulatePermeability(pZero,0)
```

```
emulatePermeability(pZero,1)
```

```
emulatePermeability(pZero,2)
```

```
# get initial permeability
```

```
currentSuction = 132e6
```

```
decreaseSuction(currentSuction)
```

```
if hydraulic_equilibrium: flow.homogeneousSuctionValue = -currentSuction
```

```
pressure_tolerance=1.1
```

```
firstIt = 0.iter
```

```
count = 0
```

```
factor = 6e-6
```

```
while 1:
```

```
    count+=1
```

```
    O.run(100,1)
```

```
    avgSuction = flow.getAverageSuction()
```

```
    unb = unbalancedForce()
```

```
    print("currentSuction", currentSuction, "avgSuction", avgSuction, "factor",
```

```
        ↳ factor)
```

```
    if currentSuction>suction_list[suction_n]:
```

```

    if unb < threshold: factor *= 1.01
    else: factor *= 0.99
    dpdt = factor * currentSuction ## 6.325e-6
    currentSuction = currentSuction - (dpdt*100)
    if hydraulic_equilibrium: flow.homogeneousSuctionValue = -currentSuction
    decreaseSuction(currentSuction)

if count!=20: continue
count = 0
newVolume = flow.totalSpecimenVolume
eps_v = (newVolume-originalVolume)/originalVolume
deltaVolume = newVolume - lastVolume
lastVolume = newVolume
mn,mx = getMnMx()
centerBody = bodyByPos((mx[0]-mn[0])/2,(mx[1]-mn[1])/2,(mx[2]-mn[2])/2)

Z = mx[2]-mn[2]

print('unbalanced force', unb, 'avgSuction', avgSuction, 'zdim change',
↳ (Z-lastZdim)/Z, 'volume change', deltaVolume, 'eps_v', eps_v,
↳ 'saturation',flow.getAverageSaturation(), 'simspeed',0.speed)
if not clusterScript: print('rss memory used (mb)',
↳ psutil.Process(os.getpid()).memory_info().rss/1e6,'rss memory
↳ used',psutil.Process(os.getpid()).memory_full_info().swap/1e6)
if (not(dynamics) or unb <= threshold) and currentSuction>targetSuction and
↳ abs((Z-lastZdim)/Z) <= 0.001 and avgSuction<=suction_list[suction_n]*1.005:
↳ #currentSuction*pressure_tolerance:
    #if currentSuction<=suction_list[suction_n]:
    suction_n+=1
    emulatePermeability(currentSuction,0) # emulating perm is relatively
↳ expensive, so avoiding too frequently
    emulatePermeability(currentSuction,1)
    emulatePermeability(currentSuction,2)
    flow.viscosity = watervisosity/(viscosity_list[suction_n])
    flow.airViscosity = airViscosity/(viscosity_list[suction_n])

    #suction_n += 1
    if suction_n == len(suction_list): #currentSuction <= targetSuction: #
        tock = time.time()
        print('target suction reached',str(currentSuction/1e6),'MPa
↳ reached at center of specimen. Total crack
↳ area',flow.getCrackArea(),' Total crack
↳ volume',flow.getCrackVolume(), 'crack porosity',
↳ flow.getCrackVolume()/flow.totalSpecimenVolume, 'total time',
↳ tock-tick)
        break

else:
    lastZdim=Z

```

```
if batch: sys.exit()

if captureStdout:
    captured_stdout = os.read(stdout_pipe[0],20000)
    os.close(stdout_fileno)
    os.dup2(stdout_save,stdout_fileno)
    os.close(stdout_pipe[0])
    f = open(scale_folder+'txt'+timeStr+identifier+'/log.txt', 'w')
    f.write(str(captured_stdout,'utf-8'))
    f.close()
    print(str(captured_stdout,'utf-8'))
    quit()
    sys.exit()
```


Bibliography

Here are the references in citation order.

- [1] IEA. *World Energy Outlook 2021*. Tech. rep. IEA, 2021 (cited on page 3).
- [2] ANDRA. *National Inventory of Radioactive Materials and Waste 2020 - The Essentials*. Tech. rep. 2020 (cited on page 3).
- [3] ANDRA. *National Inventory of Radioactive Materials and Waste*. Tech. rep. ANDRA, 2015 (cited on page 3).
- [4] N. Mokni and J. D. Barnichon. 'Hydro-mechanical analysis of SEALEX in-situ tests - Impact of technological gaps on long term performance of repository seals'. In: *Engineering Geology* 205 (2016), pp. 81–92. doi: [10.1016/j.enggeo.2016.02.013](https://doi.org/10.1016/j.enggeo.2016.02.013) (cited on pages 3, 28).
- [5] J. D. Barnichon, P. Dick, and C. Bauer. 'The SEALEX in situ experiments: performance tests of repository seals'. In: *12th IRSM Congress*. International Society for Rock Mechanics and Rock Engineering, 2011 (cited on pages 3, 14, 28).
- [6] R. Pusch. 'Highly Compacted Sodium Bentonite for Isolating Rock-Deposited Radioactive Waste Products.' In: *Nucl Technol* 45.2 (1979), pp. 153–157. doi: [10.13182/NT79-A32305](https://doi.org/10.13182/NT79-A32305) (cited on pages 4, 14).
- [7] A. Gens and E. E. Alonso. 'A framework for the behaviour of unsaturated expansive clays'. In: *Canadian Geotechnical Journal* 29 (1992), pp. 1013–1032 (cited on pages 4, 12–14, 28).
- [8] P. Delage, M. D. Howat, and Y. J. Cui. 'The relationship between suction and swelling properties in a heavily compacted unsaturated clay'. In: *Engineering Geology* 50.1-2 (1998), pp. 31–48. doi: [10.1016/S0013-7952\(97\)00083-5](https://doi.org/10.1016/S0013-7952(97)00083-5) (cited on pages 4, 14, 28).
- [9] R. N. Yong. 'Soil suction and soil-water potentials in swelling clays in engineered clay barriers'. In: *Engineering Geology* 54.1-2 (1999), pp. 3–13. doi: [10.1016/S0013-7952\(99\)00056-3](https://doi.org/10.1016/S0013-7952(99)00056-3) (cited on pages 4, 12–14, 28).
- [10] G. Volckaert et al. *A large-scale demonstration test for repository sealing in an argillaceous host rock (Reseal project. Phase I)*. Tech. rep. EU publications, 2000 (cited on pages 4, 14, 28).
- [11] A. Gens et al. 'Hydromechanical behaviour of a heterogeneous compacted soil: experimental observations and modelling'. In: *Géotechnique* 61.5 (2011), pp. 367–386. doi: [10.1680/geot.sip11.p.015](https://doi.org/10.1680/geot.sip11.p.015) (cited on pages 4, 12–14, 23, 28, 46).
- [12] E. Romero, J. Vaunat, and V. Merchán. 'Suction effects on the residual shear strength of clays'. In: *Journal of Geo-Engineering Sciences* 2.1-2 (2014), pp. 17–37. doi: [10.3233/JGS-141320](https://doi.org/10.3233/JGS-141320) (cited on pages 4, 14, 28).
- [13] A. Molinero-Guerra et al. 'In-depth characterisation of a mixture composed of powder / pellets MX80 bentonite'. In: *Applied Clay Science* 135 (2017), pp. 538–546 (cited on pages 4, 14, 81).
- [14] A. Molinero-Guerra. 'Experimental and numerical characterizations of the hydro-mechanical behavior of a heterogeneous material : pellet / powder bentonite mixture'. PhD thesis. Université Paris-Est, 2018 (cited on pages 4, 7, 13, 15, 16, 23, 24, 26, 28–31, 34, 47, 81, 82).
- [15] N. Mokni et al. 'Modelling the long-term hydro-mechanical behaviour of a bentonite pellet/powder mixture with consideration of initial structural heterogeneities'. In: *Geotechnique* 70.7 (2020), pp. 563–580. doi: [10.1680/jgeot.18.P.110](https://doi.org/10.1680/jgeot.18.P.110) (cited on pages 4, 43, 82).
- [16] B. Darde et al. 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'. In: *Engineering Geology* 278.May (2020), p. 105847. doi: [10.1016/j.enggeo.2020.105847](https://doi.org/10.1016/j.enggeo.2020.105847) (cited on pages 7, 43, 54).
- [17] B. Darde, A.-m. Tang, and J.-m. Pereira. 'Comportment hydromecanique de pellets de bentonite: caracterisation au laboratoire et simulations DEM swelling'. In: *Nationales de Geotechnique et de Geologie de l'ingenieur*. 2018 (cited on pages 7, 13, 30, 31).

- [18] H. Darcy. 'Les fontaines Publiques de la ville de Dijon, Paris: Dalmont, 1856'. In: *Google Scholar* (1856) (cited on page 11).
- [19] M. Muskat. 'The flow of homogeneous fluids through porous media'. In: *Soil Science* 46.2 (1938), p. 169 (cited on page 11).
- [20] E. E. Alonso, J. Vaunat, and A. Gens. 'Modelling the mechanical behaviour of expansive clays'. In: *Engineering Geology* 54.1-2 (1999), pp. 173–183. doi: [10.1016/S0013-7952\(99\)00079-4](https://doi.org/10.1016/S0013-7952(99)00079-4) (cited on pages 11–13).
- [21] T. W. Lambe. 'The engineering behavior of compacted clay'. In: *Journal of the Soil Mechanics and Foundations Division* 84.2 (1958), pp. 1651–1655 (cited on page 11).
- [22] S. Horpibulsuk et al. 'Compressibility and permeability of Bangkok clay compared with kaolinite and bentonite'. In: *Applied Clay Science* 52.1-2 (2011), pp. 150–159. doi: [10.1016/j.clay.2011.02.014](https://doi.org/10.1016/j.clay.2011.02.014) (cited on page 12).
- [23] R. Pusch. 'Permeability of highly compacted bentonite'. In: *Technical Report* 80-16 (1980), p. 46 (cited on page 12).
- [24] K Norrish. 'The Swelling of Montmorillonite'. In: *Discussions of the Faraday Society* (1954), pp. 120–134 (cited on page 12).
- [25] A. Anandarajah and P. Amarasinghe. 'Discrete-element study of the swelling behaviour of Namontmorillonite'. In: *Géotechnique* 63.8 (2013), pp. 674–681. doi: [10.1680/geot.12.P.012](https://doi.org/10.1680/geot.12.P.012) (cited on page 12).
- [26] M. Yao and A. Anandarajah. 'Three-Dimensional Discrete Element Method of Analysis of Clays'. In: *Journal of Engineering Mechanics* 129.6 (2003), pp. 585–596. doi: [10.1061/\(ASCE\)0733-9399\(2003\)129:6\(585\)](https://doi.org/10.1061/(ASCE)0733-9399(2003)129:6(585)) (cited on page 12).
- [27] D. R. Katti et al. 'Multiscale modeling of swelling clays: A computational and experimental approach'. In: *KSCE Journal of Civil Engineering* 13.4 (2009), pp. 243–255. doi: [10.1007/s12205-009-0243-0](https://doi.org/10.1007/s12205-009-0243-0) (cited on page 12).
- [28] P. Amarasinghe and A. Anandarajah. 'Behavior of Swelling Clays: A Molecular Dynamic Study'. In: *GeoCongress*. American Society of Civil Engineers, 2012, pp. 2452–2461 (cited on page 12).
- [29] E. E. Alonso, A. Gens, and A. Josa. 'A constitutive model for partially saturated soils'. In: *Geotechnique* 40.3 (1990), pp. 405–430. doi: [10.1680/geot.1990.40.3.405](https://doi.org/10.1680/geot.1990.40.3.405) (cited on pages 12, 13).
- [30] A. Tarantino. 'A possible critical state framework for unsaturated compacted soils'. In: *Geotechnique* 57.4 (2007), pp. 385–389. doi: [10.1680/geot.2007.57.4.385](https://doi.org/10.1680/geot.2007.57.4.385) (cited on pages 12, 13).
- [31] E. E. Alonso, E. Romero, and C. Hoffmann. 'Hydromechanical behaviour of compacted granular expansive mixtures: Experimental and constitutive study'. In: *Geotechnique* 61.4 (2011), pp. 329–344. doi: [10.1680/geot.2011.61.4.329](https://doi.org/10.1680/geot.2011.61.4.329) (cited on pages 12, 13).
- [32] N. Mokni. 'Analysis of hydro-mechanical behaviour of compacted bentonite/sand mixture using a double structure formulation'. In: *Environmental Earth Sciences* 75.14 (2016). doi: [10.1007/s12665-016-5872-2](https://doi.org/10.1007/s12665-016-5872-2) (cited on pages 12, 13, 15, 23–25, 30, 46, 47, 55).
- [33] A. M. Ridley. 'Soil suction — what it is and how to successfully measure it'. In: *Australian Centre for Geomechanics* (2015), pp. 27–46 (cited on page 13).
- [34] G. Pinder and W. Gray. *Essentials of multiphase flow and transport in porous media*. John Wiley & Sons, 2008 (cited on page 13).
- [35] E. Romero, A. Gens, and A. Lloret. 'Water permeability, water retention and microstructure of unsaturated compacted Boom clay'. In: *Engineering Geology* 54.1-2 (1999), pp. 117–127. doi: [10.1016/S0013-7952\(99\)00067-8](https://doi.org/10.1016/S0013-7952(99)00067-8) (cited on page 13).
- [36] B. Darde et al. 'Modelling the behaviour of bentonite pellet-powder mixtures upon hydration from dry granular state to saturated homogeneous state'. In: *Engineering Geology* 278.September (2020), p. 105847. doi: [10.1016/j.enggeo.2020.105847](https://doi.org/10.1016/j.enggeo.2020.105847) (cited on pages 13, 23, 47).

- [37] C. H. Delahaye and E. E. Alonso. 'Soil heterogeneity and preferential paths for gas migration'. In: *Engineering Geology* 64.2-3 (2002), pp. 251–271. doi: [10.1016/S0013-7952\(01\)00104-1](https://doi.org/10.1016/S0013-7952(01)00104-1) (cited on page 13).
- [38] S. Olivella and E. E. Alonso. 'Gas flow through clay barriers'. In: *Geotechnique* 58.3 (2008), pp. 157–176. doi: [10.1680/geot.2008.58.3.157](https://doi.org/10.1680/geot.2008.58.3.157) (cited on page 13).
- [39] C. Imbert et al. 'Hydro mechanical behaviour of a heterogeneous swelling clay material'. In: *Poromechanics II* (2020), pp. 225–230. doi: [10.1201/9781003078807-34](https://doi.org/10.1201/9781003078807-34) (cited on page 13).
- [40] J. B. Martino et al. *The Tunnel Sealing Experiment : The Construction and Performance of Full Scale Clay and Concrete Bulkheads at Elevated Pressure and Temperature*. Tech. rep. INIS-BE-V-001, 2006, pp. 2–6 (cited on page 14).
- [41] D. A. Dixon et al. 'Enhanced Sealing Project (ESP): evolution of a full-sized bentonite and concrete shaft seal'. In: *Clays in Natural and Engineered Barriers for Radioactive Waste Confinement*. Ed. by S Norris et al. Vol. 400. Geological Society of London, 2014. doi: [10.1144/SP400.33](https://doi.org/10.1144/SP400.33) (cited on page 14).
- [42] N. Mokni et al. 'Effect of technological macro voids on the performance of compacted bentonite/sand seals for deep geological repositories'. In: *International Journal of Rock Mechanics and Mining Sciences* 88 (2016), pp. 87–97. doi: [10.1016/j.ijrmms.2016.07.011](https://doi.org/10.1016/j.ijrmms.2016.07.011) (cited on pages 14, 30).
- [43] Y. Wan et al. 'Crack Characteristic and Permeability Change of Compacted Clay Liners with Different Liquid Limits under Dry-Wet Cycles'. In: *Advances in Civil Engineering* 2018 (2018). doi: [10.1155/2018/5796086](https://doi.org/10.1155/2018/5796086) (cited on page 14).
- [44] M. De Camillis et al. 'Hydraulic conductivity and swelling ability of a polymer modified bentonite subjected to wet–dry cycles in seawater'. In: *Geotextiles and Geomembranes* 44.5 (2016), pp. 739–747. doi: [10.1016/j.geotexmem.2016.05.007](https://doi.org/10.1016/j.geotexmem.2016.05.007) (cited on page 14).
- [45] G. Didier, A. Bouazza, and D. Cazaux. 'Gas permeability of geosynthetic clay liners'. In: *Geotextiles and Geomembranes* 18 (2000), pp. 235–250. doi: [10.17794/rgn.2017.1.2](https://doi.org/10.17794/rgn.2017.1.2) (cited on page 14).
- [46] J. F. Liu, F. Skoczylas, and J. Talandier. 'Gas permeability of a compacted bentonite-sand mixture: Coupled effects of water content, dry density, and confining pressure'. In: *Canadian Geotechnical Journal* 52.8 (2015), pp. 1159–1167. doi: [10.1139/cgj-2014-0371](https://doi.org/10.1139/cgj-2014-0371) (cited on page 14).
- [47] T. Wei et al. 'Influences of degree of saturation and stress cycle on gas permeability of unsaturated compacted Gaomiaozi bentonite'. In: *Engineering Geology* 254. April (2019), pp. 54–62. doi: [10.1016/j.enggeo.2019.04.005](https://doi.org/10.1016/j.enggeo.2019.04.005) (cited on page 14).
- [48] J. F. Liu et al. 'Laboratory investigation on gas permeability of compacted GMZ bentonite under a coupled hydraulic-mechanical effect'. In: *Engineering Geology* 276 (2020). doi: [10.1016/j.enggeo.2020.105761](https://doi.org/10.1016/j.enggeo.2020.105761) (cited on page 14).
- [49] C. O'Sullivan. *Particulate discrete element modelling: a geomechanics perspective*. CRC Press, 2011, p. 574 (cited on page 14).
- [50] P. A. Cundall and O. D. L. Strack. 'A discrete numerical model for granular assemblies'. In: *Géotechnique* 29.1 (1979), pp. 47–65. doi: [10.1680/geot.1979.29.1.47](https://doi.org/10.1680/geot.1979.29.1.47) (cited on pages 14, 20).
- [51] V. Šmilauer and B. Chareyre. 'DEM Formulation, Release Yade documentation'. In: *Yade Documentation* (2015), pp. 137–160. doi: [10.5281/zenodo.34044](https://doi.org/10.5281/zenodo.34044) (cited on pages 15, 83).
- [52] H. P. Zhu et al. 'Discrete particle simulation of particulate systems: A review of major applications and findings'. In: *Chemical Engineering Science* 63.23 (2008), pp. 5728–5770. doi: [10.1016/j.ces.2008.08.006](https://doi.org/10.1016/j.ces.2008.08.006) (cited on page 15).
- [53] L. Scholtès and F. V. Donzé. 'A DEM model for soft and hard rocks: Role of grain interlocking on strength'. In: *Journal of the Mechanics and Physics of Solids* 61 (2012), pp. 352–369. doi: [10.1016/j.jmps.2012.10.005](https://doi.org/10.1016/j.jmps.2012.10.005) (cited on page 15).
- [54] R. A. Caulk. 'Modeling acoustic emissions in heterogeneous rocks during Element Method'. In: *Open Geomechanics* 2 (2020) (cited on page 15).

- [55] F. Camborde, C. Mariotti, and F. V. Donzé. 'Numerical study of rock and concrete behaviour by discrete element modelling'. In: *Computers and Geotechnics* 27.4 (2000), pp. 225–247. doi: [10.1016/S0266-352X\(00\)00013-6](https://doi.org/10.1016/S0266-352X(00)00013-6) (cited on page 15).
- [56] L. Scholtès et al. 'Micromechanics of granular materials with capillary effects (DOI:10.1016/j.ijengsci.2008.07.002)'. In: *International Journal of Engineering Science* 47.11-12 (2009), pp. 1460–1471. doi: [10.1016/j.ijengsci.2009.10.003](https://doi.org/10.1016/j.ijengsci.2009.10.003) (cited on page 15).
- [57] B. Chareyre et al. 'Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings'. In: *Transport in Porous Media* 94.2 (2012), pp. 595–615. doi: [10.1007/s11242-012-0057-2](https://doi.org/10.1007/s11242-012-0057-2) (cited on pages 15, 21, 25, 26, 49, 60).
- [58] E. Catalano, B. Chareyre, and E. Barthelemy. 'Pore-scale modeling of fluid-particles interaction and emerging poromechanical effects'. In: *International Journal for Numerical and Analytical Methods in Geomechanics* 38.1 (2014), pp. 51–71. doi: [10.1002/nag.2198](https://doi.org/10.1002/nag.2198) (cited on pages 15, 16, 60, 64).
- [59] E. Papachristos et al. 'Intensity and volumetric characterizations of hydraulically driven fractures by hydro-mechanical simulations'. In: *International Journal of Rock Mechanics and Mining Sciences* 93 (2017), pp. 163–178. doi: [10.1016/j.ijrmms.2017.01.011](https://doi.org/10.1016/j.ijrmms.2017.01.011) (cited on pages 15, 16, 27).
- [60] C. Chalak et al. 'Partially saturated media: from DEM simulation to thermodynamic interpretation'. In: *European Journal of Environmental and Civil Engineering* 21.7-8 (2017), pp. 798–820. doi: [10.1080/19648189.2016.1164087](https://doi.org/10.1080/19648189.2016.1164087) (cited on pages 15, 16).
- [61] C. Yuan, B. Chareyre, and F. Darve. *Deformation and stresses upon drainage of an idealized granular material*. 2017. doi: [10.1007/s11440-017-0601-x](https://doi.org/10.1007/s11440-017-0601-x) (cited on pages 15, 16).
- [62] T. Sweijen et al. 'Dynamic pore-scale model of drainage in granular porous media: the pore-unit assembly method'. In: *Water Resources Research* (2018). doi: [10.1029/2017WR021769](https://doi.org/10.1029/2017WR021769) (cited on pages 15, 16, 23).
- [63] R. Caulk et al. 'A pore-scale thermo-hydro-mechanical model for particulate systems'. In: *Computer Methods in Applied Mechanics and Engineering* 372 (2020). doi: [10.1016/j.cma.2020.113292](https://doi.org/10.1016/j.cma.2020.113292) (cited on pages 15, 16).
- [64] H. Péron, T. Hueckel, and L. Laloui. 'An improved volume measurement for determining soil water retention curves'. In: *Geotechnical Testing Journal* 30.1 (2007), pp. 1–8. doi: [10.1520/gtj100167](https://doi.org/10.1520/gtj100167) (cited on pages 15, 16, 26).
- [65] J. F. Liu and F. Skoczylas. 'Experimental research on water retention and gas permeability of compacted bentonite/sand mixtures'. In: *Soils and Foundations* 54.5 (2014), pp. 1027–1038. doi: [10.1016/j.sandf.2014.09.011](https://doi.org/10.1016/j.sandf.2014.09.011) (cited on pages 15, 16, 26).
- [66] I. M. Smith, D. V. Griffiths, and L. Margetts. *Programming the finite element method*. John Wiley & Sons, 2013 (cited on page 16).
- [67] S.-H. Ju. 'A simple OpenMP scheme for parallel iteration solvers in finite element analysis'. In: *Computer Modeling in Engineering & Sciences (CMES)* 64.1 (2010), pp. 91–108 (cited on page 16).
- [68] P. Jimack and N. Touheed. 'Developing parallel finite element software using MPI'. In: *High Performance Computing for Computational Mechanics* (2000), pp. 15–38 (cited on page 16).
- [69] Y. Liu et al. 'GPU accelerated fast FEM deformation simulation'. In: *APCCAS 2008-2008 IEEE Asia Pacific Conference on Circuits and Systems*. IEEE, 2008, pp. 606–609 (cited on page 16).
- [70] A. Kakay, E. Westphal, and R. Hertel. 'Speedup of FEM micromagnetic simulations with graphical processing units'. In: *IEEE transactions on magnetics* 46.6 (2010), pp. 2303–2306 (cited on page 16).
- [71] K. Chen. *Matrix preconditioning techniques and applications*. Vol. 19. Cambridge University Press, 2005 (cited on page 16).
- [72] J. D. Booth et al. 'A multilevel Cholesky conjugate gradients hybrid solver for linear systems with multiple right-hand sides'. In: *Procedia Computer Science* 4 (2011), pp. 2307–2316. doi: [10.1016/j.procs.2011.04.251](https://doi.org/10.1016/j.procs.2011.04.251) (cited on page 16).

- [73] T. A. Davis and W. W. Hager. ‘Dynamic supernodes in sparse cholesky update/downdate and triangular solves’. In: *ACM Transactions on Mathematical Software* 35.4 (2009). doi: [10.1145/1462173.1462176](https://doi.org/10.1145/1462173.1462176) (cited on page 16).
- [74] A. Molinero-Guerra et al. ‘Analysis of the structural changes of a pellet/powder bentonite mixture upon wetting by X-ray computed microtomography’. In: *Applied Clay Science* 165.April (2018), pp. 164–169. doi: [10.1016/j.clay.2018.07.043](https://doi.org/10.1016/j.clay.2018.07.043) (cited on pages 43, 54).
- [75] H. R. Hertz. ‘Ueber die Berührung fester elastischer Körper (On Contact Between Elastic Bodies)’. In: *Journal für die reine und angewandte Mathematik (Crelle’s Journal)* 1 (1882). doi: [10.1515/crll.1882.92.156](https://doi.org/10.1515/crll.1882.92.156) (cited on pages 20, 43).
- [76] R. D. Mindlin. ‘Compliance of Elastic Bodies in Contact’. In: *Journal of Applied Mechanics* 16.3 (1949), pp. 259–268. doi: [10.1115/1.4009973](https://doi.org/10.1115/1.4009973) (cited on pages 20, 43).
- [77] B. Darde et al. ‘Hydro-mechanical behaviour of high-density bentonite pellet on partial hydration’. In: *Geotechnique Letters* 8.4 (2018), pp. 330–335. doi: [10.1680/jgele.18.00114](https://doi.org/10.1680/jgele.18.00114) (cited on pages 43, 44).
- [78] R Van Genuchten. ‘A closed form equation for predicting the hydraulic conductivity of unsaturated soils.’ In: *Soil Science Society*, 44, 892-898 (1980) (cited on pages 22, 25, 47).
- [79] Q. Wang et al. ‘Experimental study on the swelling behaviour of bentonite/claystone mixture’. In: *Engineering Geology* 124.1 (2012), pp. 59–66. doi: [10.1016/j.enggeo.2011.10.003](https://doi.org/10.1016/j.enggeo.2011.10.003) (cited on page 48).
- [80] V. Smilauer et al. *Yade documentation 3rd ed.* 2021 (cited on pages 20, 53, 54).
- [81] R. A. Caulk, N. Mokni, and B. Chareyre. ‘Modeling the transience of partial saturation and cracks in heterogeneous swelling clays using the Discrete Element and Finite Volume methods’. In: *Granular Matter* () (cited on page 19).
- [82] P. A. Witherspoon et al. ‘Validity of Cubic Law for fluid flow in a deformable rock fracture’. In: *Water Resources Research* 16 (1980), p. 1016. doi: [10.1029/WR016i006p01016](https://doi.org/10.1029/WR016i006p01016) (cited on page 27).
- [83] M. Oda. ‘Fabric tensor for discontinuous geological materials’. In: *Soils and Foundations* 4 (1982), pp. 96–108 (cited on page 27).
- [84] R. Shertzer. ‘Fabric tensors and effective properties of granular materials with applications to snow’. PhD thesis. Montana State University, 2011, p. 258 (cited on page 27).
- [85] E. Fogel and M. Teillaud. ‘The Computational Geometry Algorithms Library CGAL’. In: *ACM Communications in Computer Algebra* (2015). doi: [10.1145/2768577.2768579](https://doi.org/10.1145/2768577.2768579) (cited on page 28).
- [86] G. Pekmezi, D. Littlefield, and B. Chareyre. ‘Statistical distributions of the elastic moduli of particle aggregates at the mesoscale’. In: *International Journal of Impact Engineering* 139 (2020), p. 103481. doi: <https://doi.org/10.1016/j.ijimpeng.2019.103481> (cited on page 28).
- [87] S. M. Shirazi et al. ‘Permeability and swelling characteristics of bentonite’. In: *International Journal of Physical Sciences* 5.11 (2010), pp. 1647–1659 (cited on page 33).
- [88] Yade. *Yade source code*. 2021. URL: <https://gitlab.com/yade-dev/trunk> (cited on pages 38, 72).
- [89] Yade. *Yade home page*. 2021. URL: <https://www.yade-dem.org/doc/index.html> (cited on pages 38, 72).
- [90] R. Caulk, E. Catalano, and B. Chareyre. ‘Accelerating Yade’s poromechanical coupling with matrix factorization reuse, parallel task management, and GPU computing’. In: *Computer Physics Communications* 248 (2020). doi: [10.1016/j.cpc.2019.106991](https://doi.org/10.1016/j.cpc.2019.106991) (cited on page 59).
- [91] E. Catalano et al. ‘A pore-scale hydro-mechanical coupled model for geomaterials’. In: *II International Conference on Particle-based Methods. Fundamentals and Applications*. January 2011. 2011, pp. 798–809 (cited on page 60).
- [92] E. Catalano. ‘A pore-scale coupled hydromechanical model for biphasic granular media. Application to granular sediment hydrodynamics’. PhD thesis. Universite de Grenoble, 2012, p. 200 (cited on page 61).
- [93] T. a. Davis. ‘User Guide for CHOLMOD : a sparse Cholesky factorization and modification package’. In: *Department of Computer and Information Science and . . .* (2013), pp. 1–140. doi: [10.1.1.220.238](https://doi.org/10.1.1.220.238) (cited on page 63).

- [94] S. C. Rennich, D. Stosic, and T. A. Davis. 'Accelerating sparse Cholesky factorization on GPUs'. In: *Parallel Computing* 59 (2016), pp. 140–150. doi: [10.1016/j.parco.2016.06.004](https://doi.org/10.1016/j.parco.2016.06.004) (cited on page 63).
- [95] R. A. Hosn et al. 'Discrete numerical modeling of loose soil with spherical particles and interparticle rolling friction'. In: *Granular matter* 19.1 (2017), p. 4 (cited on page 70).
- [96] O. Stamati et al. 'Spam: software for practical analysis of materials'. In: *Journal of Open Source Software* 5.51 (2020), p. 2286 (cited on page 8).