Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering

Dissertation Thesis

Mathematical modeling of two-phase compositional flow in porous media

Prague 2022

Jakub Solovský

This thesis is submitted to the Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, in partial fulfilment of the requirements for the degree of Doctor of Philosophy (Ph.D.) in Mathematical Engineering.

Bibliografický záznam

Autor	Ing. Jakub Solovský, České vysoké učení technické v Praze, Fakulta jaderná a fyzikálně inženýrská, Katedra matematiky
Název práce	Matematické modelování dvoufázového kompozičního proudění v porézním prostředí
Studijní program	Aplikace přírodních věd
Studijní obor	Matematické inženýrství
Školitel	Doc. Ing. Radek Fučík, Ph.D., České vysoké učení technické v Praze, Fakulta jaderná a fyzikálně inženýrská, Katedra matematiky
Akademický rok	2022
Počet stran	174
Klíčová slova	BDDC, kompoziční proudění, metoda rozdělení oblasti, rozpouštění plynu, uvolňování plynu, kinetický model přestupu hmoty, smíšená hybridní metoda konečných prvků, MPI, porézní prostředí, dvoufázové proudění

Bibliographic Entry

Author	Ing. Jakub Solovský, Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Department of Mathematics
Title of dissertation	Mathematical modeling of two-phase compositional flow in porous media
Degree programme	Application of Natural Sciences
Field of study	Mathematical Engineering
Supervisor	Doc. Ing. Radek Fučík, Ph.D., Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Department of Mathematics
Academic year	2022
Number of pages	174
Keywords	BDDC, compositional flow, domain decomposition, gas dissolution, gas exsolution, kinetic mass transfer, mixed-hybrid finite element method, MPI, porous media, two-phase flow

Abstrakt

Tato práce se věnuje matematickému modelování dvoufázového kompozičního proudění v porézním prostředí s přestupem hmoty mezi fázemi. Hlavním cílem práce je zkoumání vývinu a rozpouštění plynu v nenasycené zóně porézního prostředí s využitím laboratorních dat.

Je formulován matematický model a rovnice popisující dvoufázové proudění, transport komponent a kinetický model přestupu hmoty mezi fázemi v porézním prostředí. Pro numerické řešení vzniklého systému rovnic navrhneme numerickou metodu založenou na smíšené hybridní metodě konečných prvků, kterou implementujeme sériově i paralelně a důkladně otestujeme na úlohách se známým řešením. Spočítáme experimentální řád konvergence potvrzující první řád přesnosti metody a diskutujeme modifikace metody pro úlohy v heterogenním porézním prostředí. Pro paralelní variantu metody provedeme testy škálovatelnosti, které ukazují dobrou efektivitu metody a její implementace při použití až tisíce CPU jader.

S využitím experimentálních dat ukážeme význam kinetického modelu přestupu hmoty pro předpověď vývinu CO_2 v laboratorních podmínkách. Navíc ukážeme rozdílnou dynamiku vývinu a rozpouštění plynu a její reprezentaci v matematickém modelu. Cílem studie je ukázat rozdíly v přestupu hmoty pro různé režimy proudění a navrhnout strategii pro další výzkum. Numerické schéma je dále využito pro zkoumání transportu kontrastní látky v jednofázovém případě v heterogenním porézním prostředí. V tomto případě jsou využita data z experimentů v laboratorních podmínkách zkoumajících únik slané vody z hlubinného rezervoáru.

Abstract

This work deals with the mathematical modeling of two-phase compositional flow in porous media with interphase mass transfer. The main goal is to investigate the gas exsolution and dissolution in the unsaturated zone of porous media using data from the laboratory experiments.

We formulate the mathematical model and present the equations describing the two-phase flow, component transport, and kinetic interphase mass transfer in porous media. For the numerical solution of the resulting system, we propose a numerical method based on the mixed hybrid finite element method (MHFEM) that is implemented both in serial and parallel, and thoroughly benchmarked on problems with known solutions. The experimental orders of convergence are computed confirming the first order of accuracy of the method and the necessary modification for the problems in heterogeneous media are discussed. The scaling tests are computed for the parallel variant of the method showing a good efficiency of the method and its implementation up to a thousand of CPU cores.

Using the experimental data we show the importance of the kinetic mass transfer model for the accurate predictions of multiphase CO_2 evolution at the laboratory-scale. Moreover, we demonstrate the different dynamics of the exsolution and dissolution processes and their representation in the mathematical model. The study aims to point out the differences in the mass transfer processes between the different flow patterns and expose strategies that should be explored in future research. In addition, the numerical scheme is used for the investigation of the single-phase problem of tracer transport in highly heterogeneous porous media in the laboratory scenarios mimicking the brine leakage from a deep aquifer.

Acknowledgements

I would like to thank everyone who gave me any kind of support in any of my activities during the time when I was a Ph.D. student. I would like to thank my supervisor, Doc. Ing. Radek Fučík, Ph.D., who has supported me throughout this research project. I am also grateful to Prof. Tissa H. Illangasekare, CESEP, Colorado School of Mines, Golden, Colorado, for providing experimental datasets and for his ideas and insights into the field concerning flow and transport in porous media

Our work related to the topic of this thesis was supported by the following projects:

- Advanced Supercomputing Methods for Implementation of Mathematical Models, project of the Student Grant Agency of the Czech Technical University in Prague
  No. SGS11/161/OHK4/3T/14
- Advanced supercomputing methods for mathematical modeling of natural processes, project of the Student Grant Agency of the Czech Technical University in Prague
  No. SGS14/206/OHK4/3T/14
- Application of advanced supercomputing methods for mathematical modeling of natural processes, project of the Student Grant Agency of the Czech Technical University in Prague
  No. SGS17/194/OHK4/3T/14
- Development and application of advanced methods for mathematical modeling of natural and industrial processes using high-performance computing, project of the Student Grant Agency of the Czech Technical University in Prague
  No. SGS20/184/OHK4/3T/14
- Investigation of shallow subsurface flow with phase transitions, project No. 17-06759S of the Czech Science Foundation
- Multiphase flow, transport, and structural changes related to water freezing and thawing in the subsurface, project no. 21-09093S of the Czech Science Foundation
- Development and Validation of Porous Media Fluid Dynamics and Phase Transitions Models for Subsurface Environmental Application, project of Czech Ministry of Education, Youth and Sports Kontakt II LH14003
- Computational Models and Experimental Investigation of Fluid Dynamics, Mass Transfer and Transport, and Phase Transitions in Porous Media for Environmental Applications, project of Czech Ministry of Education, Youth and Sports Inter-Excellence LTAUSA19021
- HPC-EUROPA3 (INFRAIA-2016-1-730897), with the support of the European Council Research Innovation Action under the H2020 Programme
- A New Inverse Theory for Joint Parameter and Boundary Conditions Estimation to Improve Characterization of Deep Geologic Formations and Leakage Monitoring, National Science Foundation Hydrologic Sciences Program Award 1702060

Author's declaration

I confirm having prepared the thesis by my own and having listed all used sources of information in the bibliography.

Prague, October 31 2022 Jakub Solovský

Contents

Nomenclature

Acronyms

NZ	Non-zero entries of the matrix, page 46
`MPI`	Message passing interface, page 45
BDDC	Balancing Domain Decomposition by Constraints, page 51
BiCGStab	Biconjugate gradient stabilized method, page 46
CSC	Compressed sparse column, page 45
CSR	Compressed sparse row, page 45
DNAPL	Dense non-aqueous phase liquid, page 22
DOF	Degree of freedom, page 61
EOC	Experimental order of convergence, page 65
IC	Incomplete Cholesky factorization, page 46
ILU	Incomplete LU factorization, page 46
MHFEM	Mixed-hybrid finite element method, page 25
NAPL	Non-aqueous phase liquid, page 8
PCG	Preconditioned conjugate gradient method, page 46
REV	Representative elementary volume, page 9
RTN	Raviart-Thomas-Nédélec space, page 28

Greek letters

α^{vG}	van Genuchten parameter for capillary pressure model, page 15	$[\mathbf{Pa}^{-1}]$
ϵ^{vG}	Mualem parameter for relative permeability model, page 17	$[-]$
γ^{vG}	Mualem parameter for relative permeability model, page 17	$[-]$
λ^{BC}	Brooks-Corey parameter for capillary pressure model, page 15	$[-]$
λ_α	Mobility of phase α, page 13	$[\mathbf{Pa}^{-1}\ \mathbf{s}^{-1}]$

μ	Dynamic viscosity, page 11	$[\mathrm{Pa\ s}]$
ν	Kinematic viscosity, page 11	$[\mathrm{m^2\ s^{-1}}]$
Φ	Porosity, page 9	$[-]$
ψ	Hydraulic head, page 12	$[\mathrm{m}]$
ψ_p	Pressure head, page 12	$[\mathrm{m}]$
ψ_z	Elevation head, page 12	$[\mathrm{m}]$
ρ	Density, page 11	$[\mathrm{kg\ m^{-3}}]$
σ	Surface tension, page 14	$[\mathrm{J\ m^{-2}}]$
τ_α	Tortuosity of phase α, page 20	$[-]$
θ	Contact angle, page 14	$[\mathrm{rad}]$

Latin letters

Re	Reynolds number, page 11	$[-]$
Sh	Sherwood number, page 22	$[-]$
$\mathbb{D}_\alpha^\kappa$	Diffusion and dispersion tensor of component κ in phase α, page 20	$[\mathrm{m}]$
$\mathbb{K}$	Absolute permeability tensor, page 11	$[\mathrm{m^2}]$
$\tilde{\mathbb{K}}$	Hydraulic conductivity tensor, page 12	$[\mathrm{m\,s^{-1}}]$
$\boldsymbol{g}$	Gravity vector, page 11	$[\mathrm{m\ s^{-2}}]$
$\boldsymbol{u}$	Macroscopic velocity, page 11	$[\mathrm{m\ s^{-1}}]$
a_L	Longitudinal dispersivity, page 20	$[\mathrm{m}]$
a_T	Transverse dispersivity, page 20	$[\mathrm{m}]$
$a_{\alpha\beta}$	Specific interfacial area separating phases α and β, page 21	$[\mathrm{m^{-1}}]$
$C_{\beta,s}^\kappa$	Saturated concentration of component κ in phase β, page 21	$[\mathrm{kg\ m^{-3}}]$
C_β^κ	Concentration of component κ in phase β, page 21	$[\mathrm{kg\ m^{-3}}]$
C_H	Gas specific constant in Van't Hoff equation, page 24	$[K]$
d_{50}	Mean size of the grains, page 22	$[\mathrm{m}]$
D_α^κ	Molecular diffusion of component κ in phase α, page 20	$[\mathrm{m^2\ s^{-1}}]$
$K_{\alpha\to\beta}^\kappa$	Mass transfer rate coefficient of component κ from phase α to β, page 21	$[\mathrm{m\ s^{-1}}]$
$k_{\alpha\to\beta}^\kappa$	Mass transfer coefficient of component κ from phase α to β, page 22	$[\mathrm{s^{-1}}]$
K_H	Henry's law constant, page 23	$\left[\mathrm{mol\ m^{-3}\ Pa^{-1}}\right]$

$k_{r\alpha}$	Relative permeability of phase α, page 13	[−]
m^{vG}	van Genuchten parameter for capillary pressure model, page 15	[−]
M_g	Molar mass of gass, page 23	$\left[\mathrm{kg\ mol^{-1}}\right]$
n^{vG}	van Genuchten parameter for capillary pressure model, page 15	[−]
p	Pressure, page 11	[Pa]
p_c	Macroscopic capillary pressure, page 15	[Pa]
p_c^m	Microscopic capillary pressure, page 14	[Pa]
p_d	Entry pressure, page 15	[Pa]
$Q^\kappa_{\alpha\to\beta}$	Mass flux of component κ from phase α to β, page 21	$[\mathrm{kg\ m^{-3}\ s^{-1}}]$
S_α	Saturation of phase α, page 12	[−]
S_c	Critical gas saturation, page 17	[−]
$S_{\alpha r}$	Residual saturation of phase α, page 15	[−]
S^e_α	Effective saturation of phase α, page 15	[−]
T	Temperature, page 24	$[K]$
X^κ_α	Mass fraction of component κ in phase α, page 20	[−]

Numerical methods and schemes

$\|\cdot\|$	Euclidean norm, page 11
$\|\cdot\|_d$	d-dimensional Lebesgue measure, page 29
$\#$	Number of elements in the set, page 28
$\mathcal{A}$	Conservative flux coefficient in the general formulation of the problem, page 26
$\mathcal{D}$	Diffustion coefficient in the general formulation of the problem, page 26
$\mathcal{F}$	Sink/source coefficient in the general formulation of the problem, page 26
$\mathcal{M}$	Mobility coefficient in the general formulation of the problem, page 26
$\mathcal{N}$	Time dependency coefficient in the general formulation of the problem, page 26
$\mathcal{R}$	Reaction coefficient in the general formulation of the problem, page 26
$\mathcal{U}$	Non-conservative flux coefficient in the general formulation of the problem, page 26
$\mathcal{W}$	External conservative forces coefficient in the general formulation of the problem, page 26
$\mathcal{Z}$	Vector of unknown functions in the general formulation of the problem, page 26
Δt_k	Time step between the times t_{k+1} and t_k , page 33 [s]

$\delta_{E,F}$	Kronecker delta symbol, page 28
ϵ	Tolerance for the relative norm of the residual for the iterative methods, page 62
$\mathcal{E}_h$	Set of all mesh sides, page 28
$\mathcal{E}_K$	Set of all mesh sides of an element K, page 28
$\mathcal{K}_h$	Set of all mesh elements, page 28
$\mathfrak{R}$	Set of real numbers, page 25
$\mathbb{A}$	Matrix of the linear problem, page 38
$\mathbb{A}_i$	Part of the matrix $\mathbb{A}$ corresponding to i-th subdomain, page 50
$\mathbb{C}_i$	Matrix defining coarse degrees of freedom in i-th subdomain, page 52
$\mathbb{I}$	Identity matrix, page 52
$\mathbb{R}_i$	Restriction matrix to i-th subdomain, page 50
$\mathbb{R}_{C_i}$	Restriction matrix of coarse degrees of freedom to i-th subdomain, page 54
$\mathbb{S}$	Schur complement matrix, page 49
$\mathbb{S}_i$	Part of the Schur complement matrix corresponding to i-th subdomain, page 50
$\mathbb{S}_{C_i}$	Part of the coarse problem matrix corresponding to i-th subdomain, page 53
$\mathbb{S}_C$	Coarse problem matrix, page 54
$\mathbb{W}_i$	Weight matrix on i-th subdomain, page 52
Ω	Computational domain, page 25
Ω_i	i-th subdomain in the domain decomposition approach, page 48
$\partial\Omega$	Boundary of the computational domain, page 27
Φ_i	Coarse basis functions on i-th subdomain, page 53
Φ_i^*	Adjoint coarse basis functions on i-th subdomain, page 53
Θ	Zero matrix, page 32
$\boldsymbol{\omega}$	Basis function of the Raviart-Thomas-Nédélec space, page 28
$\boldsymbol{n}(\boldsymbol{x})$	Outward unit normal at the point $\boldsymbol{x}$, page 27
$\boldsymbol{q}_i$	Conservative flux, page 27
$\boldsymbol{v}_i$	Conservative velocity, page 27
$\boldsymbol{x}$	Spatial coordinate, page 25
$\|\|E_{h,g}\|\|_p$	Error of the numerical solution on mesh with size h in p-norm, page 65
d	Dimension of the problem, page 25

$D_0(\mathcal{K}_h)$	Set of element-wise constant functions on elements $\mathcal{K}_h$, page 30	
h	Mesh size, page 61	[m]
t	Time, page 25	[s]
Z_j	j-th unknown function in the general formulation of the problem, page 25	
EI	Variant of the numerical method using exact integration to compute the entries of local coefficient matrices, page 40	
ML	Variant of the numerical method using mass lumping to compute the entries of local coefficient matrices, page 40	

Subscripts

α	Phase, page 12
ℓ	Liquid, page 92
g	Gas, page 92
n	Non-wetting phase, page 14
w	Wetting phase, page 14

Superscripts

Γ	Interface in the iterative substructuring methods, page 49
κ	Component, page 20
C	Coarse porous medium, page 18
F	Fine porous medium, page 18
I	Interior of the subdomains in the iterative substructuring methods, page 49
k, $k+1$	Time levels, page 33
upw	Quantity obtained by the upwind technique, page 34

1 Introduction

The main topic of this thesis is the mathematical modeling of two-phase compositional flow in porous media. During the past decades, the modeling of these processes gained substantial attention among researchers in many fields, for example, ecological, medical, petrochemical, and industrial applications. The mathematical model can provide valuable insights that would not be possible to obtain experimentally and allows for overcoming the limitations arising from the temporal and spatial scale of problems. The significant advantage of the mathematical model over the experiment is the possibility to easily modify the settings of the problem to investigate the impact of changes in parameters and run the computational studies to the extent that would not be possible to carry out in the experiments.

In this work, we focus on the problems related to CO_2 geological sequestration which is a promising technique that can be used to reduce the emissions of greenhouse gases [77]. However, its employment provides many challenges, namely: selecting a suitable reservoir, evaluating the safety on the long-term scale, and mitigating the risks of leakage. Mathematical models of possible scenarios in the considered sites can provide valuable insights that can help in understanding the processes occurring both during the sequestration and on the long-term scale. A complete model comprising all the relevant processes on a large spatial and temporal scale covering the area of hundreds of square kilometers and depths of thousands of meters for hundreds of years would be extremely complex. As such, the model is usually divided into smaller ones focusing on a selected phenomenon only. The main advantage of this division is the possibility of a more detailed investigation with a higher temporal and spatial resolution than it would be possible in the complete model.

Two such problems investigated in close collaboration with Center for Experimental Study of Subsurface Environmental Processes, Colorado School of Mines, Golden, Colorado are considered in this work. One is motivated by CO_2 leakage from a deep aquifer and focuses on the processes occurring in the shallow subsurface. The CO_2 injected into the aquifer in the supercritical state can reach the shallow subsurface as gas or dissolved in water. Gas can accumulate below low-permeable features, re-dissolve into groundwater, and be transported further from the original leakage point.

The other problem is related to a far-field brine leakage from the reservoir. The pressure buildup as a result of sequestration can reach further from the injection point than the head of the injected CO_2 and can lead to faults in the confining layers of the aquifer. Due to increased pressure, the brine residing originally in the aquifer can migrate to the surface.

1.1 State of the art

In this work, we focus on the continuum approach to the porous media where the physical quantities are averaged over representative volume and the processes are described by a system of highly nonlinear partial differential equations and constitutive algebraic relations. An overview of models ranging from a single-phase system to more complex multi-phase compositional systems can be found, for example, in [10, 11, 62].

1.1.1 Numerical methods

Numerical methods are a necessary tool for solving the system of equations arising from the description of two-phase compositional flow in porous media as the analytical solutions are known for special simplified cases only that are too restrictive for most of the applications. For the problem of two-phase flow without capillarity, the solution was found in [21]. In [76], the semi-analytical solution of the problem including capillarity was presented for the 1D and 2D cases of incompressible phases without gravity. This solution was expanded to comprise also the 3D case in [42].

To get at least an approximation of the solution in more complicated cases, a broad range of methods was designed employing various spatial and temporal discretizations. For the two-phase flow problems, finite difference methods were proposed in [32], finite volume methods in [36] and finite elements method in [38]. The methods based on the mixed-hybrid finite element (MHFEM) approach were proposed in [44, 66].

For more complex scenarios of two-phase compositional flow problems that to a certain extent include thermodynamical models, the method employing the finite difference discretization was presented in [26]. The numerical scheme based on the mixed-hybrid finite element discretization was proposed in [65]. An alternative formulation of the thermodynamical model was used in [89].

Methods suitable for the range of problems described in this work based on various spatio-temporal discretizations were implemented in both commercial codes such as `COMSOL Multiphysics`, `TOUGH3` [69] or `FEFLOW` [30], and open source codes such as `DUNE` [9].

1.1.2 Parallel approach

Serial computations with a higher spatial resolution can become too computationally expensive, therefore, the need to speed up the computations arises. A widely used approach to parallel implementation of numerical schemes is based on the domain decomposition [110]. The main idea behind these methods is to split the original problem into coupled problems on smaller subdomains. Based on the division into subdomains, the domain decomposition methods can be divided into two main groups: with overlapping and non-overlapping subdomains. For the overlapping subdomains methods that originate from the Schwartz alternating method, we refer to [33, 34, 35, 117]. In this work, we employ the approach of non-overlapping subdomains, also referred to as the iterative substructuring. Methods of this approach use the division of the degrees of freedom into the interior and interface ones and reduce the original problem into the much smaller one formulated on the interface [93]. For linear systems, the interface problem corresponds to the reduction of the original system to the Schur complement system which is usually solved iteratively, without its explicit construction. Without a preconditioner, the convergence rate of the iterative solver deteriorates with the increasing number of subdomains [110], thus limiting the applicability of the unpreconditioned solver for larger problems. Several types of preconditioners have been designed to improve the convergence of the iterative solvers for this type of problems [16, 17, 18, 19].

In this work, we use the balancing domain decomposition by constraints (BDDC) preconditioner that was introduced in [31] for problems with a symmetric positive definite matrix. It

was extended to the advection-diffusion problems with nonsymmetric matrices in [114] with the implementation based on the change of basis without forming the coarse problem explicitly. In [118], BDDC with explicit coarse basis functions and the coarse problem was applied to the linear system arising from the Euler equations, and the concept of adjoint coarse basis functions was introduced. In [54], the method was applied to nonsymmetric saddle-point problems arising from a linearization of the stationary Navier–Stokes equations and this approach was extended in [55] to a multilevel method. Extensions of the BDDC method towards flow in porous media include [105, 106, 119] while [108, 112] consider the mixed-hybrid finite element discretization of the problem.

1.2 Research goals

The main goal of this work is to investigate the evolution of a two-phase mixture of gas and water in a shallow subsurface. Due to the complexity of the problem this main topic is divided into several tasks comprising the various aspects of the problem. The following goals of this work are pursued:

- To formulate the mathematical model of the two-phase compositional flow in porous media with interphase mass transfer.
- To propose a numerical method suitable for solving the system of governing equations of such problems.
- To implement and verify the numerical method. The accuracy and convergence properties can be explored using the benchmark problems in homogeneous porous media with a known solution. For problems in heterogeneous porous media, where the exact solution is not known, the reference solution available in the literature can be used.
- To implement a parallel variant of the method and demonstrate the performance on the selected benchmark problems to ensure that the implementation preserves a good parallel efficiency for using a higher number of CPU cores.
- To study the impact of the mass transfer models on the two-phase CO_2 evolution in the shallow subsurface using experimental data and the results of the numerical model.
- To study component transport in highly heterogeneous porous media in scenarios motivated by brine leakage problems induced by CO_2 sequestration.

1.3 Content of this thesis

The thesis is organized as follows.

- In Chapter 1, we state the motivation for our research and present the state of the art.
- In Chapter 2, we present the continuum approach to describe flow, transport, and interphase mass transfer in porous media and formulate the mathematical model describing the problems considered in this work.
- In Chapter 3, we propose a numerical scheme based on the MHFEM approach suitable for solving problems described by the mathematical model presented in Chapter 2.

- In Chapter 4, the implementation of the proposed numerical scheme in `C++` is presented focusing on the parallel implementation based on BDDC.
- In Chapter 5, the numerical scheme is benchmarked to verify the convergence properties of the method. The problems in this work are motivated by the laboratory experiments and focus on processes in heterogeneous porous media, therefore, the accuracy of the numerical solution at the material interfaces is studied in detail.
- In Chapter 6, the efficiency of the parallel implementation is studied. The weak and strong scaling tests are performed and several approaches to solving unsteady problems are compared.
- In Chapter 7, the CO_2 evolution in shallow subsurface is investigated using the experimental data and numerical simulations, emphasizing the impact of the mass transfer model describing the CO_2 dissolution and exsolution.
- In Chapter 8, the numerical scheme is used to investigate the tracer transport in a highly heterogeneous porous medium. This problem is motivated by the laboratory scenarios mimicking the far-field brine leakage.
- In Chapter 9, we conclude the thesis, summarize the achieved results and propose future research goals.

1.4 Achieved results

- We formulated a mathematical model describing the two-phase compositional flow in porous media with the kinetic model for interphase mass transfer. This model is a result of combining the existing models for the considered phenomena available in the literature.
- We developed a numerical method based on the mixed-hybrid finite elements spatial discretization with semi-implicit discretization in time for both structured and unstructured meshes in 2D and 3D. The method can deal with degenerate cases where only one of the phases is present in certain regions and also phase appearance and disappearance as a result of the interphase mass transfer. The numerical scheme was published in [43].
- The numerical method was implemented in `C++` in both serial and parallel variants. The serial variant is suitable for computations on rather coarse meshes, typically in parameter sensitivity studies where many independent computations are required, while the parallel variant of the method is needed for computations with higher spatial resolution.
- The convergence of the method was verified on a benchmark problem with a known semi-analytical solution showing that the method is convergent with the first order of accuracy. We also showed that the method can correctly capture the flow in heterogeneous porous media. The scaling tests showed a good efficiency of the parallel implementation up to a thousand of CPU cores. The convergence studies were published in [43, 102] and the parallel implementation was published in [101, 104]
- Using the measurements from the laboratory-scale experiments, we showed that employing the correct mass transfer model is necessary to capture the behavior observed in the experiments. While the equilibrium approach is sufficient for the quasi-one-dimensional cases, the kinetic model is needed in the case of a two-dimensional flow field and when larger distances are considered. Our findings also indicate that the dynamics of the dissolution

and exsolution processes differ and thus different transfer rate coefficients should be used for modeling dissolution and exsolution processes. The results were published in [46, 103].

- Motivated by the laboratory experiments that focused on brine leakage from deep aquifers induced by the CO_2 sequestration, we showed that our numerical scheme is capable of solving tracer transport problems in highly heterogeneous porous media mimicking the geological settings of the existing site. The experimental and modeling results using a different numerical method were published in [4].

1.5 Author's contribution in journal papers

The author's contribution to the impacted publications related to this thesis is given in the following.

- Solovský, J., Fučík, R., Plampin, M. R., Illangasekare, T. H., and Mikyška, J. (2020). Dimensional effects of inter-phase mass transfer on attenuation of structurally trapped gaseous carbon dioxide in shallow aquifers. *Journal of Computational Physics (IF 4.502)*, 405:109178.

  ▷ Jakub Solovský was the leading author, implemented the numerical method, and ran all the computations. He wrote the main body of the paper.

- Solovský, J., Fučík, R., and Šístek, J. (2022). BDDC for MHFEM discretization of unsteady two-phase flow in porous media. *Computer Physics Communications (IF 5.297)*, 271:108199.

  ▷ Jakub Solovský was the leading author, implemented the method in parallel, and ran all the benchmarks. He wrote the main body of the paper.

- Fučík, R., Klinkovský, J., Solovský, J., Oberhuber, T., and Mikyška, J. (2019). Multidimensional mixed–hybrid finite element method for compositional two-phase flow in heterogeneous porous media and its parallel implementation on GPU. *Computer Physics Communications (IF 5.297)*, 238:165–180.

  ▷ Jakub Solovský was co-author. He implemented the method in serial and ran the convergence benchmarks.

- Askar, A. H., Illangasekare, T. H., Trautz, A., Solovský, J., Zhang, Y., and Fučík, R. (2021). Exploring the Impacts of Source Condition Uncertainties on Far-Field Brine Leakage Plume Predictions in Geologic Storage of CO2: Integrating Intermediate-Scale Laboratory Testing With Numerical Modeling. *Water Resources Research (IF 6.391)*, 57(9):e2021WR029679.

  ▷ Jakub Solovský was co-author. He participated in collecting and analyzing the experimental data. He helped with the development and verification of the mathematical model.

1.6 Future work

Beyond the presented thesis and the achieved results, there remain several challenging tasks that need to be further explored. These include:

- To expand the mathematical model to comprise the effects neglected in the current model to make it suitable for a wider range of problems. The results presented in this work showed that even small fluctuations in temperature can significantly impact the dissolution and exsolution processes. In the field, the temperature can change significantly in time which is to an extent also included in this work and also within a domain, therefore, the non-isothermal model is needed for reliable long-term predictions. Although Henry's law gives good results for the range of temperature and pressure considered in this work it has limitations for higher temperature and pressure values that would be encountered in the deeper aquifers. In addition, the employment of Henry's law also prohibits using the model for more complex multicomponent systems.
- To overcome the limitations of the current parallel implementation. The drop in efficiency around approximately a thousand CPU cores is typical for the two-level BDDC approach employed in this work [107]. To overcome this limitation, the multilevel approach was proposed [5, 73, 107, 113] posing additional challenges for an efficient parallel implementation.

Mathematical model of flow and transport in porous media 2

In this chapter, we present the mathematical model describing two-phase flow in porous media with component transport and interphase mass transfer. First, we define the porous medium for the purposes of this work and the physical quantities needed to describe this medium and the studied processes within it. Then, we formulate the governing equations for the respective processes and comment on additional assumptions and simplifications considered in this work. The notation used in this chapter is mainly adopted from [8, 10, 11].

2.1 Porous media

A porous medium is a body composed of a persistent solid part also called the solid matrix and the remaining void space also called pore space or pores that can be filled with one or more fluids. This is a rather general definition and any material consisting of void space within could be considered a porous medium under this definition. However, the range of possible materials and their properties would be too wide to be dealt with by a single mathematical model. Therefore, to formulate the mathematical model of flow in porous media in the cases relevant for this work, additional assumptions are added [28]:

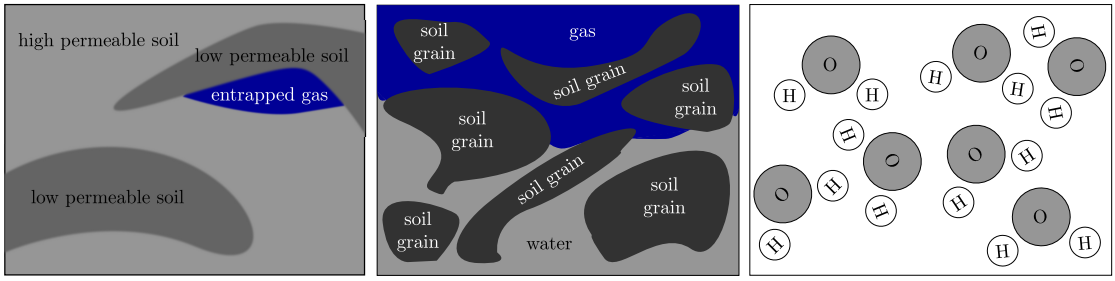
1. The void space of the porous media is interconnected.

2. The dimensions of the pores are large in comparison with the mean free path length of the fluid molecules.

3. The dimensions of the pores are small enough so that the fluid flow is governed by adhesive forces at fluid-solid interfaces and cohesive forces at fluid-fluid interfaces in the case of a multi-phase system.

The first assumption eliminates isolated regions where no flow can occur. The second assumption allows for describing the fluids occupying the pores using the continuum approach, in detail described later in Section 2.1.2. The last assumption excludes special cases such as a network of pipes from the definition of porous media for the purposes of this work. Examples of porous media satisfying these conditions are sand, soil, or organic tissue such as kidneys or myocardium.

In addition to the three assumptions above, in this work, we also assume that the solid matrix is rigid and does not change over time. Generally, the solid matrix can be affected by both



(a) Macroscale (~ 10 m). (b) Microscale ($\sim 10^{-3}$ m). (c) Molecular scale ($\sim 10^{-9}$ m).

Figure 2.1: Three different scales the porous medium is considered at.

physical and chemical processes taking place in porous media. For example, fractures in porous rocks can open or close as a result of applied pressure, changing the pore space configuration and the overall pore space volume. Additionally, the products of chemical reactions can precipitate in the pores and, as a result, can partially or even completely clog certain pores. These changes in the solid matrix are, however, beyond the scope of this work. For details, we refer, for example, to [53].

2.1.1 Phase

A phase is defined as a chemically homogeneous portion of the system that is separated from the rest of the system by a definite physical boundary [10, 11]. Based on the physical properties of the fluids within the pores, a phase can consist of one or more fluids. For example, a mixture of gases in the system forms a single phase as the gases are always completely miscible. On the other hand, water and oil form a two-phase system with a definite physical boundary between water and oil.

In this work, we are interested mainly in the two-phase systems with one phase being water. In the benchmark problems in Chapters 5 and 6, the other phase is referred to as non-aqueous phase liquid (NAPL) representing for example oil. In the problems arising from laboratory experiments in Chapter 7, the other phase is a gaseous CO_2.

2.1.2 Continuum approach

An important characteristic of a porous medium is that it can be considered on different scales. The three usually considered scales: macro, micro, and molecular are illustrated in Figure 2.1. In this work, we are interested in the modeling at the macroscale illustrated in Figure 2.1a, i.e., the dimensions of a domain of interest are in the range of meters. On this scale, we can identify the regions consisting of materials with different physical characteristics such as average grain size or dominant grain shape but the position of the individual solid grains cannot be determined and, therefore, the exact configuration of the void space is not known.

Individual grains of solid material and the pore structure determined by their positions are resolved at the microscale which is illustrated in Figure 2.1b. In addition, on this scale, we can identify which fluid occupies a given pore and track the fluid-fluid interfaces. Dimensions of the domain of interest are in the range of millimeters.

Magnifying even further to the molecular scale illustrated in Figure 2.1c with dimension in the range of 10^{-9} m, individual fluid molecules and their interactions are resolved.

The processes in the porous medium are controlled by physical and chemical effects on all three scales. The fluid properties such as the viscosity, diffusion coefficients, etc., are determined on the molecular scale by the properties of the individual fluid molecules and their interactions.

The flow through a porous medium is controlled by the configuration of the pore space at the microscale. At the macroscale, the configuration of regions with different properties such as a mean grain size controls the larger-scale flow patterns and can result, for example, in preferential flow paths through the domain. At the microscale, the flow through individual pores can be described by the Navier-Stokes equations with appropriate boundary conditions on the solid material surface and the computational domain boundary.

At the macroscale, however, this configuration is not known and, therefore, this approach is not suitable. Moreover, such numerical simulations in a domain with typical dimensions in meters and with solid material grains smaller than one millimeter, which are usual dimensions for the problems of interest, would be too computationally demanding. These limitations lead to the introduction of a different continuum approach at the macroscale. This approach is based on the representative elementary volume (REV) strategy [10, 11]. Instead of the microscopic physical quantity at a given point such as the fluid velocity or which fluid is present at a given point, an average of the microscopic quantity over the REV is considered.

The selection of a REV is the key component of this approach. It should be sufficiently large to statistically estimate all the relevant physical quantities and small enough to be considered as a negligible portion of the overall domain at the macroscale [12]. The main advantage of this approach is that it leads to macroscopic equations that are independent of the exact pore space configuration at the microscale that is not known in the applications of the macroscale model and are formulated using only the measurable statistical properties of the porous medium and fluids within the pores.

2.1.3 Porosity

Here, the continuum approach based on averaging over the REV is demonstrated on one of the basic macroscopic property of the porous medium: porosity. The porosity describes the ratio of the void space within a given volume of porous material to this total volume. To mathematically define the porosity, first, at the microscopic level, the void space indicator function $\gamma(\boldsymbol{x})$ on a domain Ω is defined by

$$\gamma(\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x} \in \text{void space}, \\ 0 & \boldsymbol{x} \in \text{solid matrix}, \end{cases} \quad \forall \boldsymbol{x} \in \Omega. \tag{2.1}$$

At the macroscale, the porosity at position $\boldsymbol{x_0}$ with respect to volume V is then defined by

$$\Phi(\boldsymbol{x_0}) = \frac{1}{V} \int_V \gamma(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{2.2}$$

where the volume V is a ball with a center at $\boldsymbol{x}_0$ and radius r as shown in Figure 2.2. For the REV approach to be valid, the volume V must be selected such that the value of porosity $\Phi(\boldsymbol{x}_0)$ does not depend on the exact value of r. If we study the dependency of the porosity on the value of r, the behavior illustrated in Figure 2.3 is usually observed.

For values of r much smaller than the pore and grain sizes, the porosity is either 1 or 0 based on whether the position $\boldsymbol{x}_0$ is located within the solid matrix or in the pore space and small changes in r can lead to significant changes in the porosity value. As r increases beyond a certain threshold value r_{micro} and V covers more pores and solid grains, the porosity becomes almost constant. When increasing r further beyond another threshold value r_{macro}, the volume V can contain larger scale heterogeneity, such as layers of different porous materials, and the porosity can change again.

If such values r_{micro} and r_{macro} exist, the radius for REV V can be then selected within the range

$$r_{micro} \ll r \ll r_{macro}, \tag{2.3}$$

otherwise, the REV cannot be established and this continuum approach cannot be used [56]. In this work, we assume that the assumptions for the establishment of REV are satisfied and in the following text, we present the governing equations of flow in porous media based on this continuum approach.

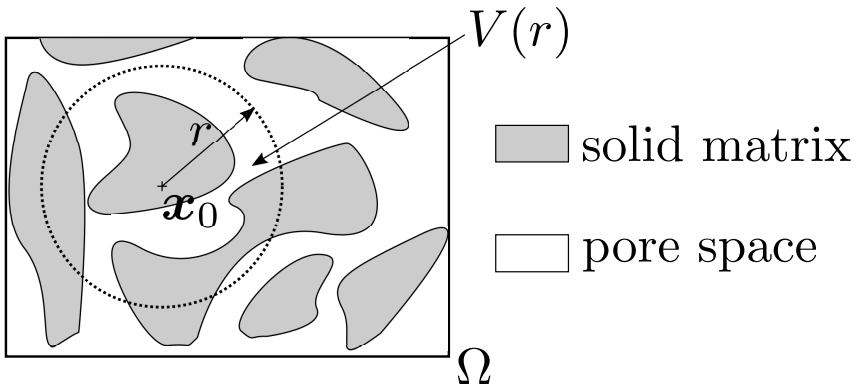


Figure 2.2: Illustration of the averaging REV $V(r)$.

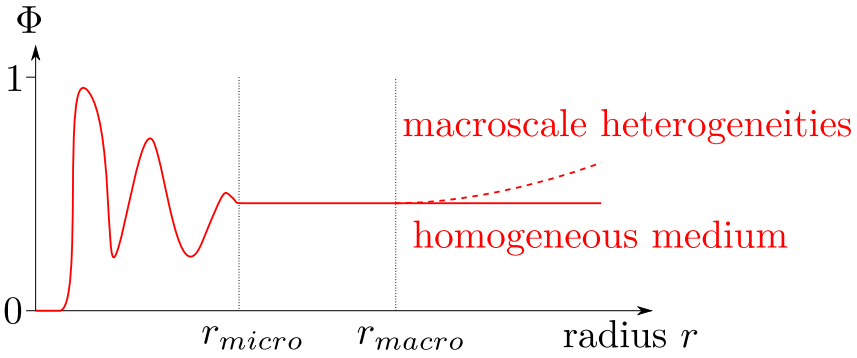


Figure 2.3: Porosity Φ as a function of the averaging volume radius r.

2.2 Single-phase flow

The governing equations for a single-phase system, when pores are filled with a single fluid or multiple completely miscible fluids, for example, a mixture of gases are summarized in this section.

2.2.1 Mass balance

The macroscopic fluid mass conservation law is given by

$$\frac{\partial(\Phi\rho)}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = f, \tag{2.4}$$

where the quantities have the following meaning:

$\Phi(\boldsymbol{x})$	$[-]$	is the porosity of the porous medium defined in Section 2.1.3. In the case of a heterogeneous porous medium, the porosity Φ is a function of the position $\boldsymbol{x}$.
$\rho(t, \boldsymbol{x})$	$[\mathrm{kg\ m^{-3}}]$	is the density of the fluid. For incompressible fluids, the density is constant. For compressible fluids, the density is given by the equation of state, for example, the ideal gas state equation.
$\boldsymbol{u}(t, \boldsymbol{x})$	$[\mathrm{m\ s^{-1}}]$	is the macroscopic fluid velocity. At the microscopic level, the flow occurs through pores only and the average velocity $\boldsymbol{u}/\Phi$ is observed.
$f(t, \boldsymbol{x})$	$[\mathrm{kg\ m^{-3}\ s^{-1}}]$	is the sink or source term.

2.2.2 Darcy's law

At the macroscale, the momentum balance can be obtained by the local averaging [84, 116] or homogenization [64] of the Navier-Stokes equations describing the flow at the microscale. These procedures result in the expression

$$\boldsymbol{u} = -\frac{1}{\mu}\mathbb{K}(\nabla p - \rho \boldsymbol{g}), \tag{2.5}$$

where the quantities have the following meaning:

$p(t, \boldsymbol{x})$	$[\mathrm{Pa}]$	is the fluid pressure.
$\boldsymbol{g}$	$[\mathrm{m\ s^{-2}}]$	is the gravity vector.
$\mathbb{K}(\boldsymbol{x})$	$[\mathrm{m^2}]$	is the symmetric tensor of absolute permeability characterizing the solid matrix. In the case of a heterogeneous porous medium, tensor $\mathbb{K}$ is a function of position $\boldsymbol{x}$.
$\mu(t, \boldsymbol{x})$	$[\mathrm{Pa\ s}]$	is the dynamic viscosity of the fluid.

The one-dimensional variant of relation (2.5) was discovered experimentally by Henry Darcy in 1856 and (2.5) is named after him as the Darcy's law.

The Darcy's law is valid for the slow flow of a Newtonian fluid through a porous medium with a rigid solid matrix. The range of validity of the Darcy's law can be given in terms of the dimensionless Reynolds number Re $[-]$, for the porous media defined by

$$\mathrm{Re} = \frac{\delta}{\nu}\left\|\boldsymbol{u}\right\|, \tag{2.6}$$

where $\|\cdot\|$ denotes the Euclidean norm and the remaining quantities have the following meaning:

δ	$[\mathrm{m}]$	is the representative microscopic length characterizing the pore space. The value of δ is often selected as the mean grain diameter.
ν	$[\mathrm{m^2\ s^{-1}}]$	is the kinematic viscosity of the fluid.

The Darcy's law is valid for flows satisfying the condition $\mathrm{Re} < 1$. In some cases, the limit can be increased up to $\mathrm{Re} < 10$. This range of feasible Reynolds numbers covers most of the problems of flow in porous media, except, for example, the flow in the vicinity of high-rate pumping wells. For higher values of Re, the inertial effects play a more important role and can no longer be neglected. In this case, more complex expressions than (2.5) are needed to accurately describe

the flow, for details, see [10, 11]. However, such cases are beyond the scope of this work and we assume that equation (2.5) is valid for all the problems considered here.

2.2.3 Hydraulic head

The Darcy's law was in (2.5) given in terms of phase pressure p. It can be equivalently formulated using the hydraulic (piezometric) head ψ [m] defined by

$$\psi = \psi_z + \psi_p, \tag{2.7}$$

where the quantities have the following meaning:

ψ_z	[m]	is the elevation head characterizing the gravitational potential energy of the fluid arising from elevation.
ψ_p	[m]	is the pressure head characterizing the energy due to pore fluid pressure.

Using the hydraulic head the Darcy's law is given by

$$\boldsymbol{u} = -\tilde{\mathbb{K}}\nabla\psi, \tag{2.8}$$

where $\tilde{\mathbb{K}}$ [$\mathrm{m\,s^{-1}}$] is the hydraulic conductivity [10] related to absolute permeability by

$$\mathbb{K} = \tilde{\mathbb{K}}\frac{\mu}{\rho\,\|\boldsymbol{g}\|}. \tag{2.9}$$

2.3 Two-phase flow

In this section, we consider the extension of the single-phase flow case described in Section 2.2: a situation where the pore space is filled with two immiscible fluids. The phases are denoted by α and the quantities corresponding to the given phase are distinguished by the subscript. Only the two-phase case is considered in this work, however, the notation and the introduced physical quantities can be also used for the description of a general, multi-phase system consisting of three or more phases.

2.3.1 Saturation

For the formulation of the governing equations, the description of the phase distribution within a porous medium is needed. The saturation of phase α at the macroscale is defined similarly to the porosity in Section 2.1.3 by averaging the microscopic quantity. At the microscale, each point of the pore space is occupied by exactly one phase. Similarly to the void space indicator function $\gamma(\boldsymbol{x})$ in (2.1), the α-phase indicator function $\gamma_\alpha(t, \boldsymbol{x})$ in domain Ω is defined by

$$\gamma_\alpha(t,\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x} \in \text{phase } \alpha \text{ at time } t, \\ 0 & \text{otherwise}, \end{cases} \quad \forall \boldsymbol{x} \in \Omega, \tag{2.10}$$

that is, in addition to the spatial coordinate $\boldsymbol{x}$, also a function of time t. Unlike the solid matrix that is assumed constant in time, the distribution of phase α evolves in time.

The saturation S_α [−] at a given point $\boldsymbol{x}_0$ is then defined by using REV around this point

$$S_\alpha(t,\boldsymbol{x}_0) = \frac{\int_{REV}\gamma_\alpha(t,\boldsymbol{x})\mathrm{d}\boldsymbol{x}}{\int_{REV}\gamma(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}. \tag{2.11}$$

The saturation S_α indicates the ratio of the volume occupied by phase α to the overall volume of the pore space within REV. From the definition (2.11) follows the following properties of the saturation S_α

$$0 \leq S_\alpha \leq 1, \forall \alpha \in J, \qquad \sum_{\alpha \in J} S_\alpha = 1, \tag{2.12}$$

where J is the index set of all phases present within volume V.

2.3.2 Mass balance

With saturation defined by (2.11), the macroscopic mass conservation of phase α is expressed by

$$\frac{\partial(\Phi S_\alpha \rho_\alpha)}{\partial t} + \nabla \cdot (\rho_\alpha \boldsymbol{u}_\alpha) = f_\alpha, \tag{2.13}$$

where the quantities have the same meaning as in (2.4), their correspondence to each phase α is distinguished by the subscript.

2.3.3 Darcy's law

The macroscopic velocity $\boldsymbol{u}_\alpha$ is given by the extension of the single-phase Darcy's law

$$\boldsymbol{u}_\alpha = -\frac{1}{\mu_\alpha} \mathbb{K}_\alpha (\nabla p_\alpha - \rho_\alpha \boldsymbol{g}), \tag{2.14}$$

where, in contrast to (2.5), the permeability tensor $\mathbb{K}_\alpha$ depends also on the saturation of phase α and it is assumed that it can be split into two terms

$$\mathbb{K}_\alpha = k_{r\alpha}(S_\alpha)\mathbb{K}, \tag{2.15}$$

where $k_{r\alpha}(S_\alpha)$ $[-]$ is the relative permeability function and the absolute permeability tensor $\mathbb{K}$ is a function of the solid matrix properties only independent of the phase saturations that has the same meaning as in the single-phase case in (2.5). The relative permeability $k_{r\alpha}$ represents the fact that the flow of phase α through the pore space can be partially blocked by other phases occupying the pores. Models for $k_{r\alpha}$ and their properties are in detail discussed later in Section 2.3.6.

For clarity, the mobility λ_α of the phase α is introduced by

$$\lambda_\alpha = \frac{k_{r\alpha}}{\mu_\alpha}, \tag{2.16}$$

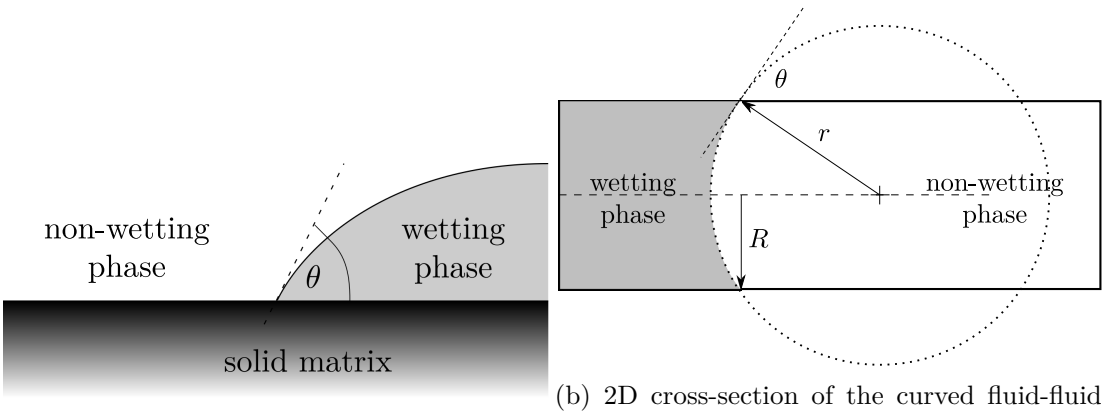
and the Darcy's law can be then written as

$$\boldsymbol{u}_\alpha = -\lambda_\alpha \mathbb{K} (\nabla p_\alpha - \rho_\alpha \boldsymbol{g}). \tag{2.17}$$

2.3.4 Capillarity

In addition to the forces arising from the pressure gradient governing the single-phase flow, the sharp interface between the phases at the microscale gives rise to the capillary forces that play an important role in the two-phase flow. The imbalance between the adhesive and cohesive forces at the fluid-fluid interface leads to a curved shape of the interface. The imbalance between

the forces can be quantified by the surface tension σ [$\mathrm{J\,m^{-2}}$]. As illustrated in Figure 2.4a, the interaction of the adhesive (fluid-solid) and cohesive (fluid-fluid) forces results in a contact angle θ between the solid surface and the fluid-fluid interface. This contact angle θ allows for the classification of the two fluids at the interface. The fluid for which $\theta < \pi/2$ is called the wetting phase fluid (denoted by subscript w) and the other fluid is called the non-wetting phase fluid (denoted by subscript n). Whether a given fluid is wetting or non-wetting depends on the two fluids that are present at the interface. A given fluid can act as wetting in contact with one fluid and non-wetting with the other.



(a) Contact angle θ between a solid surface and fluid-fluid interface.

(b) 2D cross-section of the curved fluid-fluid interface with radius r and contact angle θ for the cylindrical pore with radius R.

Figure 2.4: Capillary effects at the microscale.

The curved interface at the microscale between the wetting (w) and non-wetting (n) phase is the result of the discontinuity in the microscopic pressure of the non-wetting phase p_n^m and the wetting phase p_w^m. This difference in the microscopic pressures is called microscopic capillary pressure p_c^m,

$$p_c^m = p_n^m - p_w^m \geq 0. \tag{2.18}$$

At the pore scale, it can be shown that for a cylindrical pore with radius R, illustrated in Figure 2.4b, the capillary pressure is given by [8]

$$p_c^m = \frac{2\sigma \cos\theta}{R}. \tag{2.19}$$

For most of the porous media, however, the pore shapes are more complex than cylinders. Moreover, at the macroscale, the exact geometry of the pores is not known and the continuum approach used does not allow for tracking the individual fluid-fluid interfaces within the pore space. Therefore, an equation such as (2.19) is not suitable for modeling the capillary pressure at the macroscale.

2.3.5 Macroscopic capillary pressure

To resolve the capillary pressure at the macroscale, analogically to (2.18), the macroscopic capillary pressure is defined by the difference between the macroscopic pressures of the non-wetting (p_n) and wetting (p_w) phases

$$p_c = p_n - p_w \geq 0. \tag{2.20}$$

The macroscopic capillary pressure p_c is in general a function of state variables such as the phase saturation, fluid composition, temperature, and also the interface area between the fluids and

between the fluid and solid matrix [59]. In this work, we do not consider all these dependencies. Instead, we use the standard approach [10, 11] for modeling the two-phase flow in porous media, where the capillary pressure p_c is assumed to be a function of the wetting phase saturation S_w only.

The relationships between the capillary pressure p_c and the wetting phase saturation S_w are based on theoretical considerations and are given in the form of empirical models, usually containing several parameters that are fitted to the experimental data. Two such widely used models are discussed later. Before we list the models, we discuss the general properties of the $p_c \leftrightarrow S_w$ relation and define additional quantities to describe the fluid distribution in porous media needed for a more convenient form of the $p_c(S_w)$ functions.

Effective and residual saturations

When a porous medium, initially fully saturated with the wetting phase, undergoes a draining process, the wetting phase saturation decreases and the wetting phase remains only in smaller pores, resulting in an increase in the capillary pressure. Finally, a certain value of the wetting phase saturation is reached and the saturation cannot be reduced by mechanical displacement by the non-wetting phase only. However, it can be further reduced, for example, by a phase transition such as evaporation. The amount of wetting phase that remains present in the pores is characterized by the wetting phase residual saturation denoted by S_{wr}. Similarly, a certain amount of the non-wetting phase remains within a porous medium during the imbition process and cannot be mechanically displaced by the wetting phase. Analogically, this amount is characterized by the value of non-wetting phase residual saturation denoted by S_{nr}.

For the description of flow in porous media, it is advantageous to exclude the fluids present in the pores quantified by the residual saturations and define the effective saturation S_α^e by

$$S_\alpha^e = \frac{S_\alpha - S_{\alpha r}}{1 - S_{nr} - S_{wr}}, \tag{2.21}$$

which describes only the portion of the volume occupied by the phase that is mobile and can be mechanically displaced from the porous medium by the other phase. By definition, the effective saturations S_α^e have the same properties as the phase saturations S_α given in (2.12).

Brooks-Corey

The Brooks-Corey model [20] for capillary pressure p_c as a function of the wetting phase effective saturation S_w^e is given by

$$p_c(S_w^e) = p_d {S_w^e}^{-\frac{1}{\lambda^{\mathrm{BC}}}}, \tag{2.22}$$

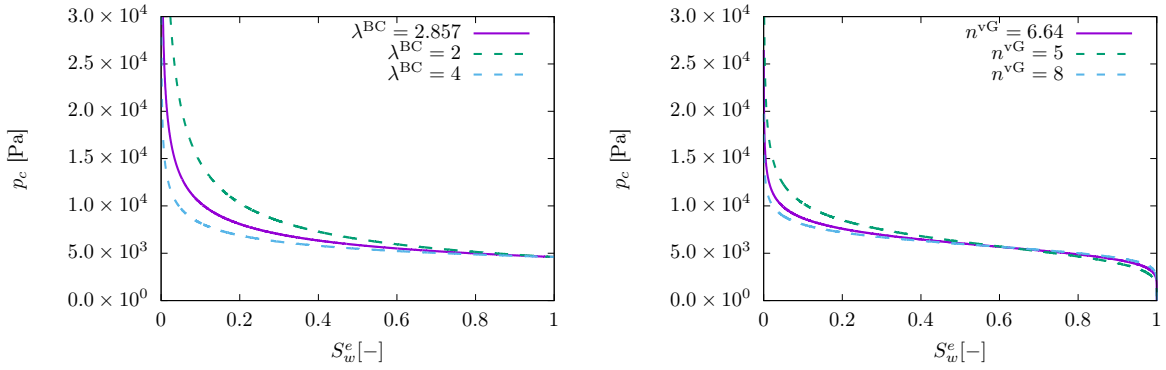
where $p_d\,[\mathrm{Pa}]$ is the entry pressure of the porous medium and $\lambda^{\mathrm{BC}}\,[-]$ is the model parameter characterizing the pore size distribution. The usual shape of the $p_c(S_w^e)$ functions for this model is illustrated in Figure 2.5a.

van Genuchten

The van Genuchten model [47] for capillary pressure p_c as a function of the wetting phase effective saturation S_w^e is given by

$$p_c(S_w^e) = \frac{1}{\alpha^{\mathrm{vG}}} \left({S_w^e}^{-\frac{1}{m^{\mathrm{vG}}}} - 1 \right)^{\frac{1}{n^{\mathrm{vG}}}}, \tag{2.23}$$

where $\alpha^{\mathrm{vG}}\left[\mathrm{Pa}^{-1}\right]$ is related to the entry pressure, $n^{\mathrm{vG}}\,[-]$ is an independent parameter, and m^{vG} is usually selected as $m^{\mathrm{vG}} = 1 - \frac{1}{n^{\mathrm{vG}}}$. The usual shape of the $p_c(S_w^e)$ functions for this model is illustrated in Figure 2.5b.



(a) Brooks-Corey (b) van Genuchten

Figure 2.5: The Brooks-Corey and van Genuchten models for capillary pressure. The solid line corresponds to Sand A used in benchmarks in Chapters 5 and 6 with the parameters given in Table 5.1. The dashed lines show the influence of parameters λ^{BC} and n^{vG}.

The significant difference between the two aforementioned models for capillary pressure is for values S_w^e close to 1, i.e., porous media fully saturated by wetting phase as follows from (2.22) and (2.23). The van Genuchten model has the following properties

$$\lim_{S_w^e \to 1-} p_c(S_w^e) = 0, \qquad \lim_{S_w^e \to 1-} \frac{\mathrm{d}p_c(S_w^e)}{\mathrm{d}S_w^e} = -\infty, \tag{2.24}$$

whereas for the Brooks-Corey model holds

$$\lim_{S_w^e \to 1-} p_c(S_w^e) = p_d, \qquad \lim_{S_w^e \to 1-} \frac{\mathrm{d}p_c(S_w^e)}{\mathrm{d}S_w^e} = -\frac{p_d}{\lambda^{BC}} S_w^{e\,-\frac{1}{\lambda^{BC}}-1}. \tag{2.25}$$

2.3.6 Relative permeability

The permeability tensor $\mathbb{K}_\alpha$ in (2.15) is split into two terms: $\mathbb{K}_\alpha = k_{r\alpha}(S_\alpha)\mathbb{K}$, where the relative permeability $k_{r\alpha}$ is assumed to be a function of the phase saturation and can be considered as a scaling factor satisfying the following condition

$$0 \leq k_{r\alpha} \leq 1. \tag{2.26}$$

In this section, we present two models for relative permeability $k_{r\alpha}$. These two models are in close relation to the two capillary pressure models presented in Section 2.3.5 that they are usually used together.

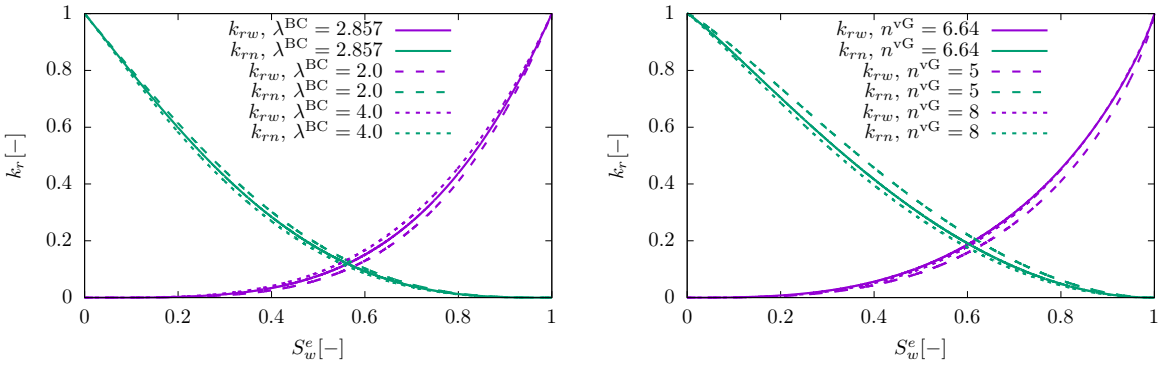
Burdine

The Burdine model [22] is usually used together with the Brooks-Corey model for capillary pressure and is given by

$$k_{rw}(S_w^e) = (S_w^e)^{\frac{2+3\lambda^{BC}}{\lambda^{BC}}}, \tag{2.27}$$

$$k_{rn}(S_n^e) = (S_n^e)^2 \left(1 - (1 - S_n^e)^{\frac{2+\lambda^{BC}}{\lambda^{BC}}}\right), \tag{2.28}$$

where λ^{BC} is the parameter of the Brooks-Corey model for the capillary pressure (2.22). The usual shapes of the $k_{rw}(S_w^e)$ and $k_{rn}(S_n^e)$ functions for this model are shown in Figure 2.6a.



(a) Burdine (b) Mualem

Figure 2.6: The Burdine and Muelaem models for relative permeability. The solid line corresponds to Sand A used in benchmarks in Chapters 5 and 6 with the parameters given in Table 5.1. The dashed lines show the influence of parameters λ^{BC} and n^{vG}.

Mualem

The Mualem model [80] is usually used together with the van Genuchten model for capillary pressure and is given by

$$k_{rw}(S_w^e) = (S_w^e)^{\epsilon^{\mathrm{vG}}} \left(1 - \left(1 - (S_w^e)^{\frac{1}{m^{\mathrm{vG}}}}\right)^{m^{\mathrm{vG}}}\right)^2, \tag{2.29}$$

$$k_{rn}(S_n^e) = (S_n^e)^{\gamma^{\mathrm{vG}}} \left(1 - (1 - S_n^e)^{\frac{1}{m^{\mathrm{vG}}}}\right)^{2m^{\mathrm{vG}}}, \tag{2.30}$$

where $\epsilon^{\mathrm{vG}} = \frac{1}{2}$ and $\gamma^{\mathrm{vG}} = \frac{1}{3}$ are usually selected and m^{vG} is the parameter of the van Genuchten model for the capillary pressure (2.23). The usual shapes of the $k_{rw}(S_w^e)$ and $k_{rn}(S_n^e)$ functions for this model are shown in Figure 2.6b.

2.3.7 Critical gas saturation

The relative permeability models presented in Section 2.3.6 were proposed for the case of the flow of two immiscible fluids. However, as reported by [90, 111], the concept of mechanically immobile residual saturation is not sufficient for modeling gas dissolution and exsolution processes, especially in cases, where no gas is initially present in the porous medium which is the case of scenarios studied later in Chapter 7. As observed in the experiments [90, 111], a certain threshold value of gas saturation has to be reached during exsolution from the other phase before the gas phase can become mobile. Such a value is referred to as the critical gas saturation denoted by S_c.

This threshold value cannot be described by a value of residual saturation S_{nr}. No gas initially present in a porous medium is represented by a zero value of S_{nr} which means the gas phase becomes mobile immediately once developed. Increasing the value of S_{nr} when the gas develops to make it immobile results in the change in effective saturation S_n^e: residual and effective saturations are interconnected by (2.21) and the saturation S_n in this redistribution process between S_{nr} and S_n^e must remain the same. As follows from the models presented in Section 2.3.5, such a change in the effective saturation would result in a change in the capillary pressure and it would mean non-uniqueness of the capillary pressure values.

To represent the behavior observed in the experiments in the model using effective saturation and the relative permeability from Section 2.3.6, the following modification of the relative

permeability function for the non-wetting phase was proposed in [103]

$$\widehat{k_{rn}}(S_n) = \begin{cases} 0, & \text{if } S_n < S_c, \\ k_{rn}(\frac{S_n - S_c}{1 - S_c}), & \text{otherwise,} \end{cases} \tag{2.31}$$

where k_{rn} is given by one of the models given in Section 2.3.6 and the value of critical gas saturation S_c is determined based on the experimental data. Details about finding the value of S_c are given later in Chapter 7 for several scenarios.

2.3.8 Fluid behavior at material interfaces

In this section, we focus on the interface between two materials with different saturation–capillary pressure relations. The material with a lower entry pressure is referred to as the coarse one and is denoted by superscript C. The other material is referred to as the fine one and is denoted by superscript F. Since there are no sinks or sources assumed at the interface, the normal component of the mass flux $\rho_\alpha \boldsymbol{u}_\alpha$ for each phase α is continuous across the interface.

As in [81], we assume that a mobile wetting phase is present on both sides of the interface from which follows the continuity of wetting phase pressure p_w across the interface. If the non-wetting phase is present on both sides of the interface, non-wetting phase pressure p_n is also assumed to be continuous. From the continuity of phase pressures follows the continuity of capillary pressure p_c which is defined by (2.20) as their difference. In the typical models for capillary pressure such as those presented in Section 2.3.5, the capillary pressure is related to the values of saturation and the continuity of capillary pressure implicates the discontinuity in the saturation values across the interface. The values of saturation $S_w^{e,C}$ and $S_w^{e,F}$ in the coarse and fine sand, respectively, are for the van Genuchten model illustrated in Figure 2.7.

In the case of the Brooks-Corey model, the *barrier effect* can be simulated. When the non-wetting phase is initially not present and approaches the interface from the side of the coarse material, the following situation can occur. When the capillary pressure p_c^C is lower than the entry pressure of the fine material, the non-wetting phase cannot penetrate it and accumulates at the interface. Once the threshold value of the capillary pressure $p_c^C = p_d^F$ is exceeded, the non-wetting phase penetrates the fine material. The threshold pressure value at which the non-wetting phase penetrates the fine material can be given in terms of effective wetting phase saturation $S_w^{e,*}$ by

$$S_w^{e,*} = \left(p_c^C\right)^{-1} \left(p_d^F\right). \tag{2.32}$$

Note that this accumulation at the interface cannot be described by the van Genuchten model (2.23) for which the capillary pressure for a fully water-saturated porous medium is zero.

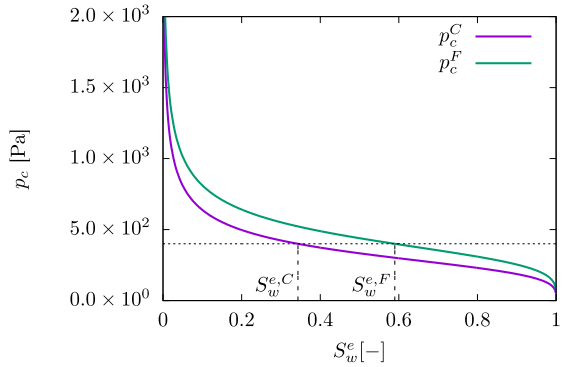


Figure 2.7: Discontinuity of the saturation values across the material interface for the van Genuchten model for capillary pressure. The dotted line show the common value of p_c at the interface.

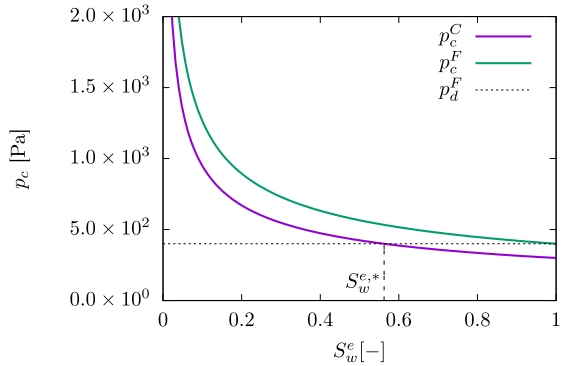


Figure 2.8: Barrier effect for the Brooks-Corey model for capillary pressure. Threshold value $S_w^{e,*}$ for which the non-wetting phase can penetrate the finer sand.

2.3.9 Complete set of two-phase flow equations

To conclude this section, we present the complete set of equations describing the two-phase flow in porous media,

$$\frac{\partial(\Phi S_w \rho_w)}{\partial t} + \nabla \cdot (\rho_w \boldsymbol{u}_w) = f_w, \tag{2.33}$$

$$\boldsymbol{u}_w = -\lambda_\alpha \mathbb{K}(\nabla p_w - \rho_w \boldsymbol{g}), \tag{2.34}$$

$$\frac{\partial(\Phi S_n \rho_n)}{\partial t} + \nabla \cdot (\rho_n \boldsymbol{u}_n) = f_n, \tag{2.35}$$

$$\boldsymbol{u}_n = -\lambda_\alpha \mathbb{K}(\nabla p_n - \rho_n \boldsymbol{g}), \tag{2.36}$$

$$S_n + S_w = 1, \tag{2.37}$$

$$p_n - p_w = p_c. \tag{2.38}$$

There are several possibilities for how to choose the primary unknowns of the system, usually following from the selected numerical scheme. In this work, the capillary pressure and the pressure of the non-wetting phase are selected as the primary unknowns. Alternatives are discussed later in Chapter 5.

2.4 Compositional flow

In this section, we present the governing equations for the compositional flow within the two-phase flow conceptual framework. In general, each phase considered in Section 2.3 can consist of several components that are transported within the phase and their mass can be exchanged between the phases. For example, gas dissolution into water, transport of dissolved gas, and its exsolution. Similarly to the notation of phases, the correspondence of a given quantity to some component κ is denoted by a superscript. The amount of component κ within a phase α at a given point $\boldsymbol{x}_0$ is described in terms of mass fraction $X_\alpha^\kappa\,[-]$ defined by using REV around this point

$$X_\alpha^\kappa(t,\boldsymbol{x}_0) = \frac{\text{mass of component } \kappa \text{ in phase } \alpha \text{ in } REV}{\text{mass of phase } \alpha \text{ in } REV}, \tag{2.39}$$

and satisfying the following conditions for each phase α

$$0 \leq X_\alpha^\kappa \leq 1, \forall \kappa \in L, \qquad \sum_{\kappa \in L} X_\alpha^\kappa = 1, \tag{2.40}$$

where L is the index set of all components of phase α.

2.4.1 Mass balance

The macroscopic mass conservation of component κ within phase α is given by

$$\frac{\partial(\Phi S_\alpha \rho_\alpha X_\alpha^\kappa)}{\partial t} + \nabla \cdot (\rho_\alpha \boldsymbol{u}_{X_\alpha^\kappa}) = f_\alpha^\kappa, \tag{2.41}$$

where the quantities have the same meaning as in (2.13). Here, (2.41) is given for each phase α and each component κ within that phase. The correspondence of the respective quantities to phase and component is indicated by subscript α and superscript κ, respectively.

2.4.2 Component velocity

Component macroscopic velocity $\boldsymbol{u}_{X_\alpha^\kappa}$ in (2.41) is assumed consists of two terms

$$\boldsymbol{u}_{X_\alpha^\kappa} = X_\alpha^\kappa \boldsymbol{u}_\alpha - \mathbb{D}_\alpha^\kappa \nabla X_\alpha^\kappa, \tag{2.42}$$

where $\boldsymbol{u}_\alpha$ is the macroscopic velocity of phase α given by the Darcy's law (2.14) and the first term represents the advection part of the transport. The second term $\mathbb{D}_\alpha^\kappa \nabla X_\alpha^\kappa$ represents the effects of diffusion and dispersion. In this work, we consider the isotropic porous media only where the tensor $\mathbb{D}_\alpha^\kappa$ is given by

$$\mathbb{D}_\alpha^\kappa = \|\boldsymbol{u}_\alpha\| \left(a_T \mathbb{I} + \frac{(a_L - a_T)}{\|\boldsymbol{u}_\alpha\|^2} \boldsymbol{u}_\alpha \otimes \boldsymbol{u}_\alpha \right) + \tau_\alpha \Phi S_\alpha D_\alpha^\kappa \mathbb{I}, \tag{2.43}$$

where the quantities have the following meaning:

$a_L(\boldsymbol{x})$	[m]	is the longitudinal dispersivity [11, 98].
$a_T(\boldsymbol{x})$	[m]	is the transverse dispersivity [11, 98].
$\tau_\alpha(t,\boldsymbol{x})$	$[-]$	is the tortuosity representing the impact of the pore structure on the diffusion for which we adopt the Millington-Quirk model $\tau_\alpha = \Phi^{\frac{1}{3}} S_\alpha^{\frac{7}{3}}$ [79].
D_α^κ	$[\mathrm{m}^2\,\mathrm{s}^{-1}]$	is the molecular diffusion coefficient of component κ within phase α.

2.5 Mass transfer

The mass of components within an individual phase can be exchanged between these phases, for example, gas can dissolve into water or, under different physical conditions, gas dissolved in water can exsolve into the gas phase.

In this work, we focus on the approach describing the mass transfer as a non-equilibrium process that is in (2.41) represented by the sink and source terms f_α^κ. The interphase mass exchange of a component also needs to be taken into account in the source or sink terms f_α in the mass balance equations (2.13) for the phases between which the mass of component is exchanged.

2.5.1 Kinetic model

In the kinetic mass transfer model, the flux of component κ from phase α to β denoted by $Q^\kappa_{\alpha\to\beta}$ [kg m^{-3} s^{-1}] is represented by the first order equation in term of concentration difference $(C^\kappa_{\beta,s} - C^\kappa_\beta)$ [75, 83]

$$Q^\kappa_{\alpha\to\beta} = K^\kappa_{\alpha\to\beta} a_{\alpha\beta}(C^\kappa_{\beta,s} - C^\kappa_\beta), \tag{2.44}$$

where the quantities have the following meaning:

$Q^\kappa_{\alpha\to\beta}$	[kg m^{-3} s^{-1}]	is the mass flux of component κ from phase α to β.
$K^\kappa_{\alpha\to\beta}$	[m s^{-1}]	is the mass transfer rate coefficient.
$a_{\alpha\beta}$	[m^{-1}]	is the specific interfacial area, i.e., the interface area within a REV, separating phases α and β.
$C^\kappa_{\beta,s}$	[kg m^{-3}]	is the saturated concentration (also called solubility limit) of component κ in phase β.
C^κ_β	[kg m^{-3}]	is the current concentration of component κ in phase β.

The interfacial area $a_{\alpha\beta}$ in (2.44) is a parameter which is unknown at the macroscale. Therefore, to employ the kinetic model (2.44) together with the governing equations presented in this chapter, parameter $a_{\alpha\beta}$ needs to be estimated from the known quantities or eliminated from the model.

An alternative approach to obtain the value of $a_{\alpha\beta}$ is to extend the set of governing equations for the two-phase compositional flow in porous media with an additional equation explicitly describing the interfacial area evolution. This approach was proposed in [57, 58, 59, 60, 82], however, such models are beyond the scope of this work. In the following sections, we show the two methods to deal with the value of $a_{\alpha\beta}$ within the available macroscopic quantities: the approach based on the Sherwood number and the Dual-domain approach.

Sherwood number

The commonly used approach for modeling the dissolution of an immobile dense non-aqueous phase liquid (DNAPL) sources into the surrounding water using the kinetic model (2.44) is to lump the mass transfer coefficient $K^\kappa_{\alpha\to\beta}$ and the interfacial area $a_{\alpha\beta}$ into a single parameter $k^\kappa_{\alpha\to\beta}$ [s^{-1}] called the mass transfer coefficient [68, 78, 91, 92].

In the case of DNAPL dissolution, the wetting phase (w) is water, the non-wetting phase (n) is DNAPL, and the only considered component (κ) is DNAPL dissolved in water. The mass transfer coefficient $k^\kappa_{n\to w}$ is then related to the modified dimensionless Sherwood number Sh by

$$k^\kappa_{n\to w} = \mathrm{Sh}\frac{D_m}{d_{50}^2}, \tag{2.45}$$

where the quantities have the following meaning:

D_m	$[\mathrm{m}^2\ \mathrm{s}^{-1}]$	is the aqueous phase molecular diffusion coefficient.
d_{50}	$[\mathrm{m}]$	is the mean size of the porous media grains.

The Sherwood number $\mathbf{Sh}$ is then related to the Reynolds number $\mathbf{Re}$ (2.6) and the non-wetting phase saturation S_n by

$$\mathbf{Sh} = \tilde{\alpha}\mathbf{Re}^{\tilde{\beta}} S_n^{\tilde{\gamma}}, \tag{2.46}$$

where $\tilde{\alpha}$, $\tilde{\beta}$, $\tilde{\gamma}$ are dimensionless parameters that need to be determined using experimental data. Relation (2.46) is not the only one considered for the scenarios of DNAPL dissolution. In [67], various empirical relations for Sherwood number $\mathbf{Sh}$ can be found for different scenarios and ranges of physical conditions such as temperature, pressure, flow velocity, etc., under which they are valid.

Dual-domain approach

The dual-domain approach was proposed in [39] for the modeling of air sparging at the laboratory scale using the kinetic model and is based on the dual-media formulation [6, 48, 51]. In this case, the two-phase system with air as the non-wetting phase and water as the wetting phase is considered. The only component κ is air dissolved in water. The main idea of the dual-domain approach is to divide each REV into two parts: one containing smaller pores leading to high values of the capillary pressure and almost full saturation with water, second containing larger pores leading to lower values of capillary pressure and forming pathways for the gas flow resulting in higher values of the gas saturation. The two domains are then coupled through the first-order mass transfer rate (2.44). Using this approach, the product $k^{\kappa}_{n\to w} a_{nw}$ can be estimated in the form

$$k^{\kappa}_{n\to w} a_{nw} = \left(\Phi S_n \tau_g D_n + \frac{\Phi S_w \tau_w D_w}{H^n_{nw}} \right) \frac{a_{I-II}}{d_{I-II}}, \tag{2.47}$$

where the physical quantities have the following meaning:

H^n_{nw}	$[-]$	is the dimensionless Henry's constant giving the solubility limit of air in water. Details about Henry's law are given in Section 2.5.2.
a_{I-II}	$[\mathrm{m}^{-1}]$	is the specific interfacial area between the two parts containing larger and smaller pores of the REV.
d_{I-II}	$[\mathrm{m}]$	is the diffusion distance between the two parts of REV.
D_α	$[\mathrm{m}^2\ \mathrm{s}^{-1}]$	is the molecular diffusion coefficient of phase α ($\alpha \in \{n, w\}$) .

The factor $\frac{a_{I-II}}{d_{I-II}}$ is then used as a fitting parameter.

2.5.2 Local equilibrium model

Local equilibrium models have been introduced in [68, 78, 91, 92]. The main assumption behind these models is that the thermodynamic equilibrium is reached instantaneously instead of being a transient process as in the case of the kinetic model (2.44).

In general, the equilibrium state can be computed using the thermodynamic principles by finding the state with the minimal (or the maximal) selected potential, for example, the Gibbs free energy, the Helmholtz free energy, etc., under a certain set of constrains [50, 61]. The numerical solution of this global optimization problem is a rather challenging task itself and can be computationally demanding. Therefore, for certain scenarios, simplified relations that are valid under additional assumptions are used instead, for example, determining the equilibrium state of the system in a narrow range of temperature and pressure. An example of such simplified

relations is the Henry's law for dilute solutions and low gas pressures. From (2.44), we assume that for the applications of the models in numerical computations, the equilibrium model can be considered as the kinetic model with a (hypothetical) large enough mass transfer coefficient $k^{\kappa}_{\alpha\to\beta}$, meaning that the equilibrium state is reached fast enough to be considered immediate.

It was shown that the applicability of the equilibrium model is limited and for certain scenarios, the equilibrium model gives wrong results, for example, air sparging in [2, 39, 40]. The differences between the kinetic and equilibrium models for the scenarios of carbon dioxide dissolution into water and exsolution from water are studied later in Chapter 7.

Henry's law

As we stated, finding the equilibrium state using thermodynamic principles is not always needed and the accuracy of simplified relations can be sufficient for a narrow range of physical parameters. In this work, we are interested in carbon dioxide dissolution into and exsolution from water in the range of temperature from 15°C to 40°C in the shallow subsurface with the maximal depth of 2.5 m. Therefore, the pressure values do not change significantly and are within the validity range of the Henry's law. To determine the saturated concentration C_s of a gas dissolved in water as a function of the gas pressure p_g, Henry's law [99] is employed in the form

$$C_s = K_H p_g M_g, \tag{2.48}$$

where the quantities have the following meaning:

K_H	$\left[\mathrm{mol\ m^{-3}\ Pa^{-1}}\right]$	is the Henry's law constant. Here, for convenience different but equivalent formulation of the Henry's law than in 2.5.1 is used.
M_g	$\left[\mathrm{kg\ mol^{-1}}\right]$	is the molar mass of gas.

This value of C_s is then employed in the kinetic model (2.44). The values for the Henry's constant K_H can be found in the literature, for instance [99], for a given couple of solution and solvent at a reference temperature.

2.5.3 Temperature effects

The full non-isothermal model capturing the energy balance and heat transfer between the fluids and the porous medium is not considered in this work. For details about such models, we refer for example to [25]. Although the studied scenarios were designed to be run under isothermal conditions, this was not always fully satisfied. Our observations in Chapter 7 show that the violation of isothermal assumptions by the ambient temperature fluctuation on a daily basis needs to be taken into account. The changes in most of the material properties and physical quantities within the considered range of temperature were negligible and the rate of the temperature changes was low enough for the effects of spatial violation of the isothermal assumption not to play an important role. However, the dependence of the solubility limit of dissolved CO_2 in water cannot be neglected in the studied scenarios. The dependency of the Henry's law constant on the temperature is given by the Van't Hoff equation [99] in the form

$$K_H(T) = K_{H,ref} \exp\left(C_H \left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right), \tag{2.49}$$

where the quantities have the following meaning:

T_{ref}	[K]	is the reference temperature.
$K_{H,ref}$	$\left[\text{mol m}^{-3}\ \text{Pa}^{-1}\right]$	is Henry's constant at a reference temperature T_{ref}.
T	[K]	is the temperature at which $K_H(T)$ is computed.
C_H	[K]	is the gas-specific constant whose value can be found in literature [99].

2.6 Chapter summary

We presented the governing equations for the two-phase compositional flow with the kinetic interphase mass transfer model needed to describe the most general case considered in this work and stated the additional assumptions and simplifications for the model to be valid. The mathematical model is formulated in the form of a coupled system of non-linear partial differential equations and algebraic relations.

Numerical solution 3

Except for special cases that are too restrictive for application to engineering problems, there is no known exact solution of the system of equations arising from the description of the two-phase flow in porous media with component transport and interphase mass transfer presented in Chapter 2. With numerical methods, however, one can find at least a good approximation of the solution. In this chapter, we present a numerical scheme suitable for solving this type of problem.

The problems of flow in porous media studied in this work range from incompressible two-phase flow in homogeneous porous media to more complex ones motivated by experiments and studied physical phenomena, including transport and interphase mass transfer in heterogeneous porous media. The idea to cover various problems by a single numerical solver leads to a numerical scheme formulated for a system of equations in the coefficient form covering the most complex case considered. For the application of this numerical scheme to a given problem, we need to select the primary unknowns and specify which physical quantities are represented by each coefficient in the general formulation and how they are expressed by the selected primary unknowns. The transition to problems where some of the processes are not needed requires only setting the corresponding coefficients to zero with no additional modifications of the scheme needed. This approach allows for the description of the numerical scheme without specifying the solved problem exactly. In addition, it separates the implementation of the numerical scheme from the formulation of the solved problem.

The numerical scheme is based on the mixed-hybrid finite element (MHFEM) spatial discretization with a semi-implicit time-stepping strategy to linearize the equations. The description of the scheme is presented in a unified form for both 2D and 3D cases. The numerical scheme can be also used for solving 1D problems which is not considered in this work, for details, see [43].

3.1 Problem formulation

The numerical scheme presented here is designed for the solution of a system of n partial differential equations in the coefficient form for $i \in \{1, \dots, n\}$

$$\sum_{j=1}^{n} N_{i,j} \frac{\partial Z_j}{\partial t} + \sum_{j=1}^{n} \boldsymbol{u}_{i,j} \cdot \nabla Z_j + \nabla \cdot \left[m_i \left(- \sum_{j=1}^{n} \mathbb{D}_{i,j} \nabla Z_j + \boldsymbol{w}_i \right) + \sum_{j=1}^{n} Z_j \boldsymbol{a}_{i,j} \right] + \sum_{j=1}^{n} r_{i,j} Z_j = f_i, \quad (3.1)$$

where $Z_j = Z_j(\boldsymbol{x}, t)$, $j \in \{1, \dots, n\}$, are the unknown functions. The problem is solved on a spatial domain $\boldsymbol{x} \in \Omega \subset \Re^d$, where $d \in \{2, 3\}$ and $\boldsymbol{x} = (x_1, \dots, x_d)^T$ is a vector of spatial coordinates, and time interval $t \in [0, t_{fin}]$ where t_{fin} denotes the final time.

Equation (3.1) covers the most general problem that can be solved using the numerical scheme presented in this chapter: a system of non-linear unsteady advection-diffusion-reaction equations. The coefficients in (3.1) have the following meaning:

- $N_{i,j}(\boldsymbol{x},t)$ are scalar coefficients describing the time dependency of the problem. Setting them to zero leads to the representation of steady problems.
- $\boldsymbol{a}_{i,j}(\boldsymbol{x},t)$ and $\boldsymbol{u}_{i,j}(\boldsymbol{x},t)$ are vector coefficients describing the advection terms in conservative and non-conservative form, respectively.
- $m_i(\boldsymbol{x},t)$ and $\mathbb{D}_{i,j}(\boldsymbol{x},t)$ are scalar and tensor coefficients, respectively, describing the diffusion. The splitting of diffusion into two coefficients allows for considering vanishing diffusion problems without special treatment of the zero diffusion regions that, for example, occurs when a phase disappears in the case of two-phase flow [43]. In this case, only the mobility m_i becomes zero while the tensor $\mathbb{D}_{i,j}$ is assumed to have eigenvalues bounded from below by a positive constant.
- $\boldsymbol{w}_i(\boldsymbol{x},t)$ are vector coefficients describing the action of external conservative forces.
- $r_{i,j}(\boldsymbol{x},t)$ are scalar coefficients describing the reactions.
- $f_i(\boldsymbol{x},t)$ are scalar coefficients describing the sinks and sources.

The coefficients are $\forall i,j \in \{1,\dots,n\}$ functions of time t and spatial coordinates $\boldsymbol{x}$, but can also be functions of unknowns Z_j. In the aforementioned description, the dependency of both unknowns and coefficients on time t and spatial coordinates $\boldsymbol{x}$ is emphasized. In the following text, unless stated otherwise, this dependency is assumed, but the notation $(\boldsymbol{x},t)$ for both unknowns and coefficients is omitted for clarity.

To allow for a compact presentation of the underlying problems the unknown functions are gathered into vector $\mathcal{Z} = \{Z_j\}_{j=1}^n$. In addition, the coefficients in (3.1) are gathered into vectors $\mathcal{M} = \{m_i\}_{i=1}^n$, $\mathcal{W} = \{\boldsymbol{w}_i\}_{i=1}^n$, $\mathcal{F} = \{f_i\}_{i=1}^n$, and matrices $\mathcal{N} = \{N_{i,j}\}_{i,j=1}^n$, $\mathcal{U} = \{\boldsymbol{u}_{i,j}\}_{i,j=1}^n$, $\mathcal{A} = \{\boldsymbol{a}_{i,j}\}_{i,j=1}^n$, $\mathcal{D} = \{\mathbb{D}_{i,j}\}_{i,j=1}^n$, $\mathcal{R} = \{r_{i,j}\}_{i,j=1}^n$.

The numerical scheme treats separately two closely related vector terms in the general formulation (3.1): conservative flux $\boldsymbol{q}_i$ and conservative velocity $\boldsymbol{v}_i$ defined by

$$\boldsymbol{q}_i = m_i \left(-\sum_{j=1}^{n} \mathbb{D}_{i,j} \nabla Z_j + \boldsymbol{w}_i \right), \qquad \forall i \in \{1,\dots,n\}, \tag{3.2a}$$

and

$$\boldsymbol{v}_i = -\sum_{j=1}^{n} \mathbb{D}_{i,j} \nabla Z_j + \boldsymbol{w}_i, \qquad \forall i \in \{1,\dots,n\}, \tag{3.2b}$$

respectively. In addition, these vectors are needed for the prescription of the Neumann boundary conditions. Furthermore, we denote the partial velocity $\boldsymbol{v}_{i,j}$

$$\boldsymbol{v}_{i,j} = -\mathbb{D}_{i,j} \nabla Z_j, \qquad \forall i,j \in \{1,\dots,n\}. \tag{3.3}$$

Using the notation from (3.2b) and (3.3), the conservative velocity can be assembled from partial velocities and conservative forces as

$$\boldsymbol{v}_i = \sum_{j=1}^{n} \boldsymbol{v}_{i,j} + \boldsymbol{w}_i. \tag{3.4}$$

The system of differential equations (3.1) is endowed with Dirichlet or Neumann boundary conditions or their combination on different parts of the domain boundary $\partial\Omega$

$$Z_i(\boldsymbol{x},t) = Z_{i,Dir}(\boldsymbol{x},t), \qquad \forall(\boldsymbol{x},t) \in \partial\Omega_{i,Dir}, \qquad \forall i \in \{1,\dots,n\}, \tag{3.5}$$

$$\boldsymbol{v}_i(\boldsymbol{x},t)\cdot\boldsymbol{n}(\boldsymbol{x}) = v_{i,Neu}(\boldsymbol{x},t), \qquad \forall(\boldsymbol{x},t) \in \partial\Omega_{i,Neu}, \qquad \forall i \in \{1,\dots,n\}, \tag{3.6}$$

where $\boldsymbol{n}(\boldsymbol{x})$ is the outward unit normal to $\partial\Omega$, $Z_{i,Dir}$ is the value of the Dirichlet boundary condition for the unknown function Z_i, and $\boldsymbol{v}_{i,Neu}$ is the value of the Neumann boundary condition for conservative velocity $\boldsymbol{v}_i$. The Neumann and Dirichlet parts of the boundary $\partial\Omega_{i,\bullet}$ of $\partial\Omega$ satisfy the conditions

$$\overline{\partial\Omega_{i,Dir} \cup \partial\Omega_{i,Neu}} = \partial\Omega, \qquad \forall i \in \{1,\dots,n\}, \tag{3.7}$$

$$\partial\Omega_{i,Dir} \cap \partial\Omega_{i,Neu} = \emptyset, \qquad \forall i \in \{1,\dots,n\}. \tag{3.8}$$

The system (3.1) is also endowed with the initial conditions

$$Z_i(\boldsymbol{x},0) = Z_{i,0}(\boldsymbol{x}), \qquad \forall \boldsymbol{x} \in \Omega, \qquad \forall i \in \{1,\dots,n\}, \tag{3.9}$$

where $Z_{i,0}$ is the value of the initial condition for unknown Z_i.

3.2 Weak formulation

To obtain a weak formulation of (3.1), we use the standard procedure for the mixed-hybrid finite element method in detail described in [15]. We recall the functional spaces for the scalar unknowns

$$H^1(\Upsilon) = \left\{ v :\ v \in L_2(\Upsilon), \frac{\partial v}{\partial x_q} \in L_2(\Upsilon), \forall q \in \{1,\dots,d\} \right\}, \tag{3.10}$$

and for the vector unknowns

$$H(\mathrm{div},\Upsilon) = \left\{ \boldsymbol{q} :\ \boldsymbol{q} \in (L_2(\Upsilon))^d, \mathrm{div}\,\boldsymbol{q} \in L_2(\Upsilon) \right\}. \tag{3.11}$$

We assume that vector coefficients and unknowns $\boldsymbol{u}_{i,j}$, $\boldsymbol{v}_{i,j}$, $\boldsymbol{w}_{i,j}$, $\boldsymbol{a}_{i,j}$ belong to $H(\mathrm{div},\Omega)$, scalar coefficient $N_{i,j}$, m_i, $r_{i,j}$, f_i belong to $L_2(\Omega)$, and scalar unknowns Z_j belong to $H^1(\Omega)$. Then, we consider functions $\varphi \in L_2(\Omega)$ and $\boldsymbol{\omega} \in H(\mathrm{div},\Omega)$ and use the notation for the velocity term (3.2b). Multiplying (3.1) by φ, (3.2b) by $\boldsymbol{\omega}$, and integrating over the domain Ω, we obtain the mixed weak formulation of the problem in the form

$$\int_\Omega \left(\sum_{j=1}^n N_{i,j} \frac{\partial Z_j}{\partial t} \right) \varphi + \int_\Omega \left(\sum_{j=1}^n \boldsymbol{u}_{i,j} \cdot \nabla Z_j \right) \varphi + \int_\Omega \left(\nabla \cdot \left[\boldsymbol{q}_i + \sum_{j=1}^n Z_j \boldsymbol{a}_{i,j} \right] \right) \varphi + \int_\Omega \left(\sum_{j=1}^n r_{i,j} Z_j \right) \varphi = \int_\Omega f_i \varphi,$$

$$\forall \varphi \in L^2(\Omega), \quad \forall i \in \{1,\dots,n\}, \tag{3.12}$$

$$\int_\Omega \boldsymbol{q}_i \cdot \boldsymbol{\omega} = \int_\Omega m_i \left(-\sum_{j=1}^n \mathbb{D}_{i,j} \nabla Z_j + \boldsymbol{w}_i \right) \cdot \boldsymbol{\omega},$$

$$\forall \boldsymbol{\omega} \in H(\mathrm{div},\Omega), \quad \forall i \in \{1,\dots,n\}. \tag{3.13}$$

In the following sections, we define suitable finite-dimensional subspaces of $L_2(\Omega)$, $H^1(\Omega)$, and $H(\mathrm{div},\Omega)$ to obtain a discrete formulation of (3.12) and (3.13).

3.3 Mixed-hybrid finite element method

The spatial discretization of the numerical scheme is based on the mixed-hybrid finite element method that uses different approximations for the vector and scalar terms. Before we proceed with the approximations, we define the feasible numerical meshes.

3.3.1 Mesh

We assume that the computational domain $\Omega \in \mathfrak{R}^d$ is covered by a conforming mesh consisting of one type of elements. The set of all mesh elements is denoted by $\mathcal{K}_h$. For the sake of simplicity the following mesh types are considered only:

- Unstructured meshes with triangular and tetrahedral elements in 2D and 3D, respectively.
- Structured meshes with rectangular and cuboidal elements in 2D and 3D, respectively.

Similarly to the set of all mesh elements $\mathcal{K}_h$, the set of all the mesh *sides* $\mathcal{E}_h$ is introduced, allowing for a unified description of the numerical scheme for both 2D and 3D cases.

- In 2D, $\mathcal{E}_h$ consists of edges of elements $K \in \mathcal{K}_h$, i.e., sides of triangles or rectangles based on the mesh type.
- In 3D, $\mathcal{E}_h$ consists of faces of elements $K \in \mathcal{K}_h$, i.e., faces of tetrahedrons or cuboids based on the mesh type.

In addition, the numerical scheme requires computations over all sides of an element, for convenience, the set of all sides of element K is denoted by $\mathcal{E}_K$. The number of entries in a given set is denoted by #. In this notation, the number of elements and sides of a numerical mesh is denoted by $\#\mathcal{K}_h$ and $\#\mathcal{E}_h$, respectively.

3.3.2 Vector terms

For the approximation of the vector terms in the system (3.1), we assume that the vector functions on each element $K \in \mathcal{K}_h$ belong to the functional space $H(\mathrm{div}, K)$ and these vectors are on each element K approximated in the lowest order Raviart-Thomas-Nédélec space $\mathbf{RTN}_0(K)$ [15].

For the expression of the vector as a linear combination of the basis vectors, a suitable base of the $\mathbf{RTN}_0(K)$ space is needed. The basis functions $\boldsymbol{\omega}$ are selected to satisfy the following conditions

$$\nabla \cdot \boldsymbol{\omega}_{K,E}(\boldsymbol{x}) = \frac{1}{|K|_d}, \qquad \forall \boldsymbol{x} \in K, \quad \forall K \in \mathcal{K}_h, \quad \forall E \in \mathcal{E}_h, \tag{3.14a}$$

$$\boldsymbol{\omega}_{K,E}(\boldsymbol{x}) \cdot \boldsymbol{n}_{K,F}(\boldsymbol{x}) = \delta_{E,F} \frac{1}{|E|_{d-1}}, \quad \forall \boldsymbol{x} \in F, \quad \forall K \in \mathcal{K}_h, \quad \forall E, F \in \mathcal{E}_K, \tag{3.14b}$$

$$\int_K \nabla \cdot \boldsymbol{\omega}_{K,E}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 1, \qquad \forall K \in \mathcal{K}_h, \quad \forall E \in \mathcal{E}_K, \tag{3.14c}$$

$$\int_E \boldsymbol{\omega}_{K,E}(\boldsymbol{x}) \cdot \boldsymbol{n}_{K,E}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 1, \qquad \forall K \in \mathcal{K}_h, \quad \forall E \in \mathcal{E}_K, \tag{3.14d}$$

where the subscript K denotes the element on which the basis function is considered, the subscript E denotes to which side of element K the basis function corresponds, and $\boldsymbol{n}_{K,F}(\boldsymbol{x})$ is the outward unit normal to side F of element K. The Kronecker delta symbol $\delta_{E,F}$ is defined by $\delta_{E,F} = 1$ if $E = F$ and $\delta_{E,F} = 0$ otherwise. In the following sections, we provide the basis functions satisfying the conditions (3.14) for all the mesh elements considered in this work.

2D and 3D unstructured meshes: simplices

For the unstructured meshes that consist of triangles and tetrahedrons in 2D and 3D, respectively, the basis functions on $K \in \mathcal{K}_h$ in a space of dimension d are selected in the form

$$\boldsymbol{\omega}_{K,E}(\boldsymbol{x}) = \frac{1}{d|K|_d}\left(\boldsymbol{x} - \boldsymbol{x}_E\right), \qquad \forall E \in \mathcal{E}_K, \tag{3.15}$$

where $|\cdot|_d$ denotes the d-dimensional Lebesgue measure.

2D structured mesh: rectangles

For the 2D structured mesh, where the elements in $\mathcal{K}_h$ are rectangles, the basis functions on the reference rectangle $\begin{pmatrix} h_x & 0 \end{pmatrix} \times \begin{pmatrix} 0 & h_y \end{pmatrix}$ are selected in the form

$$\boldsymbol{\omega}_{K,E_1}(\boldsymbol{x}) = \frac{1}{|K|_2}\begin{pmatrix} x_1 - h_x \\ 0 \end{pmatrix}, \qquad \boldsymbol{\omega}_{K,E_2}(\boldsymbol{x}) = \frac{1}{|K|_2}\begin{pmatrix} x_1 \\ 0 \end{pmatrix}, \tag{3.16a}$$

$$\boldsymbol{\omega}_{K,E_3}(\boldsymbol{x}) = \frac{1}{|K|_2}\begin{pmatrix} 0 \\ x_2 - h_y \end{pmatrix}, \qquad \boldsymbol{\omega}_{K,E_4}(\boldsymbol{x}) = \frac{1}{|K|_2}\begin{pmatrix} 0 \\ x_2 \end{pmatrix}, \tag{3.16b}$$

where the vector $\boldsymbol{x} = (x_1, x_2)^T$ and the edges of the rectangle are numbered as follows: E_1 is the edge $x_1 = 0$, E_2 is the edge $x_1 = h_x$, E_3 is the edge $x_2 = 0$, E_4 is the edge $x_2 = h_y$.

3D structure mesh: cuboids

For the 3D structured mesh, where the elements in $\mathcal{K}_h$ are cuboids, the basis functions on the reference cuboid $\begin{pmatrix} h_x & 0 \end{pmatrix} \times \begin{pmatrix} 0 & h_y \end{pmatrix} \times \begin{pmatrix} 0 & h_z \end{pmatrix}$ are selected in the form

$$\boldsymbol{\omega}_{K,E_1}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} x_1 - h_x \\ 0 \\ 0 \end{pmatrix}, \qquad \boldsymbol{\omega}_{K,E_2}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} x_1 \\ 0 \\ 0 \end{pmatrix}, \tag{3.17a}$$

$$\boldsymbol{\omega}_{K,E_3}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} 0 \\ x_2 - h_y \\ 0 \end{pmatrix}, \qquad \boldsymbol{\omega}_{K,E_4}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} 0 \\ x_2 \\ 0 \end{pmatrix}, \tag{3.17b}$$

$$\boldsymbol{\omega}_{K,E_5}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} 0 \\ 0 \\ x_3 - h_z \end{pmatrix}, \qquad \boldsymbol{\omega}_{K,E_6}(\boldsymbol{x}) = \frac{1}{|K|_3}\begin{pmatrix} 0 \\ 0 \\ x_3 \end{pmatrix}, \tag{3.17c}$$

where the vector $\boldsymbol{x} = (x_1, x_2, x_3)^T$ and the faces of the cuboid are numbered as follows: E_1 is the face $x_1 = 0$, E_2 is the face $x_1 = h_x$, E_3 is the face $x_2 = 0$, E_4 is the face $x_2 = h_y$, E_5 is the face $x_3 = 0$, E_6 is the face $x_3 = h_z$.

Approximation of vector terms

With the selected basis functions, all the vector coefficients in (3.1), the introduced conservative velocity $\boldsymbol{v}_i$ (3.2b), and partial velocities $\boldsymbol{v}_{i,j}$ (3.3) are approximated in $\mathrm{RTN}_0(K)$, $\forall K \in \mathcal{K}_h$ as

follows

$$\boldsymbol{v}_i(\boldsymbol{x},t)\,|_K \approx \sum_{E\in\mathcal{E}_K} v_{i,K,E}(t)\boldsymbol{\omega}_{K,E}(\boldsymbol{x}), \tag{3.18a}$$

$$\boldsymbol{v}_{i,j}(\boldsymbol{x},t)\,|_K \approx \sum_{E\in\mathcal{E}_K} v_{i,j,K,E}(t)\boldsymbol{\omega}_{K,E}(\boldsymbol{x}), \tag{3.18b}$$

$$\boldsymbol{u}_{i,j}(\boldsymbol{x},t)\,|_K \approx \sum_{E\in\mathcal{E}_K} u_{i,j,K,E}(t)\boldsymbol{\omega}_{K,E}(\boldsymbol{x}), \tag{3.18c}$$

$$\boldsymbol{w}_i(\boldsymbol{x},t)\,|_K \approx \sum_{E\in\mathcal{E}_K} w_{i,K,E}(t)\boldsymbol{\omega}_{K,E}(\boldsymbol{x}), \tag{3.18d}$$

$$\boldsymbol{a}_{i,j}(\boldsymbol{x},t)\,|_K \approx \sum_{E\in\mathcal{E}_K} a_{i,j,K,E}(t)\boldsymbol{\omega}_{K,E}(\boldsymbol{x}), \tag{3.18e}$$

where $v_{i,K,E}$, $v_{i,j,K,E}$, $u_{i,j,K,E}$, $w_{i,K,E}$, and $a_{i,j,K,E}$ denote coefficients in the linear combination for the corresponding vector term referred to as *components*.

Vector terms on each element $K \in \mathcal{K}_h$ given by (3.18) in general does not belong to $H(\mathrm{div},\Omega)$. To belong to this space, the vector terms need to be continuous over internal sides $E \in \mathcal{E}_h$ in terms of their normal traces [15]. Using the property of the basis functions of $\mathrm{RTN}_0(K)$ (3.14d), this continuity condition at side E common to elements K_1 and K_2 ($E = \mathcal{E}_{K_1} \cap \mathcal{E}_{K_2}$), can be formulated in terms of balancing the corresponding components of vector terms

$$u_{i,j,K_1,E} + u_{i,j,K_2,E} = 0, \qquad \forall i,j \in \{1,\dots,n\}, \tag{3.19a}$$

$$w_{i,K_1,E} + w_{i,K_2,E} = 0, \qquad \forall i \in \{1,\dots,n\}, \tag{3.19b}$$

$$a_{i,j,K_1,E} + a_{i,j,K_2,E} = 0, \qquad \forall i,j \in \{1,\dots,n\}. \tag{3.19c}$$

For the vector unknowns $\boldsymbol{q}_i$, $\boldsymbol{v}_i$, and $\boldsymbol{v}_{i,j}$, the continuity of the normal traces will be discussed later in Section 3.6.

3.3.3 Scalar terms

In contrast to the vector terms, we employ the discontinuous Galerkin approach and the scalar terms are approximated in the space of element-wise constant functions, for a given set of mesh elements $\mathcal{K}_h$ denoted by $D_0(\mathcal{K}_h)$. Basis function $\varphi(\boldsymbol{x})_K$ on a mesh element $K \in \mathcal{K}_h$ is defined by

$$\varphi(\boldsymbol{x})_K = \begin{cases} 1 & \boldsymbol{x} \in K, \\ 0 & \text{otherwise}, \end{cases} \tag{3.20}$$

and has the following properties

$$\int_K \varphi(\boldsymbol{x}) = |K|_d, \qquad \forall K \in \mathcal{K}_h, \tag{3.21a}$$

and

$$\nabla\varphi(\boldsymbol{x}) = \boldsymbol{0}, \qquad \forall \boldsymbol{x} \in K, \qquad \forall K \in \mathcal{K}_h. \tag{3.21b}$$

Approximation of scalar and tensor terms

With the approximation described above, the scalar and tensor terms in (3.1) are approximated in $D_0(\mathcal{K}_h)$, $\forall K \in \mathcal{K}_h$ as follows

$$Z_j(\boldsymbol{x},t)\,|_K \approx Z_{j,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22a}$$
$$N_{i,j}(\boldsymbol{x},t)\,|_K \approx N_{i,j,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22b}$$
$$m_i(\boldsymbol{x},t)\,|_K \approx m_{i,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22c}$$
$$\mathbb{D}_{i,j}(\boldsymbol{x},t)\,|_K \approx \mathbb{D}_{i,j,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22d}$$
$$r_{i,j}(\boldsymbol{x},t)\,|_K \approx r_{i,j,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22e}$$
$$f_i(\boldsymbol{x},t)\,|_K \approx f_{i,K}(t)\varphi_K(\boldsymbol{x}), \tag{3.22f}$$

where $Z_{j,K}$, $N_{i,j,K}$, $m_{i,K}$, $\mathbb{D}_{i,j,K}$, $r_{i,k,K}$, $f_{i,K}$ are for simplicity referred to as the approximation of the corresponding term on element $K \in \mathcal{K}_h$.

3.4 Diffusion terms

With the definition of the finite-dimensional spaces for vector and scalar terms, we first focus on the approximation of the conservative velocity $\boldsymbol{v}_i$ in $\mathrm{RTN}_0(K)$ defined by (3.2b), $\forall K \in \mathcal{K}_h$. The goal is to express the components $v_{i,K,E}$ introduced in (3.18) as functions of scalar unknowns Z_j, $j \in \{1,\dots,n\}$. We start with the expression of the partial velocity $\boldsymbol{v}_{i,j}$ defined by (3.3) and its approximation in $\mathrm{RTN}_0(K)$ given by (3.18b)

$$\boldsymbol{v}_{i,j}\,|_K \approx \sum_{E\in\mathcal{E}_K} v_{i,j,K,E}\boldsymbol{\omega}_{K,E} = -\mathbb{D}_{i,j,K}\nabla Z_j, \tag{3.23}$$

from which we can then assemble $\boldsymbol{v}_i$ using (3.4). We assume that tensors $\mathbb{D}_{i,j,K}$ are either positive definite or zero. Note that this condition is always satisfied for the two-phase compositional flow problems in porous media.

In the case of $\mathbb{D}_{i,j,K}$ being zero, the components $v_{i,j,K,E}$ in (3.23) must be also zero: the right-hand side of (3.23) is zero and the projection into $\mathrm{RTN}_0(K)$ is trivial. The following steps are not needed and we can proceed directly to (3.31) and the following remarks.

For a symmetric positive definite tensor $\mathbb{D}_{i,j,K}$, there exists its inversion $\mathbb{D}^{-1}_{i,j,K}$. Multiplying (3.23) by this inversion, we obtain

$$\sum_{E\in\mathcal{E}_K} v_{i,j,K,E}\mathbb{D}^{-1}_{i,j,K}\boldsymbol{\omega}_{K,E} = -\nabla Z_j. \tag{3.24}$$

Equation (3.24) is then multiplied by the basis function $\boldsymbol{\omega}_{K,F}$ and integrated over the element K to obtain

$$\sum_{E\in\mathcal{E}_K} v_{i,j,K,E}\int_K \boldsymbol{\omega}^T_{K,F}\mathbb{D}^{-1}_{i,j,K}\boldsymbol{\omega}_{K,E} = -\int_K \nabla Z_j\cdot\boldsymbol{\omega}_{K,F}, \qquad \forall F\in\mathcal{E}_K. \tag{3.25}$$

Using the Green formula and properties of the basis functions (3.14), the right-hand side of (3.25) is transformed into

$$-\int_K \nabla Z_j\cdot\boldsymbol{\omega}_{K,F} = \int_K Z_j\nabla\cdot\boldsymbol{\omega}_{K,F} - \int_{\mathcal{E}_K} Z_j\boldsymbol{\omega}_{K,F}\cdot\boldsymbol{n} = -Z_{j,F} + Z_{j,K}, \tag{3.26}$$

where the newly introduced quantity $Z_{j,F}$ is the mean value of the trace of Z_j over side F

$$Z_{j,F} = \frac{1}{|F|_{d-1}}\int_F Z_j. \tag{3.27}$$

Based on [15], the common value of trace (3.27) on $\boldsymbol{F}$ can be interpreted as a Lagrange multiplier allowing for formulating the discretization of weak formulation (3.13) locally on each element $\boldsymbol{K} \in \mathcal{K}_h$ without explicitly enforcing the continuity in the terms of normal traces of vector terms $\boldsymbol{q}_i$, $\boldsymbol{v}_i$, and $\boldsymbol{v}_{i,j}$.

Then, the equations (3.25) and (3.26) combined together give

$$\sum_{E \in \mathcal{E}_K} v_{i,j,K,E} \int_K \boldsymbol{\omega}_{K,F} \mathbb{D}^{-1}_{i,j,K} \boldsymbol{\omega}_{K,E} = -Z_{j,F} + Z_{j,K}, \qquad \forall F \in \mathcal{E}_K. \tag{3.28}$$

The integrals over $K \in \mathcal{K}_h$ of the combination of basis functions $\boldsymbol{\omega}_{K,F}$, $\boldsymbol{\omega}_{K,E}$ and tensors $\mathbb{D}^{-1}_{i,j,K}$ in (3.28) form a coefficient matrix denoted by $\mathbb{H}_{i,j,K} = \{H_{i,j,K,E,F}\}_{E,F \in \mathcal{E}_K}$ with entries defined by

$$H_{i,j,K,E,F} = \int_K \boldsymbol{\omega}_{K,F} \mathbb{D}^{-1}_{i,j,K} \boldsymbol{\omega}_{K,E}. \tag{3.29}$$

Using $\mathbb{H}_{i,j,K}$, (3.28) can be written as

$$Z_{j,K} - Z_{j,F} = \sum_{E \in \mathcal{E}_K} v_{i,j,K,E} H_{i,j,K,E,F}, \qquad \forall F \in \mathcal{E}_K. \tag{3.30}$$

In the case of positive definite tensor $\mathbb{D}_{i,j,K}$, matrix $\mathbb{H}_{i,j,K}$ is also positive definite and its inversion is denoted by $\mathbb{B}_{i,j,K}$ ($\mathbb{B}_{i,j,K} = \mathbb{H}^{-1}_{i,j,K}$). Equation (3.30) is then multiplied by matrix $\mathbb{B}_{i,j,K}$ from the left to obtain

$$v_{i,j,K,E} = B_{i,j,K,E} Z_{j,K} - \sum_{F \in \mathcal{E}_K} B_{i,j,K,E,F} Z_{j,F}, \qquad \forall E \in \mathcal{E}_K, \tag{3.31}$$

where $B_{i,j,K,E} = \sum_{F \in \mathcal{E}_K} B_{i,j,K,E,F}$ is introduced to simplify the notation.

To unify the description of the numerical scheme for both positive definite and zero tensor $\mathbb{D}_{i,j,K}$, $\mathbb{B}_{i,j,K} = \Theta$ is defined for the case of zero tensor $\mathbb{D}_{i,j,K}$.

Equation (3.31) gives the components of $\boldsymbol{v}_{i,j}$ in the selected basis of $\mathrm{RTN}_0(K)$ as functions of scalar quantities Z_j. Using the expression of the partial velocities (3.31), the conservative velocity $\boldsymbol{v}_i$ defined by (3.2b) is expressed as a function of the unknowns Z_j. In (3.18a), the coefficients $v_{i,K,E}$ are given by

$$v_{i,K,E} = \sum_{j=1}^{n} \left(B_{i,j,K,E} Z_{j,K} - \sum_{F \in \mathcal{E}_K} B_{i,j,K,E,F} Z_{j,F} \right) + w_{i,K,E}, \qquad \forall i \in \{1, \ldots, n\}, \quad \forall K \in \mathcal{K}_h, \quad \forall E \in \mathcal{E}_K. \tag{3.32}$$

3.5 Approximation of conservation laws

In this section, we describe the complete spatio-temporal discretization of the weak formulation (3.12) of the general problem (3.1). We start with the weak formulation of the problem (3.12), employ the approximation of the scalar and vector terms given by (3.22) and (3.18), respectively. To obtain a discrete weak formulation, we use functions φ from the basis of $D_0(\mathcal{K}_h)$. Each of the basis functions defined by (3.20) equals identity on a single element $K \in \mathcal{K}_h$, which reduces the integral over the whole domain Ω in (3.12) to the integrals over elements mesh elements $K \in \mathcal{K}_h$

$$\underbrace{\int_K \sum_{j=1}^{n} N_{i,j} \frac{\partial Z_j}{\partial t}}_{I_1} + \underbrace{\int_K \sum_{j=1}^{n} \boldsymbol{u}_{i,j} \cdot \nabla Z_j}_{I_2} + \underbrace{\int_K \nabla \cdot \left[\boldsymbol{q}_i + \sum_{j=1}^{n} Z_j \boldsymbol{a}_{i,j} \right]}_{I_3} + \underbrace{\int_K \sum_{j=1}^{n} r_{i,j} Z_j}_{I_4} = \underbrace{\int_K f_i}_{I_5}, \qquad \forall i \in \{1, \ldots, n\}, \quad \forall K \in \mathcal{K}_h. \tag{3.33}$$

The partial derivative in time is approximated by the backward Euler difference and the temporal dependency of the other quantities is resolved by the semi-implicit approach in time in detail discussed in the following sections. The times t_k and t_{k+1} at which the coefficients and the unknowns are evaluated are denoted by the corresponding superscripts k and $k+1$, respectively. For clarity, the procedure is divided into five terms as highlighted in (3.33): four on the left-hand side and one on the right-hand side of the equation, and operations performed on each term are explained separately for i-th equation on a given mesh element $K \in \mathcal{K}_h$.

3.5.1 Integral I_1

The first term in (3.33) describing the time dependency of the problem using coefficient $\mathcal{N}$ is approximated by

$$\int_K \sum_{j=1}^{n} N_{i,j} \frac{\partial Z_j}{\partial t} \approx \sum_{j=1}^{n} N_{i,j,K} \frac{\mathrm{d}}{\mathrm{d}t} \int_K Z_j \approx \sum_{j=1}^{n} N^k_{i,j,K} \frac{|K|_d}{\Delta t_k} \left(Z^{k+1}_{j,K} - Z^k_{j,K} \right), \tag{3.34}$$

where:

- $N^k_{i,j,K}$ is the approximation of coefficient $N_{i,j}$ on element K at time t_k. In the case of $N_{i,j}$ independent of $\mathcal{Z}$, the value from the time t_{k+1} is used.
- $Z^k_{j,K}$ is the approximation of Z_j on element K at time t_k.
- $Z^{k+1}_{j,K}$ is the approximation of Z_j on element K at time t_{k+1}.
- Δt_k is the time step ($\Delta t_k = t_{k+1} - t_k$).

3.5.2 Integral I_2

The second term in (3.33) describing the advection in the non-conservative form using coefficient $\mathcal{U}$ is approximated by

$$\begin{aligned} &\int_K \sum_{j=1}^{n} \boldsymbol{u}_{i,j} \nabla Z_j \approx \int_K \sum_{j=1}^{n} \sum_{E \in \mathcal{E}_K} u_{i,j,K,E} \boldsymbol{\omega}_{K,E} \nabla Z_j \approx \sum_{j=1}^{n} \sum_{E \in \mathcal{E}_K} u^k_{i,j,K,E} \int_K \boldsymbol{\omega}_{K,E} \nabla Z_j \approx \\ &\sum_{j=1}^{n} \sum_{E \in \mathcal{E}_K} u^k_{i,j,K,E} \left(\int_{\mathcal{E}_K} Z_j \boldsymbol{\omega}_{K,E} \cdot \boldsymbol{n} - \int_K Z_j \nabla \cdot \boldsymbol{\omega}_{K,E} \right) \approx \sum_{j=1}^{n} \sum_{E \in \mathcal{E}_K} u^k_{i,j,K,E} \left(Z^{k,upw}_{i,j,E} - Z^{k+1}_{j,K} \right), \end{aligned} \tag{3.35}$$

where:

- $u^k_{i,j,K,E}$ is the component of $\boldsymbol{u}_{i,j}$ on element K at time t_k. In the case of $\boldsymbol{u}_{i,j}$ independent of $\mathcal{Z}$ the value from the time t_{k+1} is used.
- $Z^{k+1}_{j,K}$ is the approximation of Z_j on element K at time t_{k+1}.
- $Z^{k,upw}_{i,j,E}$ is the *upwinded* value of Z_j on side E at time t_k, details of this choice are provided in the following.

The upwind technique is used in (3.35) to ensure the numerical stability of the scheme. The *upwinded* quantity $Z^{k,upw}_{i,j,E}$ is introduced as follows. On the interior sides $E = \mathcal{E}_{K_1} \cap \mathcal{E}_{K_2}$ common to two elements $K_1, K_2 \in \mathcal{K}_h$, we assume that the advection terms are continuous over E in terms of their normal traces, i.e., their components are balanced by (3.19)

$$u^k_{i,j,K_1,E} + a^k_{i,j,K_1,E} + u^k_{i,j,K_2,E} + a^k_{i,j,K_2,E} = 0, \qquad \forall i, j \in \{1, \ldots, n\}. \tag{3.36}$$

The values of $Z_{i,j,E}^{k,upw}$ are then defined by

$$Z_{i,j,E}^{k,upw} = \begin{cases} Z_{j,K_1}^k, & \text{if } u_{i,j,K_1,E}^k + a_{i,j,K_1,E}^k > 0, \\ Z_{j,K_2}^k, & \text{if } u_{i,j,K_2,E}^k + a_{i,j,K_2,E}^k > 0, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i,j \in \{1,\dots,n\}. \tag{3.37}$$

On the boundary sides $E \in \mathcal{E}_{K_1}$ belonging to a single element $K_1 \in \mathcal{K}_h$ only, the value from the boundary condition is used instead of (3.37) in the case of $u_{i,j,K_1,E}^k + a_{i,j,K_1,E}^k < 0$.

At the first time step ($k = 0$), the values of $u_{i,j,K,E}^0 + a_{i,j,K,E}^0$ may not be known, therefore, the arithmetic average of the values from the neighboring elements is used

$$Z_{i,j,E}^{0,upw} = \frac{1}{2}\left(Z_{j,K_1}^0 + Z_{j,K_2}^0\right), \qquad \forall i,j \in \{1,\dots,n\}. \tag{3.38}$$

3.5.3 Integral I_3

The third term in (3.33) describing the diffusion using coefficients $\mathcal{M}$ and $\mathcal{D}$, the conservative forces using coefficient $\mathcal{W}$, and the advection in the conservative form using coefficient $\mathcal{A}$ is approximated by

$$\begin{aligned} &\int_K \nabla \cdot \left[m_i \left(-\sum_{j=1}^n \mathbb{D}_{i,j,K} \nabla Z_j + \boldsymbol{w}_i \right) + \sum_{j=1}^n Z_j \boldsymbol{a}_{i,j} \right] = \\ &\qquad \int_{\mathcal{E}_K} \left[m_i \left(-\sum_{j=1}^n \mathbb{D}_{i,j,K} \nabla Z_j + \boldsymbol{w}_i \right) + \sum_{j=1}^n Z_j \boldsymbol{a}_{i,j} \right] \cdot \boldsymbol{n} \approx \\ &\qquad \int_{\mathcal{E}_K} m_i \sum_{E \in \mathcal{E}_K} v_{i,K,E} \boldsymbol{\omega}_{\boldsymbol{K},\boldsymbol{E}} \cdot \boldsymbol{n} + \int_{\mathcal{E}_K} \sum_{j=1}^n \sum_{E \in \mathcal{E}_K} Z_j a_{i,j,K,E} \boldsymbol{\omega}_{K,E} \cdot \boldsymbol{n} \approx \\ &\qquad \sum_{E \in \mathcal{E}_K} m_{i,E}^{k,upw} v_{i,K,E} + \sum_{j=1}^n \sum_{E \in \mathcal{E}_K} a_{i,j,K,E} Z_{i,j,E}^{k,upw} \approx \\ &\sum_{E \in \mathcal{E}_K} m_{i,E}^{k,upw} \left(\sum_{j=1}^n \left(B_{i,j,K,E}^k Z_{j,K}^{k+1} - \sum_{F \in \mathcal{E}_K} B_{i,j,K,E,F}^k Z_{j,F}^{k+1} \right) + w_{i,K,E}^k \right) + \sum_{j=1}^n \sum_{E \in \mathcal{E}_K} a_{i,j,K,E}^k Z_{i,j,E}^{k,upw}, \end{aligned} \tag{3.39}$$

where the coefficient $v_{i,K,E}$ is substituted from (3.32) and:

- $m_{i,E}^{k,upw}$ is the *upwinded* value of m_i on side E at time t_k, details of this choice are provided in the following.
- $B_{i,j,K,E}^{k}$ are the local coefficients $B_{i,j,K,E}$ at time t_k. In the case of $B_{i,j,K,E}$ independent of $\mathcal{Z}$ the value from time t_{k+1} is used.
- $B_{i,j,K,E,F}^{k}$ are the local coefficients $B_{i,j,K,E,F}$ at time t_k. In the case of $B_{i,j,K,E,F}$ independent of $\mathcal{Z}$ the value from time t_{k+1} is used.
- $w_{i,K,E}^{k}$ is the component of $\boldsymbol{w}_i$ on element K at time t_k. In the case of $\boldsymbol{w}_i$ independent of $\mathcal{Z}$ the value from the time t_{k+1} is used.
- $a_{i,j,K,E}^{k}$ is the component of $\boldsymbol{a}_{i,j}$ on element K at time level k. In the case of $\boldsymbol{a}_{i,j}$ independent of $\mathcal{Z}$ the value from the time t_{k+1} is used.
- $Z_{j,K}^{k+1}$ is the approximation of Z_j on element K at time t_{k+1}.
- $Z_{j,F}^{k+1}$ is the approximation of trace of Z_j on side F at time t_{k+1}.
- $Z_{i,j,E}^{k,upw}$ is the *upwinded* value of Z_j on side E at time t_k, details of this choice are provided in (3.37).

Similarly to (3.35), the upwind technique is used in (3.35) to ensure the stability of the scheme. The selection of the upwinded quantities $Z_{i,j,E}^{k,upw}$ is the same as in the case of the non-conservative representation of the advection term shown in (3.37).

On the interior sides $E = \mathcal{E}_{K_1} \cap \mathcal{E}_{K_2}$ common to two elements $K_1, K_2 \in \mathcal{K}_h$, we assume that the velocity terms are continuous over E in terms of their normal traces, i.e., their components are balanced

$$v_{i,K_1,E}^{k} + v_{i,K_2,E}^{k} = 0, \qquad \forall i \in \{1, \dots, n\}, \tag{3.40}$$

for details, see Section 3.6. The newly introduced *upwinded* quantities $m_{i,E}^{k,upw}$ are defined by

$$m_{i,E}^{k,upw} = \begin{cases} m_{i,K_1}^{k}, & \text{if } v_{i,K_1,E}^{k} > 0, \\ m_{i,K_2}^{k}, & \text{if } v_{i,K_2,E}^{k} > 0, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i \in \{1, \dots, n\}. \tag{3.41}$$

On the boundary sides $E \in \mathcal{E}_{K_1}$ belonging to one element $K_1 \in \mathcal{K}_h$ only, the value from the boundary condition is used instead of (3.37) in the case of $v_{i,K_1,E}^{k} < 0$. At the first time step $(k = 0)$, the values of $v_{i,K,E}^{0}$ are not known, therefore, the arithmetic average of the values from the neighboring elements is used

$$m_{i,E}^{0,upw} = \frac{1}{2}\left(m_{i,K_1}^{0} + m_{i,K_2}^{0}\right), \qquad \forall i \in \{1, \dots, n\}. \tag{3.42}$$

3.5.4 Integral I_4

The fourth term in (3.33) describing reactions using coefficient $\mathcal{R}$ is approximated by

$$\int_K \sum_{j=1}^{n} r_{i,j} Z_j \approx |K|_d \sum_{j=1}^{n} r_{i,j,K}^{k} Z_{j,K}^{k+1}, \tag{3.43}$$

where:

- $r_{i,j,K}^{k}$ is the approximation of coefficient $r_{i,j}$ on element K at time t_k. In the case of $r_{i,j}$ independent of $\mathcal{Z}$ the value from the time t_{k+1} is used.
- $Z_{j,K}^{k+1}$ is the approximation of Z_j on element K at time t_{k+1}.

3.5.5 Integral I_5

The fifth term in (3.33) describing the sinks and sources using coefficient $\mathcal{F}$ is approximated by

$$\int_K f_i \approx |K|_d f_i^k, \tag{3.44}$$

where:

$f_{i,K}^k$ is the approximation of coefficient f_i on element K at time t_k. In the case of f_i independent of $\mathcal{Z}$ the value from the time t_{k+1} is used.

Combining all the approximations shown in equations (3.34), (3.35), (3.39), (3.43), and (3.44), we obtain the following discretization of the original system (3.1), where the conservative velocity unknowns are eliminated and the only remaining unknowns are scalar functions Z_j at time t_{k+1} and their traces

$$\sum_{j=1}^{n} N_{i,j,K}^k \frac{|K|_d}{\Delta t_k}\left(Z_{j,K}^{k+1} - Z_{j,K}^k\right) + \sum_{j=1}^{n}\sum_{E\in K} u_{i,j,K,E}^k \left(Z_{i,j,E}^{k,upw} - Z_{j,K}^{k+1}\right) + \sum_{E\in\mathcal{E}_K} m_{i,E}^{k,upw}\left(\sum_{j=1}^{n}\left(B_{i,j,K,E}^k Z_{j,K}^{k+1} - \sum_{F\in\mathcal{E}_K} B_{i,j,K,E,F}^k Z_{j,F}^{k+1}\right) + w_{i,K,E}^k\right) + \sum_{j=1}^{n}\sum_{E\in\mathcal{E}_K} a_{i,j,K,E}^k Z_{i,j,E}^{k,upw} + |K|_d \sum_{j=1}^{n} r_{i,j,K}^k Z_{j,K}^{k+1} = |K|_d f_i^k. \tag{3.45}$$

3.5.6 Matrix-vector notation

On each element $K \in \mathcal{K}_h$ and $\forall E \in \mathcal{E}_K$, we gather the unknowns $Z_{j,K}^{k+1}$ and traces $Z_{j,F}^{k+1}$ into vectors $\boldsymbol{Z}_K^{k+1} = \left\{Z_j^{k+1}\right\}_{j=1}^n$ and $\boldsymbol{Z}_F^{k+1} = \left\{Z_{j,F}^{k+1}\right\}_{j=1}^n$, respectively. Then, (3.45) formulated $\forall i \in \{1,\dots,n\}$ can be written in a compact form as

$$\mathbb{Q}_K \boldsymbol{Z}_K^{k+1} = \sum_{F\in\mathcal{E}_K} \mathbb{R}_{K,F} \boldsymbol{Z}_F^{k+1} + \boldsymbol{R}_K, \qquad \forall K \in \mathcal{K}_h, \tag{3.46}$$

where the entries of matrices $\mathbb{Q}_K$ and $\mathbb{R}_{K,F}$ are given by

$$\{\mathbb{Q}_K\}_{i,j} = \frac{|K|_d}{\Delta t_k} N_{i,j,K}^k - \sum_{E\in\mathcal{E}_K} u_{i,j,K,E}^k + \sum_{E\in\mathcal{E}_K} m_{i,E}^{k,upw} B_{i,j,K,E}^k + |K|_d r_{i,j,K}^k, \qquad \forall i,j \in \{1,\dots,n\}, \quad \forall K \in \mathcal{K}_h, \tag{3.47a}$$

$$\{\mathbb{R}_{K,F}\}_{i,j} = \sum_{E\in\mathcal{E}_K} m_{i,E}^{k,upw} B_{i,j,K,E,F}^k, \quad \forall i,j \in \{1,\dots,n\}, \quad \forall F \in \mathcal{E}_K, \quad \forall K \in \mathcal{K}_h, \tag{3.47b}$$

and the components of the vector $\boldsymbol{R}_K$ are given by.

$$\{\boldsymbol{R}_K\}_i = |K|_d f_{i,K}^k + \frac{|K|_d}{\Delta t_k}\sum_{j=1}^{n} N_{i,j,K}^k Z_{j,K}^k - \sum_{E\in\mathcal{E}_K} m_{i,E}^{k,upw} w_{i,K,E}^k - \sum_{j=1}^{n}\sum_{E\in\mathcal{E}_K} Z_{i,j,E}^{k,upw}\left(a_{i,j,K,E}^k + u_{i,j,K,E}^k\right), \quad \forall i \in \{1,\dots,n\}, \quad \forall K \in \mathcal{K}_h. \tag{3.48}$$

We assume that the coefficients in (3.1) are given such that the matrix $\mathbb{Q}_K$ in (3.46) is non-singular $\forall K \in \mathcal{K}_h$.

3.5.7 Eliminating the cell averages

Under the assumption of non-singularity of $\mathbb{Q}_K$, (3.46) can be multiplied by its inversion from the left to obtain

$$\boldsymbol{Z}_K^{k+1} = \sum_{F \in \mathcal{E}_K} \mathbb{Q}_K^{-1} \mathbb{R}_{K,F} \boldsymbol{Z}_F^{k+1} + \mathbb{Q}_K^{-1} \boldsymbol{R}_K, \qquad \forall K \in \mathcal{K}_h, \tag{3.49}$$

which allows for expressing $\boldsymbol{Z}_K^{k+1}$ in terms of traces $\boldsymbol{Z}_F^{k+1}$ and, therefore, reducing the number of unknowns at the time t_{k+1}.

3.6 Mass balance on the interior sides

We assume that no mass is produced or lost on an interior side E common to elements K_1 and K_2 ($E = \mathcal{E}_{K_1} \cap \mathcal{E}_{K_2}$), and the advection terms are balanced over E by (3.19)

$$a_{i,j,K_1,E}^k + a_{i,j,K_2,E}^k = 0, \qquad \forall i, j \in \{1, \dots, n\}, \tag{3.50a}$$

$$u_{i,j,K_1,E}^k + u_{i,j,K_2,E}^k = 0, \qquad \forall i, j \in \{1, \dots, n\}. \tag{3.50b}$$

Then, using the properties of the basis functions (3.14) the balance of the normal component of the conservative flux $\boldsymbol{q}_i$ on side E can be written in a compact form as

$$q_{i,K_1,E}^{k+1} + q_{i,K_2,E}^{k+1} = 0. \tag{3.51}$$

In addition, using the definition $\boldsymbol{q}_i = m_i \boldsymbol{v}_i$ and the approximation of velocity terms given by (3.32), mass balance given by (3.51) can be formulated in terms of Z_j and their traces as

$$\sum_{\ell=1}^{2} m_{i,K_\ell,E}^k \left[\sum_{j=1}^{n} \left(B_{i,j,K,E} Z_{j,K_\ell}^{k+1} - \sum_{F \in \mathcal{E}_{K_\ell}} B_{i,j,K_\ell,E,F} Z_{j,F}^{k+1} \right) + w_{i,K_\ell,E} \right] = 0. \tag{3.52}$$

To ensure the stability of the numerical scheme, the single upwinded value $m_{i,E}^{k,upw}$ defined by (3.41) is used instead of $m_{i,K_1,E}^k$ and $m_{i,K_2,E}^k$ to obtain

$$m_{i,E}^{k,upw} \sum_{\ell=1}^{2} \left[\sum_{j=1}^{n} \left(B_{i,j,K,E} Z_{j,K_\ell}^{k+1} - \sum_{F \in \mathcal{E}_{K_\ell}} B_{i,j,K_\ell,E,F} Z_{j,F}^{k+1} \right) + w_{i,K_\ell,E} \right] = 0. \tag{3.53}$$

The mass balance equation (3.53) can degenerate when term $m_{i,E}^{k,upw}$ vanishes. This happens, for example, in the two-phase flow in porous media when one of the phases is not present in both elements sharing the side E. To avoid this situation that would otherwise require a special treatment the following approach to deal with the value of $m_{i,E}^{k,upw}$ is used. If $m_{i,E}^{k,upw} \neq 0$, it can be factored out from (3.53) to obtain

$$\sum_{\ell=1}^{2} \left[\sum_{j=1}^{n} \left(B_{i,j,K,E} Z_{j,K_\ell}^{k+1} - \sum_{F \in \mathcal{E}_{K_\ell}} B_{i,j,K_\ell,E,F} Z_{j,F}^{k+1} \right) + w_{i,K_\ell,E} \right] = 0. \tag{3.54}$$

For non-zero value of $m_{i,E}^{k,upw}$, the mobility can be eliminated from (3.53) which motivates using the same approach also for $m_{i,E}^{k,upw} = 0$. With zero mobility, the flux $\boldsymbol{q}_i$ defined by $\boldsymbol{q}_i = m_i \boldsymbol{v}_i$ satisfies the mass balance (3.51) on E and, at the same time, (3.54) is non-degenerate and can be used to obtain information about Z_{j,K_ℓ}^{k+1} and $Z_{j,F}^{k+1}$. Otherwise, with vanishing mobility, the

left-hand side of (3.52) would be identically zero and could not be used to obtain any information about the scalar unknowns, thus producing an ill-posed problem.

Overall, on the interior sides, (3.54) is used regardless of the value of $m_{i,E}^{k,upw}$. On the benchmark problems in Chapter 5, it is shown that this approach works and the numerical scheme produces correct and reliable results.

In addition, values of Z_{j,K_ℓ}^{k+1} in (3.54) are expressed in terms of traces $Z_{j,F}^{k+1}$ using (3.49) and the mass balance on each interior side is then expressed in terms of traces of Z_j, $j \in \{1, \dots, n\}$ only, reducing the number of unknowns needed to formulate this balance equation.

3.7 Boundary conditions

The mass balance (3.52) is given for all the interior sides. To obtain the linear system for all the unknown traces at new time t_{k+1}, the boundary sides need to be also resolved. The form of the equations on these sides depends on the boundary condition type. For the Dirichlet boundary condition for the i-th equation on the side E, the value of the trace is prescribed as

$$Z_{i,E}^{k+1} = Z_{i,E}^{k+1,Dir}, \tag{3.55}$$

where $Z_{i,E}^{k+1,Dir}$ is the value of the Dirichlet boundary condition on E at the time t_{k+1}.

For the Neumann boundary condition for the i-th equation on side E, the value of $v_{i,K,E}$ is prescribed using the expression (3.32)

$$\sum_{j=1}^{n} \left(B_{i,j,K,E} Z_{j,K_\ell}^{k+1} - \sum_{F \in \mathcal{E}_{K_\ell}} B_{i,j,K_\ell,E,F} Z_{j,F}^{k+1} \right) + w_{i,K_\ell,E} = v_{i,K,F}^{k+1,Neu}, \tag{3.56}$$

where $v_{i,K,F}^{k+1,Neu}$ is the value of the Neumann boundary condition on E at the time t_{k+1} and the cell values Z_{j,K_ℓ}^{k+1} are given in terms of traces $Z_{j,F}^{k+1}$ using (3.49).

As a result of the mass balance formulated on the interior sides (3.54) and boundary conditions expressions (3.55) and (3.56) for all equations $i \in \{i, \dots, n\}$, a single sparse linear system of algebraic equations for the unknown traces of Z_j at time t_{k+1} is obtained

$$\mathbb{A} \boldsymbol{Z}^{k+1} = \boldsymbol{b}, \tag{3.57}$$

with $\boldsymbol{Z}^{k+1} = \{\{Z_{j,F}^{k+1}\}_{j=1}^{n}\}_{F \in \mathcal{E}_h}$. Once the system (3.57) is solved, the cell averages $Z_{j,K}^{k+1}$ at the new time t_{k+1} are computed $\forall j \in \{1, \dots, n\}, \forall K \in \mathcal{K}_h$ using equation (3.49) which completes the transition to the next time level.

3.8 Properties of matrix $\mathbb{A}$

In this section, we investigate the properties of matrix $\mathbb{A}$ of the linear problem (3.57) obtained by the spatio-temporal discretization of the original system of partial differential equations (3.1).

We firstly present the details of the matrix assembly and the rows and columns ordering under which the block structure of matrix $\mathbb{A}$ and the role of the diagonal and off-diagonal blocks can be explored. In the case of a single equation, we order the rows of $\mathbb{A}$ in the same way as the unknowns. With this ordering, the first unknown is the trace of Z_1 on the first side E^1, and the first row of the matrix expresses the mass balance on side E^1, etc. In the case of a system of equations ($n > 1$ in (3.1)), the ordering is selected to produce a block structure of $\mathbb{A}$. The unknowns are ordered block-wise, where the block size is the number of all sides $\#\mathcal{E}_h$. First, we include all traces of Z_1, then the traces of Z_2, etc. The rows are then ordered analogously. First,

we include all the mass balances for unknown Z_1 that are ordered in the same way as for a single equation, then the mass balances for unknown Z_2, etc. In Figure 3.1, the structure of $\mathbb{A}$ for this ordering is for a system of two equations illustrated.

In the case of a system of equations, we explore the role of the blocks of $\mathbb{A}$ in the original system (3.1). The i-th diagonal block of $\mathbb{A}$ represents the contribution of the i-th unknown to the mass balance for that unknown e.g. the role of unknowns Z_1 in the mass balance equation for Z_1, etc. The off-diagonal blocks then represent the contribution of the remaining unknowns to that equation (contribution to mass balance for Z_1 by unknowns Z_2, etc.). In other words, for the selected ordering of rows and columns, the off-diagonal terms represent the coupling among the equations.

$$\left(\begin{array}{cccccc}
a_{1,1} & a_{1,2} & \cdots & a_{1,N_E+1} & a_{1,N_E+2} & \cdots \\
a_{2,1} & a_{2,2} & \cdots & a_{2,N_E+1} & a_{2,N_E+2} & \cdots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\
a_{N_E+1,1} & a_{N_E+1,2} & \cdots & a_{N_E+1,N_E+1} & a_{N_E+1,N_E+2} & \cdots \\
a_{N_E+2,1} & a_{N_E+2,2} & \cdots & a_{N_E+2,N_E+1} & a_{N_E+2,N_E+2} & \cdots \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots
\end{array}\right)$$

Mass balances for unknown Z_1

Mass balances for unknown Z_2

Traces of Z_1 | Traces of Z_2

Figure 3.1: Illustration of the structure of $\mathbb{A} = \{a_{i,j}\}_{i,j=1}^{nN_E}$ in Eq. (3.57) for the system of two equations ($n = 2$).

From the construction of $\mathbb{A}$, we can also determine the number of possibly non-zero entries in a row for each mesh type, i.e., the number of entries in each matrix row that are computed using the scheme. Note that the computed value can be numerically equal to zero for certain problems. In contrast with, for example, Lagrangian finite elements with degrees of freedom located on element vertices, the number of non-zero entries in each row does not depend on the realization of the given type of mesh. Each row of $\mathbb{A}$ expresses the mass balance equation for one unknown on a single side E as described in Section 3.6 and the balance equation is formulated using the traces of all unknowns on all sides of the two neighboring elements sharing this side E. This results in the following numbers of non-zero entries in each matrix row: $5n$ for triangular meshes, $7n$ for rectangular and tetrahedral meshes, and $11n$ for cuboidal meshes. The sides and traces needed to formulate the mass balance on a given side E for the triangular mesh are shown in Figure 3.2.

The numbers listed above hold for the internal sides, where (3.54) is used. On side E, where the Neumann boundary condition is prescribed by (3.56), the value of $v_{i,K,E}$ is given in terms of traces on all the sides of the neighboring element K. This gives the following number of non-zero entries in each matrix row: $3n$ for triangular meshes, $4n$ for rectangular and tetrahedral meshes, and $6n$ for cuboidal meshes. On side E, where the Dirichlet boundary condition is prescribed by (3.55), the value of the trace is given which results in a single non-zero entry in the matrix row, regardless of the mesh element type and the dimension of the problem.

The alternative approach to deal with the Dirichlet boundary conditions is their elimination from the linear system (3.57). This procedure results in a symmetric structure of $\mathbb{A}$ that can be advantageous for certain linear solvers.

The symmetry or non-symmetry of $\mathbb{A}$ cannot be determined without specifying all the coefficients for the problem represented by (3.1). However, for most of the problems considered in this work, the resulting matrix $\mathbb{A}$ is non-symmetric. The non-symmetry can be attributed to employing the upwind technique in (3.41) and (3.37) and to the possible non-symmetry of the coefficient matrices $\mathcal{N}$, $\mathcal{U}$, $\mathcal{A}$, $\mathcal{D}$, and $\mathcal{R}$ as shown later in Chapters 5 and 7. An example of

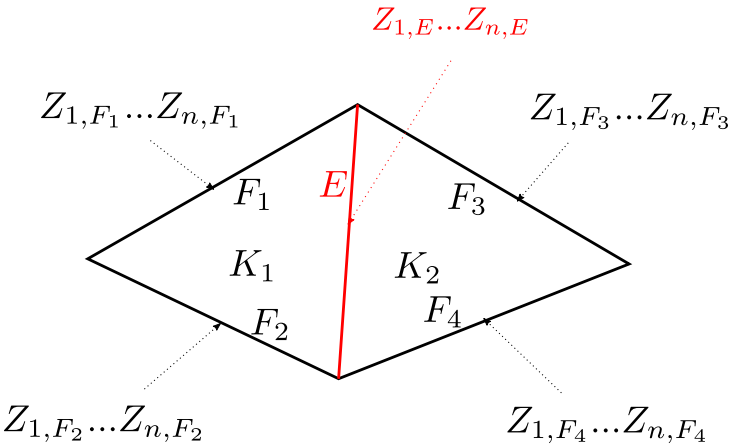


Figure 3.2: Mass balance on internal side E common to elements K_1 and K_2 in 2D and all the sides needed to formulate the mass balance on E: E and $F_1, \dots F_4$.

a problem resulting in symmetric $\mathbb{A}$ is the Laplace problem used as one of the benchmarks for parallel implementation later in Chapter 6.

3.9 Mass Lumping

Local coefficient matrices $\mathbb{H}_{i,j,K}$ introduced by (3.29) as integrals over mesh elements can be computed using two approaches: their entries can be either computed exactly or approximated. Surprisingly, the method selected to compute $\mathbb{H}_{i,j,K}$ has a significant impact on the accuracy of the numerical scheme when used for solving problems in heterogeneous porous media as will be shown later in Chapter 5.

For the exact computation of the integral, we remark that the selected basis functions of $\mathbf{RTN}_0(K)$ presented in Section 3.3.2 are the first-order polynomial vector functions. Therefore, the entries of $\mathbb{H}_{i,j,K}$ are given as the integrals of second-order polynomials over $K \in \mathcal{K}_h$. These integrals can be computed exactly using the transformation to a reference element. This approach is in the following referred to as the exact integration (EI) variant of the numerical scheme.

Instead of the exact computation, the values of the integrals can be approximated. The mass lumping approach based on the method proposed in [23] is employed and the values of the integrals in (3.29) are computed using a quadrature rule. For all $K \in \mathcal{K}_h$, the integral of an arbitrary scalar function $z(\boldsymbol{x})$ is computed as

$$\int_K z(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx \frac{|K|_d}{N_v} \sum_{p=1}^{N_v} z(\boldsymbol{x}_p), \tag{3.58}$$

where $\boldsymbol{x}_p$ is the position vector of the p-th vertex of element K and N_v is the number of vertices of the element K: $N_v = 3$ for triangle, $N_v = 4$ for tetrahedron and rectangle, and $N_v = 6$ for cuboid. This approach is in the following referred to as the mass lumping (ML) variant of the numerical scheme.

3.10 Computational algorithm

To summarize the description of the numerical scheme in a compact form, we present the steps needed to initialize the scheme and proceed from time t_k to time t_{k+1} in the form of Algorithms 1 and 2, respectively.

Algorithm 1 Initialization of the numerical scheme.

1. Set $t_0 = 0$, $k = 0$, and set Δt to a given fixed value.
2. Using the initial conditions, set $Z^0_{j,K}$, $\forall K \in \mathcal{K}_h$ and $\forall j \in \{1, ..., n\}$.
3. Initialize coefficients $\mathcal{N}$, $\mathcal{U}$, $\mathcal{M}$, $\mathcal{D}$, $\mathcal{W}$, $\mathcal{A}$, $\mathcal{R}$, and $\mathcal{F}$ in (3.1).
4. Set the initial *upwinded* values
   - For the interior sides, $m^{0,upw}_{i,E}$ using (3.42) and $Z^{0,upw}_{i,j,E}$ using (3.38), $\forall i, j \in \{1, ..., n\}$.
   - For the boundary sides, use the information from the boundary conditions.

Algorithm 2 The main computational loop of the numerical scheme. Transition from time t_k to t_{k+1}.

1. Compute local matrices $\mathbb{H}_{i,j,K}$ defined by (3.29) using either the EI or the ML approach, $\forall i, j \in \{1, ..., n\}$, $\forall K \in \mathcal{K}_h$.
2. Compute inversions of $\mathbb{H}_{i,j,K}$ to obtain $\mathbb{B}_{i,j,K}$, $\forall i, j \in \{1, ..., n\}$, $\forall K \in \mathcal{K}_h$.
3. Compute the local matrices $\mathbb{Q}_K$ and $\mathbb{R}_{K,F}$ given by (3.47) and $\mathbb{Q}_K^{-1}\mathbb{R}_{K,F}$, $\forall F \in \mathcal{E}_K$, $\forall K \in \mathcal{K}_h$.
4. Compute vectors $\boldsymbol{R}_K$ given by (3.48), $\forall K \in \mathcal{K}_h$.
5. Assembly linear system (3.57) for unknown traces using the mass balance equations on the interior sides (3.54) and boundary conditions (3.55) and (3.56).
6. Solve the linear system (3.57) to obtain traces $\boldsymbol{Z}^{k+1}$ at the new time level.
7. Using (3.49), compute the values of $\boldsymbol{Z}^{k+1}_K$ from the traces $\boldsymbol{Z}^{k+1}_E$, $\forall K \in \mathcal{K}_h$.
8. Using (3.32), compute the coefficients $v^{k+1}_{i,K,E}$, $\forall i \in \{1, \ldots, n\}$, $\forall E \in \mathcal{E}_K$, $\forall K \in \mathcal{K}_h$.
9. Update the coefficients in (3.1) ($\mathcal{N}$, $\mathcal{U}$, $\mathcal{M}$, $\mathcal{D}$, $\mathcal{W}$, $\mathcal{A}$, $\mathcal{R}$, $\mathcal{F}$) that are functions of the unknowns $\mathcal{Z}$ using the computed values of $\boldsymbol{Z}^{k+1}_K$ and $\boldsymbol{Z}^{k+1}_E$.
10. Update the upwinded quantities
    - For the interior sides, $m^{k+1,upw}_{i,E}$ using (3.41) and $Z^{k+1,upw}_{i,j,E}$ using (3.37), $\forall i, j \in \{1, ..., n\}$.
    - For the boundary sides, use the information from the boundary conditions.
11. Set $t_{k+1} = t_k + \Delta t$ and repeat the loop from Step 1 until the final time t_{fin} is reached.

3.11 Steady problem

The time dependency of the general problem (3.1) is described by the coefficient $\mathcal{N}$. Solving a general steady problem represented by (3.1) is a challenging task requiring a different approach than the one presented above and as such is beyond the scope of this work. However, the numerical scheme presented here can be easily modified for the solution of a simplified steady linear problem in the form

$$\nabla \cdot \left[m_i \left(-\sum_{j=1}^{n} \mathbb{D}_{i,j} \nabla Z_j + \boldsymbol{w}_i \right) \right] + \sum_{j=1}^{n} r_{i,j} Z_j = f_i, \tag{3.59}$$

i.e., steady equation (3.1) without the advection terms. The remaining non-zero coefficients are not functions of the unknowns $\mathcal{Z}$ but can depend on spatial coordinate $\boldsymbol{x}$.

The procedure carried out with the discretization of the diffusion term in Section 3.4 is done without introducing temporal discretization, therefore, remains the same for the steady case.

For the approximation of conservation laws, we follow the procedure presented in Section 3.5 for the weak formulation (3.12) where we omit the superscripts describing the time levels and do not consider zero coefficients. For the selection of the upwinded value $m_{i,E}^{upw}$, we use the same approach that was used for the first time step in the case of an unsteady problem: for the interior sides $m_{i,E}^{upw}$ is obtained using (3.42) and for the boundary sides, the value from boundary condition is used. The coefficients $\mathcal{U}$ and $\mathcal{A}$ are zero, therefore, the upwinded values of $Z_{i,j,E}^{upw}$ are not needed. As a result, similarly to (3.46), we can formulate the conservation laws in a matrix-vector notation

$$\mathbb{Q}_K \boldsymbol{Z}_K = \sum_{F \in \mathcal{E}_K} \mathbb{R}_{K,F} \boldsymbol{Z}_F + \boldsymbol{R}_K, \tag{3.60}$$

where the entries of matrices $\mathbb{Q}_K$ and $\mathbb{R}_{K,F}$ are given by

$$\{\mathbb{Q}_K\}_{i,j} = \sum_{E \in \mathcal{E}_K} m_{i,E}^{upw} B_{i,j,K,E} + |K|_d r_{i,j,K}, \quad \forall i,j \in \{1,\dots,n\}, \quad \forall K \in \mathcal{K}_h, \tag{3.61a}$$

$$\{\mathbb{R}_{K,F}\}_{i,j} = \sum_{E \in \mathcal{E}_K} m_{i,E}^{upw} B_{i,j,K,E,F}, \quad \forall i,j \in \{1,\dots,n\}, \quad \forall F \in \mathcal{E}_K, \quad \forall K \in \mathcal{K}_h, \tag{3.61b}$$

and the components of vector $\boldsymbol{R}_K$ are given by

$$\{\boldsymbol{R}_K\}_i = |K|_d f_{i,K} - \sum_{E \in \mathcal{E}_K} m_{i,E}^{upw} w_{i,K,E}, \quad \forall i \in \{1,\dots,n\}, \quad \forall K \in \mathcal{K}_h. \tag{3.62}$$

Similarly to (3.49), under the assumption of non-singularity of $\mathbb{Q}_K$, we obtain

$$\boldsymbol{Z}_K = \sum_{F \in \mathcal{E}_K} \mathbb{Q}_K^{-1} \mathbb{R}_{K,F} \boldsymbol{Z}_F + \mathbb{Q}_K^{-1} \boldsymbol{R}_K, \quad \forall K \in \mathcal{K}_h. \tag{3.63}$$

Then, we follow the procedure described in Section 3.6 to formulate the mass balance on all sides, include the boundary conditions and assembly matrix $\mathbb{A}$. In this case, the solution of (3.57) is the solution of the solved steady problem.

To summarize the description of the numerical scheme for the solution of a simplified steady linear problem in a compact form, we present the steps in the form of Algorithm 3.

Algorithm 3 Numerical scheme for a steady linear problem.

1. Set the *upwinded* values
   - For the interior sides, $m_{i,E}^{upw} = 1/2\,(m_{i,K_1} + m_{i,K_2})$, $\forall i \in \{1, \ldots, n\}$.
   - For the boundary sides, use the information from the boundary conditions.
2. Compute local matrices $\mathbb{H}_{i,j,K}$ defined by (3.29) using either the EI or the ML approach, $\forall i, j \in \{1, ..., n\}$, $\forall K \in \mathcal{K}_h$.
3. Compute inversions of $\mathbb{H}_{i,j,K}$ to obtain $\mathbb{B}_{i,j,K}$, $\forall i, j \in \{1, ..., n\}$, $\forall K \in \mathcal{K}_h$.
4. Compute the local matrices $\mathbb{Q}_K$ and $\mathbb{R}_{K,F}$ given by (3.61) and $\mathbb{Q}_K^{-1}\mathbb{R}_{K,F}$, $\forall F \in \mathcal{E}_K$, $\forall K \in \mathcal{K}_h$.
5. Compute vectors $\boldsymbol{R}_K$ given by (3.62), $\forall K \in \mathcal{K}_h$.
6. Assembly linear system (3.57) for unknown traces using the mass balance equations on the interior sides (3.54) and boundary conditions (3.55) and (3.56).
7. Solve the linear system (3.57) to obtain traces $\boldsymbol{Z}$.
8. Using (3.63), compute the values of $\boldsymbol{Z}_K$ from the traces $\boldsymbol{Z}_E$, $\forall K \in \mathcal{K}_h$.
9. Using (3.32), compute the coefficients $v_{i,K,E}$, $\forall i \in \{1, \ldots, n\}$, $\forall E \in \mathcal{E}_K$, $\forall K \in \mathcal{K}_h$.

3.12 Chapter summary

In this chapter, we presented the numerical scheme based on the MHFEM spatial discretization with semi-implicit discretization in time designed to solve a general system of coupled non-linear advection-diffusion equations in the coefficient form covering the range of problems considered in this work. The complete description of the scheme was published in [43] and details about the mass lumping technique were published in [102].

The initialization of the numerical scheme and the main computational loop describing the transition from time t_k to t_{k+1} are summarized in Algorithms 1 and 2, respectively. The solution of a simplified steady linear problem is summarized in Algorithm 3.

Implementation and parallelization 4

In this chapter, the implementation of the numerical scheme in `C++` is described emphasizing the parallel implementation using `MPI` [100]. We focus on the implementation of the scheme for unsteady problems summarized in the previous chapter in Algorithms 1 and 2. The modifications for the solution of a steady problem are straightforward as the steps of Algorithm 3 summarizing the solution of a steady problem are comprised in either Algorithm 1 or 2.

The chapter is structured as follows. We start with the description of the serial implementation in Section 4.1. The serial implementation is usually advantageous for problems on coarse meshes resulting in smaller sizes of matrix $\mathbb{A}$ in (3.57) when the solution of the linear problem (3.57) in step 6 of Algorithm 2 is not too computationally demanding. Variants of the scheme using both direct and iterative solvers for the linear problems (3.57) are implemented. However, for finer meshes, matrix $\mathbb{A}$ in (3.57) becomes large and the solution of the linear system in step 6 of Algorithm 2 dominates the computational time which motivates to speedup the computation by employing more than one CPU core. When a certain size of the problem is reached, the serial variant of the method can no longer be used due to hardware requirements and parallel methods are essential to solving these problems.

In Section 4.2, we summarize the main ideas of the domain decomposition approach with non-overlapping subdomains. The reduction of the original linear system 3.57 to the Schur complement system and the iterative solution of such a system in subassembled form is described in Section 4.3. The BDDC preconditioner for the iterative solution of the Schur complement system is presented in Section 4.4 and its parallel implementation is described in Section 4.5.

4.1 Serial implementation

The serial implementation of the scheme following the steps of Algorithms 1 and 2 is straightforward. All the steps requiring the computation on mesh elements $K \in \mathcal{K}_h$ and sides $E \in \mathcal{E}_h$ are resolved sequentially. Based on the selected linear solver, the sparse linear system matrix $\mathbb{A}$ in (3.57) is assembled and stored in either the compressed sparse row (CSR) or column (CSC) formats. The position of non-zero entries of the matrix are represented by pairs of indices $NZ(\mathbb{A})$: $(i,k) \in NZ(\mathbb{A}) \Leftrightarrow$ matrix $\mathbb{A}$ has non-zero entry in row i and column k.

Variants of the method utilizing both a direct solver based on a suitable factorization of the matrix $\mathbb{A}$ and an iterative solver based on Krylov subspaces are implemented.

In general, the discretization of (3.1) leads to non-symmetric matrix $\mathbb{A}$. The direct solvers are in this case based on the LU factorization with partial pivotation. In this work, the numerical library `UMFPACK` [29] is used to solve the system with the non-symmetric matrix $\mathbb{A}$ stored in

the CSC format. The preconditioned biconjugate gradient stabilized method (BiCGStab) [115] summarized in Algorithm 4 is implemented as the iterative solver in this case with the incomplete LU factorization ILU(0) as a preconditioner. The construction of ILU(0) preconditioner [97] is summarized in Algorithm 5, the procedure follows the traditional LU decomposition but the non-zero entries that would change the structure of the original matrix $\mathbb{A}$ are dropped. In contrast to the direct solver, the CSR format is advantageous for the implementation of matrix-vector product required by iterative methods.

In special cases, the discretization of (3.1) can lead to symmetric matrix $\mathbb{A}$, for example, for the Laplace problem later in Chapter 6. In this case, the direct solvers can employ a simpler Cholesky decomposition instead of the LU one. In this work, the numerical library `CHOLMOD` [24] is used to solve the system with the symmetric matrix $\mathbb{A}$ stored in the CSC format. The preconditioned conjugate gradient method (PCG) [27] summarized in Algorithm 6 analogically with the incomplete Cholesky decomposition IC(0) [97] summarized in Algorithm 7 as a preconditioner is used for the matrix stored in the CSR format.

Algorithm 4 Preconditioned biconjugate gradient stabilized method for the solution of system $\mathbb{A}\boldsymbol{x} = \boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$. The preconditioner is represented by matrix $\mathbb{M}^{-1}$. Adapted from [115].

Compute $\boldsymbol{r}_0 = \boldsymbol{b} - \mathbb{A}\boldsymbol{x}_0$, set $\boldsymbol{r}_0^*$ arbitrary such that $(\boldsymbol{r}_0, \boldsymbol{r}_0^*) \neq 0$
$\rho_0 = \alpha = \omega_0 = 1$
$\boldsymbol{v}_0 = \boldsymbol{p}_0 = \boldsymbol{0}$
for $i = 0, 1, \ldots$ until convergence **do**
    $\rho_i = (\boldsymbol{r}_0^*, \boldsymbol{r}_{i-1})$
    $\beta = \frac{\rho_i}{\rho_{i-1}} \frac{\alpha}{\omega_{i-1}}$
    $\boldsymbol{p}_i = \boldsymbol{r}_{i-1} + \beta(\boldsymbol{p}_{i-1} - \omega_{i-1}\boldsymbol{v}_{i-1})$
    $\boldsymbol{y} = \mathbb{M}^{-1}\boldsymbol{p}_i$ ▷ Application of the preconditioner
    $\boldsymbol{v}_i = \mathbb{A}\boldsymbol{y}$
    $\alpha = \frac{\rho_i}{(\boldsymbol{r}_0^*, \boldsymbol{v}_i)}$
    $\boldsymbol{s} = \boldsymbol{r}_{i-1} - \alpha\boldsymbol{v}_i$
    $\boldsymbol{z} = \mathbb{M}^{-1}\boldsymbol{s}_i$ ▷ Application of the preconditioner
    $\boldsymbol{t} = \mathbb{A}\boldsymbol{z}$
    $\omega_i = \frac{(\boldsymbol{t}, \boldsymbol{s})}{(\boldsymbol{t}, \boldsymbol{t})}$
    $\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + \alpha\boldsymbol{y} + \omega_i\boldsymbol{z}$
    $\boldsymbol{r}_i = \boldsymbol{s} - \omega_i\boldsymbol{t}$
end for

Algorithm 5 Construction of incomplete LU factorization ILU(0) for matrix $\mathbb{A} = \{a_{i,j}\}$ of rank $\hat{n}$. Adapted from [97].

for $i = 2, \ldots, \hat{n}$ **do**
    for $k = 1, \ldots, i-1$ and $(i,k) \in NZ(\mathbb{A})$ **do**
        Compute $a_{i,k} = \frac{a_{i,k}}{a_{k,k}}$
        for $j = k+1, \ldots, \hat{n}$ and for $(i,j) \in NZ(\mathbb{A})$ **do**
            Compute $a_{i,j} = a_{i,j} - a_{i,k}a_{k,j}$
        end for
    end for
end for

Algorithm 6 Preconditioned conjugate gradient method for the solution of system $\mathbb{A}\boldsymbol{x} = \boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$. The preconditioner is represented by matrix $\mathbb{M}^{-1}$. Adapted from [27].

Compute $\boldsymbol{r}_0 = \boldsymbol{b} - \mathbb{A}\boldsymbol{x}_0$, $\boldsymbol{z}_0 = \mathbb{M}^{-1}\boldsymbol{r}_0$, and $\boldsymbol{p}_0 = \boldsymbol{z}_0$
for $j = 0, 1, \ldots$ until convergence **do**
  $\alpha_j = \frac{(\boldsymbol{r}_j, \boldsymbol{z}_j)}{(\mathbb{A}\boldsymbol{p}_j, \boldsymbol{p}_j)}$
  $\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \alpha_j \boldsymbol{p}_j$
  $\boldsymbol{r}_{j+1} = \boldsymbol{r}_j - \alpha_j \mathbb{A}\boldsymbol{p}_j$
  $\boldsymbol{z}_{j+1} = \mathbb{M}^{-1}\boldsymbol{r}_{j+1}$ ▷ Application of the preconditioner
  $\beta_j = \frac{(\boldsymbol{r}_{j+1}, \boldsymbol{z}_{j+1})}{(\boldsymbol{r}_j, \boldsymbol{z}_j)}$
  $\boldsymbol{p}_{j+1} = \boldsymbol{z}_{j+1} + \beta_j \boldsymbol{p}_j$
end for

Algorithm 7 Construction of incomplete Cholesky factorization IC(0) for matrix $\mathbb{A} = \{a_{i,j}\}$ of rank $\hat{n}$. Adapted from [97].

for $k = 1, \ldots, \hat{n} - 1$ **do**
  Compute $a_{k,k} = \sqrt{a_{k,k}}$
  for $j = k + 1, \ldots, \hat{n}$ and $(j, k) \in NZ(\mathbb{A})$ **do**
    Compute $a_{j,k} = \frac{a_{j,k}}{a_{k,k}}$
  end for
  for $j = k + 1, \ldots, \hat{n}$ and for $(j, k) \in NZ(\mathbb{A})$ **do**
    for $i = j, \ldots, \hat{n}$ and $(i, j) \in NZ(\mathbb{A})$ **do**
      Compute $a_{i,j} = a_{i,j} - a_{i,k}a_{j,k}$
    end for
  end for
end for
$a_{\hat{n},\hat{n}} = \sqrt{a_{\hat{n},\hat{n}}}$

4.1.1 Preconditioners

In general, a preconditioner is a form of transforming the original problem into an equivalent one with the same solution and which should be easier to solve by the selected iterative method [97]. Employing the preconditioner described by a matrix $\mathbb{M}$ from left is formally equivalent to solving a modified problem

$$\mathbb{M}^{-1}\mathbb{A}\boldsymbol{Z} = \mathbb{M}^{-1}\boldsymbol{b}, \tag{4.1}$$

instead of the original (3.57). The action of $\mathbb{M}^{-1}$ requires a solution of a linear system with matrix $\mathbb{M}$, therefore, one of the desired properties of $\mathbb{M}$ is that the solution of such a problem is inexpensive.

For the ILU(0) and IC(0) preconditioners considered in this work, $\mathbb{M}$ is constructed in a factorized form $\mathbb{M} = \mathbb{L}\mathbb{U}$ and $\mathbb{M} = \mathbb{L}\mathbb{L}^T$, respectively, where $\mathbb{L}$ is a lower triangular matrix and $\mathbb{U}$ is an upper triangular matrix. Positions and values of the non-zero entries of the matrices are given in Algorithms 5 and 7 for ILU(0) and IC(0), respectively. The application of preconditioner in Algorithms 4 and 6 then requires the solution of two linear systems with triangular matrices which can be done by the backward and forward substitutions.

4.2 Domain Decomposition

The parallel approach used in this work is based on the domain decomposition method with non-overlapping subdomains [110]. The decomposition into subdomains is advantageous both for the local computations on elements $K \in \mathcal{K}_h$ such as steps 1 - 4 of Algorithm 2 and also for the solution of the linear system (3.57) in step 6. The local computations on $K \in \mathcal{K}_h$ are independent, therefore, the efficient parallelization is straightforward. On the other hand, the parallelization of the linear solver is a more demanding task.

The description of the parallel implementation is divided into two blocks. One block contains the initialization of the scheme summarized in Algorithm 1, the preparatory steps to assembly matrix $\mathbb{A}$, and updating the information when the solution is obtained summarized Algorithm 2 except for steps 5 and 6. The other block contains the remaining two steps: the assembly of $\mathbb{A}$ and the solution of (3.57).

The first block starts with a decomposition of a computational domain Ω into N non-overlapping subdomains denoted Ω_i, $i \in \{1, \ldots, n\}$. The division of the numerical mesh is carried out such that each mesh element $K \in \mathcal{K}_h$ belongs to exactly one subdomain Ω_i. The MHFEM-based scheme uses values located on also on mesh sides and each interior side belongs to exactly two elements. Sides $E \in \mathcal{E}_h$ common to two elements belonging into two different subdomains need to be duplicated in the corresponding subdomains.

For optimal parallel performance, the subdomains should contain approximately the same number of mesh elements and, as a result, approximately the same number of degrees of freedom. Another desired property of the decomposition is to minimize the number of sides common to two different subdomains. The set of all sides shared by two different subdomains is, in the context of this chapter, referred to as the *interface*. The values corresponding to the sides on the interface are needed in both subdomains and need to be transferred between the subdomains during the computation. Thus, limiting their number reduces the communication between subdomains during the computation.

Finding the decomposition satisfying these conditions can be a challenging task, especially for unstructured meshes. In this work, a graph partitioning-based algorithm provided by the `METIS` library [70] is used to obtain such decomposition for unstructured meshes. However, for structured meshes consisting of rectangular elements on the rectangular domain and cuboidal elements on the cuboidal domain in 2D and 3D, respectively, the division into a certain number of rectangular or cuboidal subdomains can be obtained by overlay of the original mesh with a coarse grid. All the mesh elements laying in a single coarse grid cell then form one subdomain.

The aforementioned divisions of the structured meshes satisfy the desired properties and both approaches for structured meshes are compared later in Chapter 6 where the differences resulting from these two decompositions are discussed.

Once the division into subdomains is obtained, each subdomain is mapped to a single `MPI` process. This process then handles the corresponding part of the computational mesh and all the quantities required by the numerical scheme belonging to this subdomain. The implementation of the steps of Algorithms 1 and 2 that do not require the information from the other subdomains is straightforward as these steps are done independently by each `MPI` process with no communication needed.

4.2.1 Upwind

The communication between the processes is needed for the computation of the upwinded quantities $m_{i,E}^{0,upw}$ and $Z_{i,j,E}^{0,upw}$ during the initialization of the scheme in step 4 of Algorithm 1 and for the update of $m_{i,E}^{k,upw}$ and $Z_{i,j,E}^{k,upw}$ at each time step in step 10 of Algorithm 2 when the

upwinded quantity is computed on the interface side E. For these sides, only one of the two values located on elements K_1 and K_2 required by (3.37) or (3.41) is available for the given process. The type of communication and the structure of data transferred in these steps is the same, therefore, the unified description for both upwind procedures is presented in this section.

It would be too demanding to request individual values from other processes just in time when needed by evaluation of (3.37) or (3.41) on a given side and, in the case of an unsteady problem, the type of this communication (sending or receiving), can change during the computation as the sign of $u^k_{i,j,K_\bullet,E} + a^k_{i,j,K_\bullet,E}$ or $v^k_{i,K_\bullet,E}$ can change between the time steps. Instead, at the start of each step, a process creates an array of values that could be needed by other processes and an array of values that will be received from other processes. These arrays of values are then exchanged between the processes using the `MPI` operations `Isend` and `Ireceive`. The values from other processes are stored in an artificial array and the operations (3.37) or (3.41) are done locally using this array when the values from the other subdomains are required. Note that the type of this communication remains the same during the whole computation regardless of the sign of the coefficients $u^k_{i,j,K_\bullet,E} + a^k_{i,j,K_\bullet,E}$ and $v^k_{i,K_\bullet,E}$.

4.3 Schur complement system

In the following sections, we focus on the remaining steps 5 and 6 of Algorithm 2: the assembly of $\mathbb{A}$ and the solution of linear system $\mathbb{A}\boldsymbol{Z} = \boldsymbol{b}$ given by (3.57)

We start with the description of the iterative substructuring. Although it is quite a standard procedure in the domain decomposition literature (see e.g. [110]) we recall it briefly for clarity.

Based on the decomposition into N non-overlapping subdomains Ω_i, $i \in \{1, \dots, N\}$, the unknowns in the system (3.57) which are located on the mesh sides are divided into two groups: those located at the interior sides (and possibly the sides forming a physical boundary of computational domain Ω) and those located at the interface sides denoted by superscript I and Γ, respectively. In the following, all quantities corresponding to subdomain Ω_i are denoted by subscript i, $i \in \{1, \dots, N\}$.

The equations and unknowns in (3.57) are formally reordered: first the interior unknowns, then the interface ones, to obtain the following block partitioning of matrix $\mathbb{A}$

$$\begin{pmatrix} \mathbb{A}^{II} & \mathbb{A}^{I\Gamma} \\ \mathbb{A}^{\Gamma I} & \mathbb{A}^{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \boldsymbol{Z}^{k+1,I} \\ \boldsymbol{Z}^{k+1,\Gamma} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b}^I \\ \boldsymbol{b}^\Gamma \end{pmatrix}, \tag{4.2}$$

where $\mathbb{A}^{II}$ corresponds to the interior sides of the subdomains, $\mathbb{A}^{\Gamma\Gamma}$ corresponds to the interface sides, and $\mathbb{A}^{I\Gamma}$ and $\mathbb{A}^{\Gamma I}$ represent the coupling between these two parts.

The interior degrees of freedom $\boldsymbol{Z}^{k+1,I}$ in (4.2) are eliminated to obtain the reduced system for the Schur complement $\mathbb{S}$ of the interior unknowns only [110]

$$\mathbb{S}\boldsymbol{Z}^{k+1,\Gamma} = \hat{\boldsymbol{b}}, \tag{4.3a}$$

$$\mathbb{S} = \mathbb{A}^{\Gamma\Gamma} - \mathbb{A}^{\Gamma I}\left(\mathbb{A}^{II}\right)^{-1}\mathbb{A}^{I\Gamma}, \tag{4.3b}$$

$$\hat{\boldsymbol{b}} = \boldsymbol{b}^\Gamma - \mathbb{A}^{\Gamma I}\left(\mathbb{A}^{II}\right)^{-1}\boldsymbol{b}^I. \tag{4.3c}$$

When the system (4.3a) is solved, the interior unknowns $\boldsymbol{Z}^{k+1,I}$ are recovered using

$$\boldsymbol{Z}^{k+1,I} = \left(\mathbb{A}^{II}\right)^{-1}\left(\boldsymbol{b}^I - \mathbb{A}^{I\Gamma}\boldsymbol{Z}^{k+1,\Gamma}\right). \tag{4.4}$$

Note that $\mathbb{A}^{II}$ is a block diagonal matrix with blocks corresponding to individual subdomains.

The main advantage of this approach is that the reduced system (4.3a) is typically much smaller than the original one (3.57) and the elimination of the interior degrees of freedom and their recovery from the interface ones once the Schur complement system is solved can be done independently on each subdomain, i.e., in parallel. Explicit construction of (4.3a) would be too computationally and memory demanding. Instead, the action of $\left(\mathbb{A}^{II}\right)^{-1}$ on vector in (4.3b), (4.3c), and (4.4) is obtained as the solution of a linear system with matrix $\mathbb{A}^{II}$.

4.3.1 Subassembled form of matrix $\mathbb{S}$

In this section, we show the subassembled form of matrix $\mathbb{S}$ defined in (4.3b) which is convenient for performing the operations with Schur complement in parallel.

On the subdomain Ω_i, the local part of the matrix $\mathbb{A}$ in (3.57) denoted by $\mathbb{A}_i$ is assembled following the same procedure as for the overall matrix $\mathbb{A}$, here, using only the sides of Ω_i. From these local matrices $\mathbb{A}_i$, the complete matrix $\mathbb{A}$ can be formally assembled as

$$\mathbb{A} = \sum_{i=1}^{N} \mathbb{R}_{\Omega,i}^T \mathbb{A}_i \mathbb{R}_{\Omega,i}, \tag{4.5}$$

where $\mathbb{R}_{\Omega,i}$ is the rectangular matrix consisting of ones and zeros, which represents the restriction of the vector of all degrees of freedom to those located at the i-th subdomain.

Similarly to (4.2) for the complete system, the unknowns in each subdomain Ω_i are divided into two groups: those located at the interior sides and possibly the physical boundary of Ω and those located at the intersection of the subdomain boundary with the interface.

We can reorder the equations and unknowns in each local matrix $\mathbb{A}_i$ similarly to (4.2): first the interior unknowns, then the interface ones to obtain a block partitioning of $\mathbb{A}_i$ in the form

$$\mathbb{A}_i = \begin{pmatrix} \mathbb{A}_i^{II} & \mathbb{A}_i^{I\Gamma} \\ \mathbb{A}_i^{\Gamma I} & \mathbb{A}_i^{\Gamma\Gamma} \end{pmatrix}, \qquad \forall i \in \{1, \ldots, N\}, \tag{4.6}$$

with the same meaning of the superscripts I and Γ as in (4.2), here related to the subdomain Ω_i. To obtain the local Schur complement matrix $\mathbb{S}_i$, the interior degrees of freedom are eliminated locally

$$\mathbb{S}_i = \mathbb{A}_i^{\Gamma\Gamma} - \mathbb{A}_i^{\Gamma I} \left(\mathbb{A}_i^{II}\right)^{-1} \mathbb{A}_i^{I\Gamma}, \qquad \forall i \in \{1, \ldots, N\}. \tag{4.7}$$

Schur complement $\mathbb{S}$ is then formally assembled as

$$\mathbb{S} = \sum_{i=1}^{N} \mathbb{R}_i^T \mathbb{S}_i \mathbb{R}_i, \tag{4.8}$$

where $\mathbb{R}_i$ is the rectangular matrix consisting of ones and zeros, which represents the restriction of the vector of all interface degrees of freedom to those located at the i-th subdomain.

Neither the assembled Schur complement matrix $\mathbb{S}$ nor the local matrices $\mathbb{S}_i$ are constructed explicitly. Instead, only the assembly procedure (4.8) and the definition of local parts (4.7) are repeatedly used, as described in detail by Algorithm 8, where the action of the matrix $\mathbb{S}$ on a vector $\boldsymbol{x}$ is evaluated to obtain $\boldsymbol{y} = \mathbb{S}\boldsymbol{x}$.

In this work, PCG and BiCGStab are used for the solution of system (4.3a) in the symmetric and non-symmetric case, respectively. Therefore, the matrix-vector multiplication with the matrix $\mathbb{S}$ in the subassembled form described in Algorithm 8 is the only operation needed with this matrix for employing these methods as follows from the summary of these two methods in Algorithms 4 and 6.

Algorithm 8 Action of $\mathbb{S}$ on vector $\boldsymbol{x}$.

Set $\boldsymbol{y} = \boldsymbol{0}$ and repeat for each subdomain $i \in \{1, \ldots, N\}$:

1. Extract the part of $\boldsymbol{x}$ corresponding to the interface of the i-th subdomain to obtain $\boldsymbol{x}_i$ ($\boldsymbol{x}_i = \mathbb{R}_i \boldsymbol{x}$).
2. Multiply $\boldsymbol{c}_i = \mathbb{A}_i^{I\Gamma} \boldsymbol{x}_i$.
3. Solve $\mathbb{A}_i^{II} \boldsymbol{t}_i = \boldsymbol{c}_i$.
4. Multiply $\boldsymbol{z}_i = \mathbb{A}_i^{\Gamma I} \boldsymbol{t}_i$.
5. Multiply and subtract $\boldsymbol{y}_i = \mathbb{A}_i^{\Gamma\Gamma} \boldsymbol{x}_i - \boldsymbol{z}_i$.
6. Extend $\boldsymbol{y}_i$ by zeros ($\mathbb{R}_i^T \boldsymbol{y}_i$) and add the result to $\boldsymbol{y}$ ($\boldsymbol{y} = \sum_{i=1}^{N} \mathbb{R}_i^T \boldsymbol{y}_i$).

4.4 BDDC preconditioner

Without preconditioning, however, the convergence rate of the iterative solver for the Schur complement system (4.3a) deteriorates with the increasing number of subdomains [110]. Therefore, the applicability of unpreconditioned iterative methods for the solution of the Schur complement system, in this work BiCGStab and CG, is limited and not suitable for a larger number of subdomains.

The traditional preconditioners such as the diagonal Jacobi or incomplete factorizations, such as, ILU(0) employed for BiCGStab in the serial implementation in Section 4.1 cannot be used because the Schur complement matrix is not explicitly assembled. Consequently, preconditioners tailored to the solution of the Schur complement system in a subassembled form have to be used.

The Balancing Domain Decomposition by Constraints (BDDC) is one of such method. BDDC was introduced in [31] for problems with the symmetric positive definite matrix. Its extension to the advection-diffusion problems with non-symmetric matrices was presented in [114], with the implementation based on the change of basis without considering an explicit coarse problem. BDDC with an explicit coarse problem was applied to Euler equations in [118], where the concept of adjoint coarse basis functions was also introduced. In [54], the method was applied to non-symmetric saddle-point problems arising from a linearization of the stationary Navier–Stokes equations and [55] extended this approach to a multilevel method. The BDDC method was extended towards the flow in porous media in [105, 106, 119] and its application to hybrid finite element discretization includes [108, 112].

The key idea behind the BDDC method is choosing suitable coarse degrees of freedom. The choice of the coarse degrees of freedom depends on the selected spatial discretization and the dimension of the problem. For the MHFEM-based methods, the two types of coarse degrees of freedom are usually considered:

- Value at a given interface side referred to as a *corner*.
- Average over a given set of boundary sides: usually, all the sides common to the two neighboring subdomains are selected.

The solution is then found in the space of functions defined at the interface that are continuous in the chosen coarse degrees of freedom. Note that for different spatial discretizations, for example, Lagrangian finite elements with degrees of freedom located on element vertices, the classification

of the coarse degrees of freedom to *faces*, *edges*, and *vertices* [71] is possible and slightly different choices of coarse degrees of freedom are usually adopted.

In this work, the averages over sets of interface sides common to two subdomains are selected as coarse degrees of freedom for both 2D and 3D problems. From the MHFEM formulation in Chapter 3, it follows that each interface side belongs to exactly two subdomains. These averages are selected independently for each primary unknown Z_j of the original system 3.57, $j \in \{1, \ldots, n\}$, therefore, for each part of the interface common to two subdomains, there are n coarse degrees of freedom. The results presented later in Chapters 6, 7, and 8 indicate that this choice is sufficient for the problems considered in this work.

To complete the selection of coarse degrees of freedom we refer to [109] for the strategy to select corners for the MHFEM formulation and to [74, 107] for the adaptive selection of coarse degrees of freedom which are beyond the scope of this work.

The action of the BDDC preconditioner on a vector $\boldsymbol{x}$ to obtain the preconditioned vector $\boldsymbol{u}$ consists of two steps: a local and a global (coarse) correction.

4.4.1 Local correction

The local corrections do not require information from other subdomains and can therefore be performed simultaneously for all subdomains. They are the same for both symmetric and non-symmetric matrices.

First, the local part of the vector corresponding to the i-th subdomain is extracted from the original vector $\boldsymbol{x}$ and weighted as

$$\boldsymbol{x}_i = \mathbb{W}_i \mathbb{R}_i \boldsymbol{x}, \qquad \forall i \in \{1, \ldots, N\}, \tag{4.9}$$

where $\mathbb{W}_i$ is the weight matrix. Here, the diagonal matrix $\mathbb{W}_i = \frac{1}{2}\mathbb{I}$ is used. Note that for highly varying material properties between the subdomains, more advanced choices are needed to take into account these differences [108], however, as indicated by the results presented later in Chapters 6, 7, and 8, the choice $\mathbb{W}_i = \frac{1}{2}\mathbb{I}$ is sufficient for the problems considered in this work.

On each subdomain Ω_i, the following problem is solved with a vector $\boldsymbol{x}_i$ (4.9) as a part of the right-hand side

$$\begin{pmatrix} \mathbb{A}_i^{II} & \mathbb{A}_i^{I\Gamma} & \Theta \\ \mathbb{A}_i^{\Gamma I} & \mathbb{A}_i^{\Gamma\Gamma} & \mathbb{C}_i^T \\ \Theta & \mathbb{C}_i & \Theta \end{pmatrix} \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{u}_i \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{x}_i \\ \boldsymbol{0} \end{pmatrix}, \qquad \forall i \in \{1, \ldots, N\}, \tag{4.10}$$

where Θ denotes a zero matrix of appropriate dimensions and the matrix $\mathbb{C}_i$ defines the coarse degrees of freedom. The number of rows of $\mathbb{C}_i$ is the same as the number of coarse degrees of freedom corresponding to Ω_i. Corners are represented by rows with a single one in the column corresponding to the selected degree of freedom in $\mathbb{A}_i$. Averages over parts of the subdomain boundary for a given primary unknown are represented by a row of ones on the positions of the averaged sides and unknowns in $\mathbb{A}_i$. Vector $\boldsymbol{u}_i$ obtained as a part of the solution of (4.10) is the local part of the preconditioned vector. This local part is then combined with the global (coarse) correction described in the following section.

4.4.2 Coarse correction

The coarse correction requires information from all subdomains, moreover, the coarse correction for non-symmetric matrices requires additional steps in comparison with the symmetric case. To

compute the global part of the preconditioned vector, the coarse basis functions $\boldsymbol{\Phi}_i$ are needed as the solution of the following system with multiple right-hand sides

$$\begin{pmatrix} \mathbb{A}_i^{II} & \mathbb{A}_i^{I\Gamma} & \boldsymbol{\Theta} \\ \mathbb{A}_i^{\Gamma I} & \mathbb{A}_i^{\Gamma\Gamma} & \mathbb{C}_i^T \\ \boldsymbol{\Theta} & \mathbb{C}_i & \boldsymbol{\Theta} \end{pmatrix} \begin{pmatrix} \mathbb{Y}_i \\ \boldsymbol{\Phi}_i \\ \boldsymbol{\Lambda}_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Theta} \\ \boldsymbol{\Theta} \\ \mathbb{I} \end{pmatrix}, \qquad \forall i \in \{1, \ldots, N\}, \tag{4.11}$$

which is solved on each subdomain $\boldsymbol{\Omega}_i$.

The coarse basis functions are sufficient for symmetric matrix $\mathbb{A}$. However, for the case of the non-symmetric matrices, which are encountered in most of the problems considered in this work, also the adjoint coarse basis functions $\boldsymbol{\Phi}_i^*$ are needed [54, 114, 118]. The adjoint coarse basis functions are found as the solution of the following system with multiple right-hand sides

$$\begin{pmatrix} \mathbb{A}_i^{II^T} & \mathbb{A}_i^{\Gamma I^T} & \boldsymbol{\Theta} \\ \mathbb{A}_i^{I\Gamma^T} & \mathbb{A}_i^{\Gamma\Gamma^T} & \mathbb{C}_i^T \\ \boldsymbol{\Theta} & \mathbb{C}_i & \boldsymbol{\Theta} \end{pmatrix} \begin{pmatrix} \mathbb{Y}_i^* \\ \boldsymbol{\Phi}_i^* \\ \boldsymbol{\Lambda}_i^* \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Theta} \\ \boldsymbol{\Theta} \\ \mathbb{I} \end{pmatrix}, \qquad \forall i \in \{1, \ldots, N\}, \tag{4.12}$$

which is again solved on each subdomain $\boldsymbol{\Omega}_i$.

Note that for symmetric matrix $\mathbb{A}$, satisfying $\mathbb{A} = \mathbb{A}^T$ and as a result also $\mathbb{A}_i = \mathbb{A}_i^T$ for each subdomain $\boldsymbol{\Omega}_i$, equations (4.11) and (4.12) are the same and as a result, the basis functions $\boldsymbol{\Phi}_i$ and adjoint basis functions $\boldsymbol{\Phi}_i^*$ are identical. To unify the description of the BDDC method for symmetric and non-symmetric matrices in the following text, both the coarse and adjoint coarse basis functions are used even if they are identical for symmetric matrices $\mathbb{A}$.

As a side product of computing coarse basis functions and adjoint coarse basis functions in (4.11) and (4.12) we also obtain the local parts of the coarse problem matrix $\mathbb{S}_{C_i}$, defined by

$$\mathbb{S}_{C_i} = \boldsymbol{\Phi}_i^{*T} \mathbb{S}_i \boldsymbol{\Phi}_i, \qquad \forall i \in \{1, \ldots, N\}. \tag{4.13}$$

To find the expression of $\mathbb{S}_{C_i}$ as a part of the solution of (4.11), we start with the linear systems for the basis functions (4.11) and (4.12) that are formally equivalent to the following systems

$$\begin{pmatrix} \mathbb{S}_i & \mathbb{C}_i^T \\ \mathbb{C}_i & \boldsymbol{\Theta} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Phi}_i \\ \boldsymbol{\Lambda}_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Theta} \\ \mathbb{I} \end{pmatrix}, \tag{4.14a}$$

respectively

$$\begin{pmatrix} \mathbb{S}_i^T & \mathbb{C}_i^T \\ \mathbb{C}_i & \boldsymbol{\Theta} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Phi}_i^* \\ \boldsymbol{\Lambda}_i^* \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Theta} \\ \mathbb{I} \end{pmatrix}. \tag{4.14b}$$

From the multiplication of the block partitioned matrices in (4.14) it follows

$$\mathbb{C}_i \boldsymbol{\Phi}_i = \mathbb{C}_i \boldsymbol{\Phi}_i^* = \mathbb{I}. \tag{4.15}$$

To obtain the expression of $\mathbb{S}_{C_i}$, both sides of (4.14a) are multiplied by $\begin{pmatrix} \boldsymbol{\Phi}_i^{*T} & \boldsymbol{\Lambda}_i^{*T} \end{pmatrix}$ from the left, while (4.14b) is multiplied by $\begin{pmatrix} \boldsymbol{\Phi}_i^T & \boldsymbol{\Lambda}_i^T \end{pmatrix}$ from the left. The identity (4.15) is then employed and as a result, the following expressions are obtained

$$\boldsymbol{\Phi}_i^{*T} \mathbb{S}_i \boldsymbol{\Phi}_i = -\boldsymbol{\Lambda}_i, \tag{4.16a}$$

$$\boldsymbol{\Phi}_i^T \mathbb{S}_i^T \boldsymbol{\Phi}_i^* = -\boldsymbol{\Lambda}_i^*. \tag{4.16b}$$

Transposing (4.16a), we obtain the following expression

$$\boldsymbol{\Lambda}_i = -\boldsymbol{\Phi}_i^{*T} \mathbb{S}_i \boldsymbol{\Phi}_i = \boldsymbol{\Lambda}_i^{*T}, \tag{4.17}$$

from which, the local part of the coarse problem matrix $\mathbb{S}_{C_i}$ is obtained as a part of the solution of the problem for the coarse basis and adjoint coarse basis functions as,

$$\mathbb{S}_{C_i} = \boldsymbol{\Phi}_i^{*T} \mathbb{S}_i \boldsymbol{\Phi}_i = -\boldsymbol{\Lambda}_i \qquad \forall i \in \{1, \ldots, N\}. \tag{4.18}$$

Then, the matrix of the coarse problem $\mathbb{S}_C$ is assembled from these local parts as

$$\mathbb{S}_C = \sum_{i=i}^{N} \mathbb{R}_{C_i}^T \mathbb{S}_{C_i} \mathbb{R}_{C_i}, \tag{4.19}$$

where $\mathbb{R}_{C_i}$ is the rectangular matrix consisting of ones and zeros, which represents the restriction of the vector of coarse degrees of freedom to those located at the i-th subdomain.

The right-hand side vector of the coarse problem is then constructed

$$\boldsymbol{x}_C = \sum_{i=1}^{N} {\mathbb{R}_{C_i}}^T \boldsymbol{\Phi}_i^{*T} \boldsymbol{x}_i, \tag{4.20}$$

and the coarse problem

$$\mathbb{S}_C \boldsymbol{u}_C = \boldsymbol{x}_C, \tag{4.21}$$

is solved. From the solution of (4.21) the part corresponding to the subdomain $\boldsymbol{\Omega}_i$ is obtained

$$\boldsymbol{u}_{C_i} = \boldsymbol{\Phi}_i \mathbb{R}_{C_i} \boldsymbol{u}_C, \qquad \forall i \in \{1, \ldots, N\}. \tag{4.22}$$

Finally, the global and local parts of the vectors are weighted and combined to the preconditioned vector $\boldsymbol{u}$ as

$$\boldsymbol{u} = \sum_{i=1}^{N} \mathbb{R}_i^T \mathbb{W}_i (\boldsymbol{u}_i + \boldsymbol{u}_{C_i}). \tag{4.23}$$

4.4.3 Algorithm summary

To summarize the description of the preconditioner, the steps needed to obtain its action on a vector $\boldsymbol{x}$ are in the compact form presented in Algorithm 9.

4.5 Parallel implementation of BDDC

In this section, the parallel implementation of the iterative solver for the Schur complement system in a subassembled form with the BDDC preconditioner, i.e., the implementation of the steps 5 and 6 of Algorithm 2, is presented.

In this work, the Schur complement system (4.3a) is solved by the Krylov subspaces-based BiCGStab [115] or PCG [27] iterative solvers in the case of non-symmetric or symmetric matrices, respectively. Both methods are similar in terms of operations needed and involve an initialization step (Section 4.5.1), matrix-vector multiplications with matrix $\mathbb{S}$ (Section 4.5.2), scalar products (Section 4.5.3), and applications of the BDDC preconditioner (Section 4.5.4). The differences in the methods such as the additional set of vectors required by BiCGStab do not affect the implementation of these main operations, therefore, the implementation of both methods is described together.

Algorithm 9 Application of the BDDC preconditioner on the vector $\boldsymbol{x}$.

Start with the vector $\boldsymbol{x}$ and perform the following steps on each subdomain Ω_i, $i \in \{1, \ldots, N\}$:

1. Extract and weight the local part of the vector $\boldsymbol{x}_i$ using (4.9).
2. Compute the local correction $\boldsymbol{u}_i$ as the solution of (4.10).
3. Compute the coarse basis functions $\boldsymbol{\Phi}_i$ and obtain a local part of the coarse matrix S_{C_i} using (4.11).
4. In the case of non-symmetric matrix, compute adjoint coarse basis functions $\boldsymbol{\Phi}_i^*$ using (4.12)

Using the data from all subdomains, perform the following steps:

1. Assemble the coarse problem matrix $\mathbb{S}_C$ using (4.19) and the right-hand side $\boldsymbol{x}_C$ using (4.20).
2. Compute the coarse correction $\boldsymbol{u}_C$ as the solution of (4.21).

On each subdomain Ω_i, $i \in \{1, \ldots, N\}$, perform the following steps:

1. Obtain $\boldsymbol{u}_{C_i}$ using (4.22).
2. Combine local $\boldsymbol{u}_i$ and coarse $\boldsymbol{u}_{C_i}$ corrections to obtain $\boldsymbol{u}$ using (4.23).

4.5.1 Initialization

For the iterative solution, the assembled matrix $\mathbb{A}$ is not needed, instead, the method requires the assembly of the local parts $\mathbb{A}_i$ as described in Section 4.3. Therefore, in step 5 of Algorithm 2 each process assembles the corresponding local matrix $\mathbb{A}_i$ using the same approach that was described in Section 3.6 for the complete matrix $\mathbb{A}$ restricted to the its subdomain only. The construction of $\mathbb{A}_i$ requires only values that are already available to the process, therefore, $\mathbb{A}_i$ can be assembled in parallel and no communication is needed.

During each iteration of the solver (PCG or BiCGStab), the solutions of several linear systems with the same matrices are needed on each subdomain. To perform these solutions efficiently, the factorizations of the matrices are pre-computed before the actual iterations start. The solutions of these systems during the iterations are then performed by backward and forward substitutions for various right-hand sides. For symmetric matrices, the Cholesky factorizations are computed using the numerical library `CHOLMOD` [24]. For non-symmetric matrices, the LU factorizations with partial pivoting are computed using the numerical library `UMFPACK` [29].

The factorization of $\mathbb{A}_i^{II}$ is required to compute the action of matrix the $\mathbb{S}_i$ during each iteration as described in Algorithm 8. The factorization of matrices in (4.10) is required to construct the coarse basis functions for the coarse corrections and to compute the local corrections in the application of the preconditioner.

With the matrices in (4.10) factorized, coarse basis functions $\boldsymbol{\Phi}_i$ are computed by solving (4.11) with multiple right-hand sides. The adjoint coarse basis functions $\boldsymbol{\Phi}_i^*$ needed for the non-symmetric matrices are computed using the same factorization since `UMFPACK` allows to efficiently solve the system with the transposed matrix once the LU factorization of the original matrix is available [29]. As a side product of the computations of these functions, each process also obtains a local part of the coarse matrix $\mathbb{S}_{C_i}$ defined by (4.17) as discussed in Section 4.4.2.

Matrix $\mathbb{S}_C$ is then assembled from these local parts. In contrast to the Schur complement matrix $\mathbb{S}$, this is done explicitly. Here, the coarse matrix is assembled by each process and the

resulting linear systems with matrix $\mathbb{S}_C$ are solved redundantly by each process. This approach is advantageous for a small to moderate number of processes where matrix $\mathbb{S}_C$ is small. Solving the coarse problem redundantly, however, becomes the bottleneck for the scalability of the method for a large number of processes, since the matrix becomes too large to be efficiently stored and factorized in serial. This limitation was overcome by the introduction of multilevel BDDC methods in [5, 73, 107, 113] where the coarse problem is not solved exactly and only an approximation of the solution obtained by BDDC is used, nevertheless, this approach is beyond the scope of this work.

The assembled matrix $\mathbb{S}_C$ is sparse and its structure is determined by the choice of coarse degrees of freedom, however, for a small number of processes (at most 32 for the problems considered in this work), the matrix is too small to benefit from the sparse format and it is convenient to store it in a dense format. The usage of the sparsity pattern becomes advantageous for a moderate number of processes (over 32). In this case, matrix $\mathbb{S}_C$ is stored in the compressed sparse column (CSC) format required by `CHOLMOD` or `UMFPACK`. This reduces the memory requirements to store the matrix and also the solution of the sparse coarse problem of this size is faster than the solution of the same problem stored as a dense matrix. For the assembly of $\mathbb{S}_C$, the local parts $\mathbb{S}_{C_i}$ are stored in local copies of $\mathbb{S}_C$ in the positions determined by the coarse degrees of freedom by each process. The assembled $\mathbb{S}_C$ is then obtained using the `MPI` operation `Allreduce` performed on these local copies of $\mathbb{S}_C$. In the case of a matrix stored in the CSC format, the operation is performed with the array of matrix entries only as the structure of the matrix is known from the decomposition into subdomains and the choice of coarse degrees of freedom at the start of the computation. Each process obtains its copy of $\mathbb{S}_C$ and as the final step of the initialization, each process constructs the decomposition (LU or Cholesky) of the matrix $\mathbb{S}_C$.

4.5.2 Matrix-vector product

For the computation of the matrix-vector product with matrix $\mathbb{S}$, we follow the steps of Algorithm 8. The step 1 is actually not needed, because, at the beginning of the matrix-vector product computation, each process knows only the part of the vector corresponding to the unknowns at the subdomain interface mapped to this process. Steps 2–5 use only the data known to the process and are done in parallel by each process independently, and moreover, the factorization of matrices $\mathbb{A}_i^{II}$ is known from the initialization step described in Section 4.5.1. The communication is needed in step 6 only, where each subdomain needs the values from all the neighboring subdomains to assemble the final vector. This communication is realized by the combination of `MPI` operations `Isend` and `Ireceive`, where each process sends and receives the values of the corresponding part of the interface from the neighboring subdomain. The exchanged values are then added to their own part of the vector to obtain the final result. In fact, steps 1 and 6 of Algorithm 8 are naturally merged together.

4.5.3 Scalar product

For the computation of the scalar product, first, each process computes the scalar product of the parts of the vector mapped to this process. Then, these local (partial) products are added together and the sum is distributed back to all processes by the `MPI` operation `Allreduce`. In this operation, every entry of the vector is processed twice because each part of the interface in the MHFEM discretization employed in this work belongs to exactly two neighboring subdomains.

4.5.4 BDDC preconditioner

The application of the preconditioner consists of two steps: local and coarse correction. We start with the coarse correction. The factorization of $\mathbb{S}_C$ is already known by each process from the initialization step and the right-hand side of (4.21) is assembled from the local parts computed by each process using (4.20). The `MPI` operation `Iallreduce` is used on the local parts extended by zeros to the size of the full vector of the coarse right-hand side.

The communication needed to obtain the coarse vector $\boldsymbol{x}_C$ is overlapped with the computation of the local corrections $\boldsymbol{u}_i$ obtained as a solution of (4.10) (see e.g. [5]).

When the vector $\boldsymbol{x}_C$ is assembled, each process then solves the coarse problem (4.21) and computes the part of the solution $\boldsymbol{u}_C$ corresponding to its subdomain by (4.22). Based on the number of subdomains, the solution of the coarse problem (4.21) is carried out as a solution of a dense or sparse linear system depending on the pre-computed factorization in the initialization step.

The final combination of the local and coarse corrections in (4.23) is similar to the computation of the matrix-vector product in Section 4.5.2: each process exchanges values with neighboring subdomains by the `MPI` operations `Isend` and `Ireceive` and adds the received values to its own copy of $\boldsymbol{u}$ to obtain the final preconditioned vector $\boldsymbol{u}$.

4.6 Chapter summary

We presented both serial and parallel implementations of the numerical scheme proposed in Chapter 3. The serial implementation can be used together with a direct or iterative solver for the resulting linear system that needs to be solved at each time step. The serial implementation was published in [43] where also different methods for solving the linear system were tested. The parallel implementation based on iterative substructuring and iterative solution of the Schur complement system with the BDDC preconditioner was published in [104].

Verification of numerical scheme 5

In this chapter, the numerical scheme presented in Chapter 3 is verified using benchmark problems of the two-phase flow in porous media for which the solution is known. We are interested in the accuracy and convergence properties of the numerical scheme only. Therefore, the serial implementation of the numerical scheme is employed. The computational times and the performance of the parallel implementation described in Chapter 4 are investigated separately in Chapter 6.

Two benchmark problems: one in homogeneous and one in heterogeneous porous media are considered in Sections 5.1 and 5.2, respectively.

The benchmark problem in a homogeneous porous medium is the McWhorter-Sunada problem in 2D and its extension into 3D. The 2D problem and its solution were originally proposed in [76] and the improved method to find the solution was presented in [45]. The generalization into an arbitrary spatial dimension d was done in [42]. Both the original problem and its 3D generalization represent a simplified scenario of a point injection of fluid into a (partially) saturated porous media without gravity, under the assumption of incompressible phases, and radially symmetric domains.

The benchmark problem in a heterogeneous porous medium was proposed in [62] to demonstrate the barrier effect for the two-phase systems at the interface between coarse and fine sands. The physically correct numerical solution of the flow through the material interface is important for the application of the numerical scheme to the scenarios arising from the laboratory experiments in heterogeneous porous media described later in Chapters 7 and 8.

5.1 Homogeneous benchmark problem

The generalized McWhorter-Sunada problem is a problem of two-phase flow in a porous medium for which the semi-analytical solution can be found under the assumption of incompressible phases and neglected gravity in a radially symmetric domain [42, 45, 76]

The equations describing the two-phase flow in porous media are summarized in Section 2.3.9 in the form of a system of two partial differential equations ($n = 2$ in (3.1)) and an algebraic relation between the phase saturation and the capillary pressure. The representation of this problem in the general coefficient formulation (3.1) depends on the choice of the primary unknowns. In this chapter, the capillary pressure and the non-wetting phase pressure are selected as the primary unknowns, i.e., $Z_1 = p_c$ and $Z_2 = p_n$. All the remaining quantities are computed from these two primary unknowns using the relations presented in Section 2.3. For such choice of the primary variables, the McWhorter-Sunada problem is represented by the following non-zero

coefficients in (3.1),

$$\mathcal{N} = \begin{pmatrix} -\Phi\rho_w \frac{dS_w}{dp_c} & 0 \\ -\Phi\rho_n \frac{dS_w}{dp_c} & 0 \end{pmatrix}, \qquad \mathcal{M} = \begin{pmatrix} \rho_w \frac{\lambda_w}{\lambda_t} \\ \rho_n \frac{\lambda_n}{\lambda_t} \end{pmatrix}, \qquad \mathcal{D} = \begin{pmatrix} \lambda_t \mathbb{K} & -\lambda_t \mathbb{K} \\ 0 & \lambda_t \mathbb{K} \end{pmatrix}. \tag{5.1}$$

Note that a different choice of the primary variables are possible, for example, phase pressures, i.e., $Z_1 = p_w$, $Z_2 = p_n$ were used in [43]. However, the phase saturations S_α are in general not feasible as the primary unknowns as the saturation values may become discontinuous over the material interfaces in the case of heterogeneous media as discussed in Section 2.3.8.

The semi-analytical solutions for the original and generalized McWhorter-Sunada problems can be found for the following problem formulation. We assume a radial symmetry of the solution domain in the entire $\mathfrak{R}^2$ and $\mathfrak{R}^3$. The initial saturation of the non-wetting phase is denoted by S_i and in this work, $S_i = 0.05$ is prescribed. Additionally, we assume a point source $Q \left[\mathrm{m}^d\,\mathrm{s}^{-1}\right]$ of the non-wetting phase located at the origin in the form

$$Q(t) = A_d t^{\frac{d-2}{2}}, \tag{5.2}$$

where d is the dimension of the problem ($d \in \{2, 3\}$) and $A_d \left[\mathrm{m}^3 s^{-\frac{d}{2}}\right]$ is an arbitrary positive constant. Note that for $d = 2$, the source term is constant in time. Here, the values $A_2 = 10^{-5}\,\mathrm{m}^2\mathrm{s}^{-1}$ and $A_3 = 10^{-7}\,\mathrm{m}^3\mathrm{s}^{-\frac{3}{2}}$ are selected for the 2D and 3D case, respectively.

For the numerical solution of this problem, we take advantage of the radial symmetry of the domain and consider only the first quadrant and octant in 2D and 3D, respectively. Additionally, we consider a domain of finite length and choose the final time of the computation such that the head of the solution does not reach the boundary representing infinity. In 2D and 3D, the computational domain is a square with a 1 meter long side and a cube with a 1 meter long edge, respectively. One vertex of the domain is located at the origin, where the source is prescribed and the domain boundaries are aligned with the coordinate axis. The computational domains and the boundary conditions representing the radial symmetry and infinity are shown in Figures 5.2 and 5.3 for the 2D and 3D case, respectively. The final time of the computation $t_{fin} = 20000\,\mathrm{s}$ is selected for both 2D and 3D cases. The numerical solutions at the final time are shown in Figures 5.2 and 5.3 for the 2D and 3D cases, respectively

Additionally, the point source at the origin needs to be approximated in the numerical scheme. The numerical method employed here cannot deal with prescribing a flux in one point of the mesh, therefore, the point source is approximated using the boundary condition by prescribing the flux through all the sides (edges, faces) adjacent to the source point. The selection of the sides on which the boundary condition is prescribed is illustrated in Figure 5.1 for the triangular mesh in 2D and tetrahedral mesh in 3D. The values of the Neumann boundary condition are computed so that the total volume injected through the selected boundary sides is the same as the amount given by (5.2).

The porous medium for all the computations in this section is Sand A with the properties given in Table 5.1. The computations are carried out for both Brooks-Corey and van Genuchten models for the capillary pressure coupled with the corresponding Burdine and Mualem models for the relative permeability. The wetting phase is water and the non-wetting phase is NAPL. The fluid properties are given in Table 5.1.

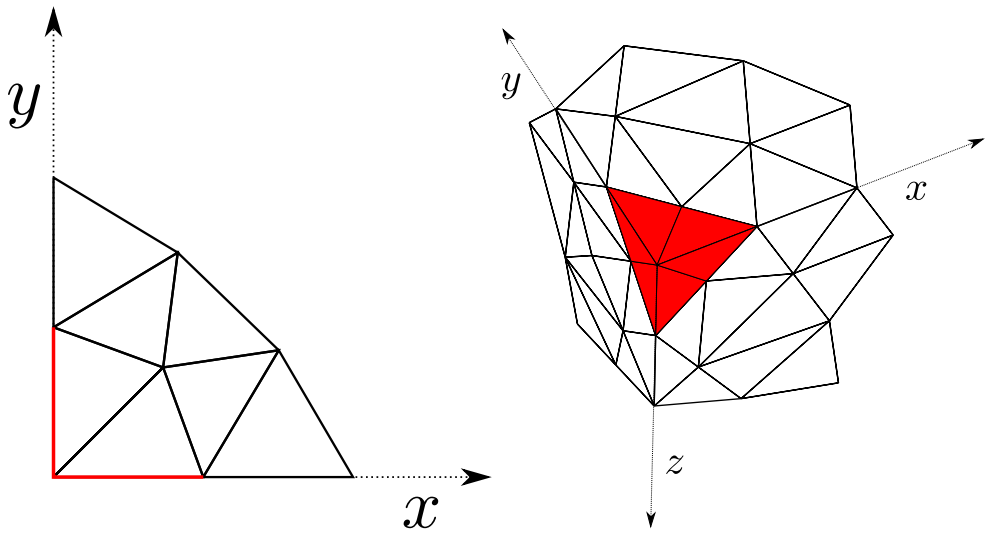
5.1.1 Computational study setup

The convergence study is carried out both in 2D and 3D on both structured and unstructured meshes. On all meshes, both the EI and ML variants of the scheme which were introduced in Section 3.9 are compared to demonstrate the differences in the accuracy of these two approaches.

Sand	ϕ [–]	K [m^2]	S_{rw} [–]	S_{rn} [–]	p_d [Pa]	λ^{BC} [–]	α^{vG} $[\mathrm{Pa}]^{-1}$	n^{vG} [–]
Sand A	0.343	$5.168 \cdot 10^{-12}$	0.04	0	4605.8	2.857	$1.71 \cdot 10^{-4}$	6.64

Fluid	ρ $[\mathrm{kg\,m}^{-3}]$	μ $[\mathrm{Pa\,s}^{-1}]$
Water	1000	$1 \cdot 10^{-4}$
NAPL	1400	$1 \cdot 10^{-4}$

Table 5.1: Sand and fluid properties for McWhorter-Sunada problem in Section 5.1. Adapted from [43].

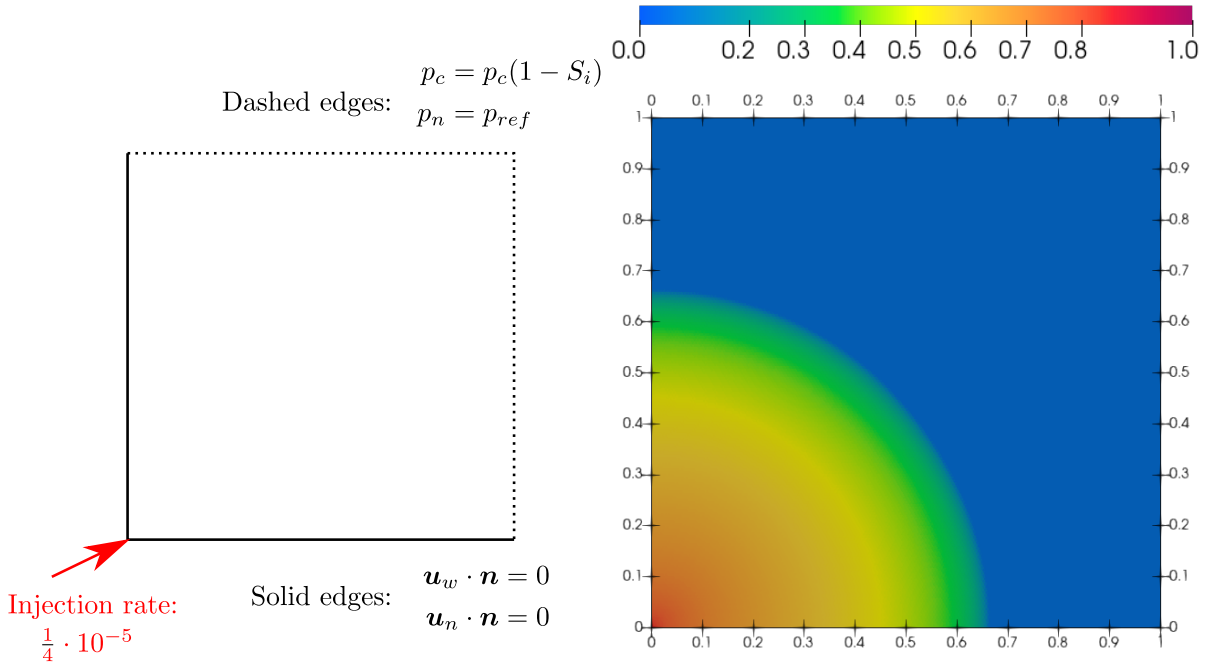


(a) 2D. (b) 3D.

Figure 5.1: Approximation of the point source at the origin. The sides on which the Neumann boundary condition is prescribed are shown in red.

For each case listed above, the study is carried out on five meshes numbered from 1 to 5. For the purposes of the convergence study, the meshes are characterized by the mesh size h, defined as the largest ball diameter circumscribed around mesh elements. The meshes from 1 to 5 are generated to approximately halve the value of h between two consecutive meshes and their properties are given in Table 5.2 where, in addition to the mesh size h, the number of elements and the number of sides for each mesh are presented. The degrees of freedom in the MHFEM-based numerical scheme employed here are located on the mesh sides, therefore the total number of degrees of freedom (# DOF) for the given mesh is twice the number of sides. All the unstructured meshes were generated by `COMSOL Multiphysics version 3.5a`.

With a single exception, the convergence study was carried out using the serial implementation of the numerical scheme presented in Section 4.1 and the computations were performed on a PC equipped with Intel Core i7-5820K, 3.6 GHz processor, and 32 GB RAM. The BiCGStab with the ILU(0) preconditioner was used as the linear solver for the resulting system (3.57) in step 6 of Algorithm 2. The iterations are terminated when the relative norm of the residual is lower



(a) Boundary conditions. (b) Numerical solution.

Figure 5.2: Boundary conditions and numerical solution (values of saturation S_n) for the 2D McWhorter-Sunada problem. Dashed lines represent infinity, solid lines represent the radial symmetry of the problem.

than a given tolerance ϵ

$$\frac{\left\| \mathbb{A}\boldsymbol{Z}_q^{k+1} - \boldsymbol{b} \right\|}{\|\boldsymbol{b}\|} < \epsilon, \tag{5.3}$$

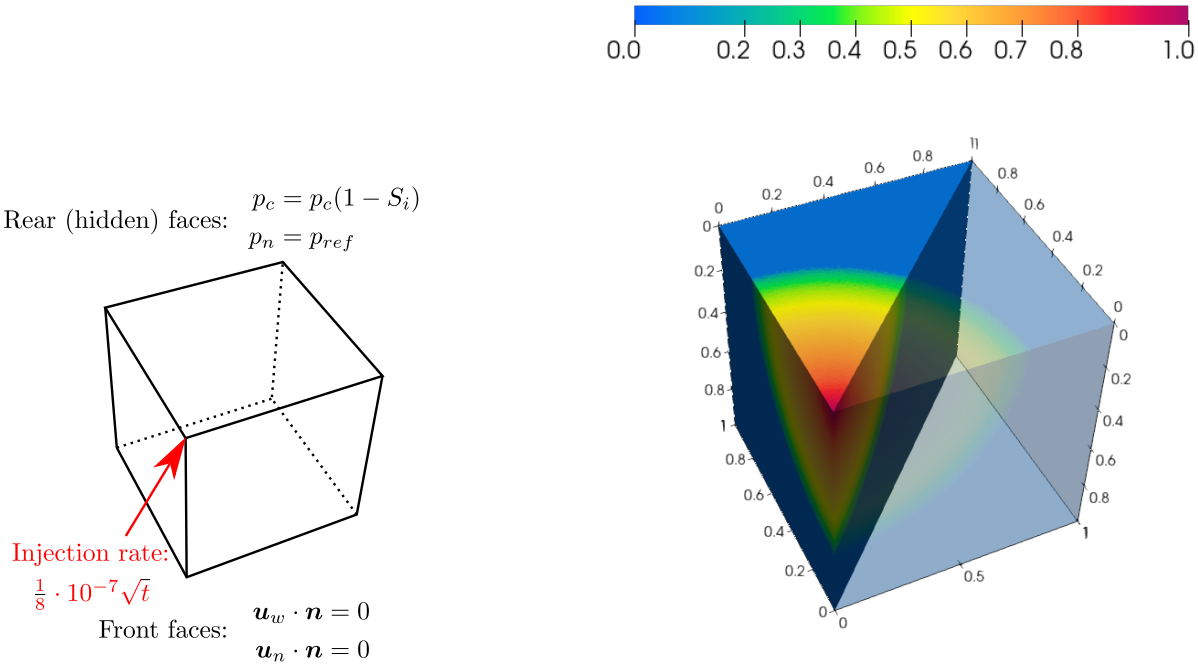
where $\boldsymbol{Z}_q^{k+1}$ is the solution obtained in q-th iteration. The value of $\epsilon = 10^{-15}$ is selected for the serial implementation in this section.

The only exception was the 3D cuboidal mesh number 5, where due to the memory requirements, the parallel implementation presented in Chapter 4 had to be used and the computations were carried out on the `Helios` cluster at the Czech Technical University in Prague, Czech Republic. The computational nodes of `Helios` were equipped with two 16-core AMD EPYC 7281, 2.1GHz processors and 128 GB RAM. The iterations are terminated when the relative norm of the residual of the Schur complement system (4.3a) is lower than a given tolerance ϵ

$$\frac{\left\| \mathbb{S}\boldsymbol{Z}_q^{\Gamma,k+1} - \hat{\boldsymbol{b}} \right\|}{\left\| \hat{\boldsymbol{b}} \right\|} < \epsilon, \tag{5.4}$$

where $\boldsymbol{Z}_q^{\Gamma,k+1}$ is the solution for the degrees of freedom located on the interface obtained in q-th iteration. The value of $\epsilon = 10^{-9}$ is selected in this case.

The computations were carried out with a fixed time step, the values of Δt are listed in Table 5.2 individually for each of the capillary pressure models considered. These time steps were selected to minimize the error of the numerical solution defined in Section 5.1.2 on the given mesh for the selected capillary pressure model [72].



(a) Boundary conditions. (b) Numerical solution.

Figure 5.3: Boundary conditions and numerical solution (values of saturation S_n) for the 3D extension of the McWhorter-Sunada problem. Rear faces represent infinity, front faces represent the radial symmetry of the problem.

5.1.2 Tools for numerical analysis

For the analysis of the numerical scheme, the numerical solution on a given mesh at the final time of the computation $t_{fin} = 20000$ s is compared to the semi-analytical solution and the error is computed. Here, we present the tools needed for this comparison. The quantity selected for comparison is the non-wetting phase saturation S_n, which is the primary unknown of the semi-analytical solution [42].

Error of the numerical solution

The semi-analytical solution of the McWhorter-Sunada problem is found under the assumption of radial symmetry. To compute the error, this semi-analytical solution denoted by g is projected onto the numerical mesh using linear interpolation. The value of the semi-analytical solution at a given point of the numerical mesh, which is needed for the comparison with the numerical solution, is determined by its distance from the origin. The numerical solution on the mesh with size h, denoted by g_h, and obtained by the scheme proposed in Chapter 3, is an element-wise constant function and no additional interpolation is used.

For the purposes of the comparison, the semi-analytical solution is available in the form of uniformly distributed 10 000 samples at given distances from the origin. The resolution is selected much higher than the spatial resolution of the finest mesh used in this study limiting the impact of the interpolation errors arising from the projection of the semi-analytical solution to the mesh.

The error of the numerical solution on a mesh with the element size h is denoted by $||E_{h,g}||_p$ and is defined as the p-norm of the difference between the semi-analytical solution g and the

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt_{BC} [s]	Δt_{vG} [s]
$2D_1^{\triangle}$	$6.71 \cdot 10^{-2}$	242	383	766	454.55	36.36
$2D_2^{\triangle}$	$3.49 \cdot 10^{-2}$	944	1 456	2 912	145.99	15.49
$2D_3^{\triangle}$	$1.64 \cdot 10^{-2}$	3 714	5 651	11 302	44.64	4.64
$2D_4^{\triangle}$	$8.73 \cdot 10^{-3}$	14 788	22 342	44 684	13.44	1.35
$2D_5^{\triangle}$	$4.23 \cdot 10^{-3}$	59 336	89 324	178 648	5.00	0.50
$2D_1^{\square}$	$9.43 \cdot 10^{-2}$	225	480	960	253.16	317.00
$2D_2^{\square}$	$4.71 \cdot 10^{-2}$	900	1 860	3 720	90.50	80.00
$2D_3^{\square}$	$2.36 \cdot 10^{-2}$	3 600	7 320	14 640	31.90	19.96
$2D_4^{\square}$	$1.18 \cdot 10^{-2}$	14 400	58 080	116 160	10.62	5.02
$2D_5^{\square}$	$5.89 \cdot 10^{-3}$	57 600	115 680	231 1360	3.57	1.26
$3D_1^{\triangle}$	$2.13 \cdot 10^{-1}$	1 312	2 937	5 874	833.33	152.67
$3D_2^{\triangle}$	$1.27 \cdot 10^{-1}$	3 697	7 773	15 546	571.43	125.79
$3D_3^{\triangle}$	$6.29 \cdot 10^{-2}$	29 673	60 839	121 678	232.56	60.24
$3D_4^{\triangle}$	$3.48 \cdot 10^{-2}$	240 372	486 875	973 750	101.01	43.86
$3D_5^{\triangle}$	$1.84 \cdot 10^{-2}$	1 939 413	3 903 609	7 807 218	25.00	20.00
$3D_1^{\square}$	$1.15 \cdot 10^{-1}$	3 375	10 800	21 600	333.33	235.29
$3D_2^{\square}$	$5.77 \cdot 10^{-2}$	27 000	83 700	167 400	131.58	58.82
$3D_3^{\square}$	$2.89 \cdot 10^{-2}$	216 000	658 800	1 317 600	53.48	14.71
$3D_4^{\square}$	$1.44 \cdot 10^{-2}$	1 728 000	5 227 200	10 454 400	22.10	3.68
$3D_5^{\square}$	$7.22 \cdot 10^{-3}$	13 824 000	41 644 800	83 289 600	10.50	1.5

Table 5.2: Mesh properties for the benchmarks described in Section 5.1 and the time steps for the given mesh and model of the capillary pressure. Subscripts BC and vG stand for the Brooks-Corey and van Genuchten capillary pressure models, respectively.

numerical solution g_h in the computational domain Ω

$$\|E_{h,g}\|_p = \left(\int_\Omega |g - g_h|^p \right)^{\frac{1}{p}}, \tag{5.5}$$

where the values of $p \in \{1, 2\}$ are considered. For the computation of the integral over domain Ω in (5.5), the integral is broken into integrals over the individual mesh elements

$$\left(\int_\Omega |g - g_h|^p \right)^{\frac{1}{p}} = \left(\sum_{K \in \mathcal{K}_h} \int_K |g - g_h|^p \right)^{\frac{1}{p}}, \tag{5.6}$$

and the integrals on each element $K \in \mathcal{K}_h$ are evaluated using the 7-point Gauss-Lobatto [1] quadrature rule.

Experimental order of convergence

At a given time t, we assume that the error of the numerical solution can be expressed in the form

$$\|E_{h,g}\|_p = C_{time,g,p}\tau + C_{space,g,p}h^{OC_{g,p}}, \tag{5.7}$$

where h denotes the mesh element size and τ denotes the time step. $C_{time,g,p}$ and $C_{space,g,p}$ are positive constants, $OC_{g,p}$ is the order of convergence of the numerical scheme in the p-norm. We assume that a time step τ is a function of mesh resolution h: here, it is selected to minimize the value of the error of the numerical solution $\|E_{h,g}\|_p$ given by (5.5) and the values for individual meshes are listed in Table 5.2. Therefore, the order of convergence is investigated as a function of mesh element size h.

The theoretical value of the order of convergence $OC_{g,p}$ of the numerical scheme introduced in (5.7) is approximated by the experimental order of convergence denoted by $EOC_{g,p}$ [37]. The experimental order of convergence $EOC_{g,p}$ computed using two numerical solutions g_{h1} and g_{h2} on two meshes with element sizes h_1 and h_2 in the p-norm is defined by

$$EOCg,p = \frac{\ln \|E_{h_1,g}\|_p - \ln \|E_{h_2,g}\|_p}{\ln h_1 - \ln h_2}, \tag{5.8}$$

which follows from (5.7) after eliminating the constants $C_{space,g,p}$ and $C_{time,g,p}$.

5.1.3 Results

The results for the EI variant of the method are shown in Tables 5.3 and 5.5 for 2D and 3D meshes, respectively. The results for the ML variant of the method are shown in Tables 5.3 and 5.5 for 2D and 3D meshes, respectively.

The results for the EI variant of the method presented here indicate that the numerical scheme is convergent with the first order of accuracy for all the considered mesh types and there is no significant difference in the convergence rate between the 2D and 3D cases. These convergence results justify the modification of the mass balance equation (3.52) where the mobility term is eliminated and the mass balance is formulated in terms of velocities only (3.54) as was described in Section 3.6.

The comparison of the two methods to compute the local coefficient matrices $\mathbb{H}_{i,j,K}$ shows that employing the ML variant of the method slightly increases the error of the numerical solution in comparison with the EI variant and the experimental order of convergence remains almost the same. Based on these results, both the EI and ML variants of the scheme can be reliably used for the solution of problems in homogeneous porous media.

Id.	Brooks-Corey				van Genuchten			
	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$
$2D_1^{\triangle}$	$1.45 \cdot 10^{-2}$	**0.92**	$3.17 \cdot 10^{-2}$	**0.78**	$1.42 \cdot 10^{-2}$	**0.98**	$2.12 \cdot 10^{-2}$	**0.94**
$2D_2^{\triangle}$	$7.94 \cdot 10^{-3}$	**0.78**	$1.91 \cdot 10^{-2}$	**0.60**	$7.51 \cdot 10^{-3}$	**0.86**	$1.15 \cdot 10^{-2}$	**0.84**
$2D_3^{\triangle}$	$4.40 \cdot 10^{-3}$	**0.95**	$1.21 \cdot 10^{-2}$	**0.69**	$3.93 \cdot 10^{-3}$	**1.05**	$6.11 \cdot 10^{-3}$	**1.03**
$2D_4^{\triangle}$	$2.41 \cdot 10^{-3}$	**0.85**	$7.84 \cdot 10^{-3}$	**0.66**	$2.03 \cdot 10^{-3}$	**0.90**	$3.19 \cdot 10^{-3}$	**0.89**
$2D_5^{\triangle}$	$1.30 \cdot 10^{-3}$		$4.85 \cdot 10^{-3}$		$1.06 \cdot 10^{-3}$		$1.68 \cdot 10^{-3}$	
$2D_1^{\square}$	$1.52 \cdot 10^{-2}$	**0.80**	$3.26 \cdot 10^{-2}$	**0.65**	$1.41 \cdot 10^{-2}$	**0.84**	$2.17 \cdot 10^{-2}$	**0.81**
$2D_2^{\square}$	$8.75 \cdot 10^{-3}$	**0.82**	$2.08 \cdot 10^{-2}$	**0.62**	$7.88 \cdot 10^{-3}$	**0.87**	$1.24 \cdot 10^{-2}$	**0.86**
$2D_3^{\square}$	$4.97 \cdot 10^{-3}$	**0.85**	$1.35 \cdot 10^{-2}$	**0.60**	$4.31 \cdot 10^{-3}$	**0.88**	$6.83 \cdot 10^{-3}$	**0.88**
$2D_4^{\square}$	$2.76 \cdot 10^{-3}$	**0.87**	$8.93 \cdot 10^{-3}$	**0.63**	$2.34 \cdot 10^{-3}$	**0.86**	$3.72 \cdot 10^{-3}$	**0.85**
$2D_5^{\square}$	$1.51 \cdot 10^{-3}$		$5.79 \cdot 10^{-3}$		$1.29 \cdot 10^{-3}$		$2.06 \cdot 10^{-3}$	

Table 5.3: Errors of the numerical solution and experimental orders of convergence in 2D for the benchmark problem described in Section 5.1. EI variant of the method.

Id.	Brooks-Corey				van Genuchten			
	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$
$2D_1^{\triangle}$	$1.48 \cdot 10^{-2}$	**0.91**	$3.22 \cdot 10^{-2}$	**0.76**	$1.44 \cdot 10^{-2}$	**0.98**	$2.16 \cdot 10^{-2}$	**0.95**
$2D_2^{\triangle}$	$8.17 \cdot 10^{-3}$	**0.77**	$1.96 \cdot 10^{-2}$	**0.59**	$7.59 \cdot 10^{-3}$	**0.86**	$1.17 \cdot 10^{-2}$	**0.85**
$2D_3^{\triangle}$	$4.56 \cdot 10^{-3}$	**0.96**	$1.25 \cdot 10^{-2}$	**0.69**	$3.95 \cdot 10^{-3}$	**1.04**	$6.15 \cdot 10^{-3}$	**1.04**
$2D_4^{\triangle}$	$2.49 \cdot 10^{-3}$	**0.86**	$8.10 \cdot 10^{-3}$	**0.68**	$2.04 \cdot 10^{-3}$	**0.90**	$3.20 \cdot 10^{-3}$	**0.89**
$2D_5^{\triangle}$	$1.33 \cdot 10^{-3}$		$4.96 \cdot 10^{-3}$		$1.06 \cdot 10^{-3}$		$1.68 \cdot 10^{-3}$	
$2D_1^{\square}$	$1.61 \cdot 10^{-2}$	**0.82**	$3.44 \cdot 10^{-2}$	**0.61**	$2.15 \cdot 10^{-2}$	**1.21**	$3.23 \cdot 10^{-2}$	**1.13**
$2D_2^{\square}$	$9.16 \cdot 10^{-3}$	**0.84**	$2.25 \cdot 10^{-2}$	**0.62**	$9.26 \cdot 10^{-3}$	**1.04**	$1.47 \cdot 10^{-2}$	**1.02**
$2D_3^{\square}$	$5.14 \cdot 10^{-3}$	**0.86**	$1.46 \cdot 10^{-2}$	**0.63**	$4.52 \cdot 10^{-3}$	**0.96**	$7.28 \cdot 10^{-3}$	**0.96**
$2D_4^{\square}$	$2.84 \cdot 10^{-3}$	**0.88**	$9.49 \cdot 10^{-3}$	**0.65**	$2.32 \cdot 10^{-3}$	**0.92**	$3.74 \cdot 10^{-3}$	**0.92**
$2D_5^{\square}$	$1.54 \cdot 10^{-3}$		$6.02 \cdot 10^{-3}$		$1.22 \cdot 10^{-3}$		$1.98 \cdot 10^{-3}$	

Table 5.4: Errors of the numerical solution and experimental orders of convergence in 2D for the benchmark problem described in Section 5.1. ML variant of the method.

Id.	Brooks-Corey				van Genuchten			
	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$
$3D_1^{\triangle}$	$1.12 \cdot 10^{-2}$		$3.38 \cdot 10^{-2}$		$1.21 \cdot 10^{-2}$		$2.43 \cdot 10^{-2}$	
		0.69		**0.60**		**0.77**		**0.73**
$3D_2^{\triangle}$	$7.82 \cdot 10^{-3}$		$2.47 \cdot 10^{-2}$		$8.13 \cdot 10^{-3}$		$1.66 \cdot 10^{-2}$	
		0.84		**0.72**		**0.93**		**0.90**
$3D_3^{\triangle}$	$4.35 \cdot 10^{-3}$		$1.49 \cdot 10^{-2}$		$4.25 \cdot 10^{-3}$		$8.84 \cdot 10^{-3}$	
		1.03		**0.92**		**1.14**		**1.12**
$3D_4^{\triangle}$	$2.37 \cdot 10^{-3}$		$8.63 \cdot 10^{-3}$		$2.17 \cdot 10^{-3}$		$4.56 \cdot 10^{-3}$	
		0.82		**0.79**		**1.04**		**1.02**
$3D_5^{\triangle}$	$1.41 \cdot 10^{-3}$		$5.23 \cdot 10^{-3}$		$1.12 \cdot 10^{-3}$		$2.39 \cdot 10^{-3}$	
$3D_1^{\square}$	$8.28 \cdot 10^{-3}$		$2.59 \cdot 10^{-2}$		$8.15 \cdot 10^{-3}$		$1.64 \cdot 10^{-2}$	
		0.83		**0.70**		**0.88**		**0.86**
$3D_2^{\square}$	$4.67 \cdot 10^{-3}$		$1.59 \cdot 10^{-2}$		$4.42 \cdot 10^{-3}$		$9.06 \cdot 10^{-3}$	
		0.84		**0.69**		**0.90**		**0.89**
$3D_3^{\square}$	$2.60 \cdot 10^{-3}$		$9.87 \cdot 10^{-3}$		$2.36 \cdot 10^{-3}$		$4.90 \cdot 10^{-3}$	
		0.86		**0.69**		**0.93**		**0.92**
$3D_4^{\square}$	$1.44 \cdot 10^{-3}$		$6.12 \cdot 10^{-3}$		$1.24 \cdot 10^{-3}$		$2.58 \cdot 10^{-3}$	
		0.84		**0.61**		**0.96**		**0.92**
$3D_5^{\square}$	$8.05 \cdot 10^{-4}$		$4.01 \cdot 10^{-3}$		$6.40 \cdot 10^{-3}$		$1.37 \cdot 10^{-3}$	

Table 5.5: Errors of the numerical solution and experimental orders of convergence in 3D for the benchmark problem described in Section 5.1. EI variant of the method.

Id.	Brooks-Corey				van Genuchten			
	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$	$\|E_{h,S_n}\|_1$	$EOC_{S_n,1}$	$\|E_{h,S_n}\|_2$	$EOC_{S_n,2}$
$3D_1^{\triangle}$	$1.13 \cdot 10^{-2}$		$3.46 \cdot 10^{-2}$		$1.22 \cdot 10^{-2}$		$2.49 \cdot 10^{-2}$	
		0.67		**0.61**		**0.77**		**0.74**
$3D_2^{\triangle}$	$7.96 \cdot 10^{-3}$		$2.52 \cdot 10^{-2}$		$8.22 \cdot 10^{-3}$		$1.70 \cdot 10^{-2}$	
		0.82		**0.72**		**0.93**		**0.91**
$3D_3^{\triangle}$	$4.50 \cdot 10^{-3}$		$1.53 \cdot 10^{-2}$		$4.30 \cdot 10^{-3}$		$8.97 \cdot 10^{-3}$	
		1.01		**0.92**		**1.13**		**1.12**
$3D_4^{\triangle}$	$2.47 \cdot 10^{-3}$		$8.64 \cdot 10^{-3}$		$2.20 \cdot 10^{-3}$		$4.63 \cdot 10^{-3}$	
		0.83		**0.79**		**1.04**		**1.02**
$3D_5^{\triangle}$	$1.44 \cdot 10^{-3}$		$5.26 \cdot 10^{-3}$		$1.15 \cdot 10^{-3}$		$2.41 \cdot 10^{-3}$	
$3D_1^{\square}$	$9.03 \cdot 10^{-3}$		$2.68 \cdot 10^{-2}$		$8.97 \cdot 10^{-3}$		$1.75 \cdot 10^{-2}$	
		0.84		**0.65**		**0.95**		**0.89**
$3D_2^{\square}$	$5.05 \cdot 10^{-3}$		$1.71 \cdot 10^{-2}$		$4.66 \cdot 10^{-3}$		$9.47 \cdot 10^{-3}$	
		0.85		**0.65**		**0.94**		**0.91**
$3D_3^{\square}$	$2.80 \cdot 10^{-3}$		$1.09 \cdot 10^{-3}$		$2.43 \cdot 10^{-3}$		$5.05 \cdot 10^{-3}$	
		0.87		**0.67**		**0.95**		**0.93**
$3D_4^{\square}$	$1.53 \cdot 10^{-3}$		$6.84 \cdot 10^{-3}$		$1.25 \cdot 10^{-3}$		$2.63 \cdot 10^{-3}$	
		0.86		**0.67**		**0.95**		**0.91**
$3D_5^{\square}$	$8.47 \cdot 10^{-4}$		$4.31 \cdot 10^{-3}$		$6.49 \cdot 10^{-4}$		$1.40 \cdot 10^{-3}$	

Table 5.6: Errors of the numerical solution and experimental orders of convergence in 3D for the benchmark problem described in Section 5.1. ML variant of the method.

5.2 Heterogeneous benchmark problem

In this section, the numerical scheme is benchmarked using the problem of two-phase flow in a heterogeneous porous medium. The problem was originally proposed in [62] to demonstrate the barrier effect of two-phase flow in a heterogeneous porous medium that was discussed in Section 2.3.8. In the case of the flow of the non-wetting phase from coarse to fine sand, the entry pressure for the fine sand needs to be reached before the non-wetting phase can penetrate the fine sand region. For this problem, there is no known way to find the exact solution that could be used as a reference for the error and experimental order of convergence computations similarly to the homogeneous case discussed in Section 5.1. Nevertheless, we are interested in investigating whether the method can correctly capture the flow across material interfaces and visually compare the numerical solution to the results reported in the literature obtained by a different numerical method that is in agreement with the experimental observations [62].

The problem setup is shown in Figure 5.4 and consists of three layers of sand: the middle one (Sand C) is finer than the remaining two (Sand B), initially fully saturated with water. NAPL is injected along the upper boundary with a given flux and both fluids are assumed to be incompressible. The origin of the coordinate system is selected in the top left corner of the domain and the gravity acts in the y direction. The sand and fluid properties are given in Table 5.7. The following boundary conditions are prescribed

$$\Gamma_1: \quad \boldsymbol{u}_n \cdot \boldsymbol{n} = -3.57 \cdot 10^{-5}\ \mathrm{m\,s^{-1}}, \quad \boldsymbol{u}_w \cdot \boldsymbol{n} = 0, \tag{5.9}$$

$$\Gamma_2 \cup \Gamma_4: \quad \boldsymbol{u}_n \cdot \boldsymbol{n} = 0, \quad \boldsymbol{u}_w \cdot \boldsymbol{n} = 0, \tag{5.10}$$

$$\Gamma_3: \quad S_w = 1, \quad p_n = 2 \cdot 10^5 + 4905\ \mathrm{Pa}. \tag{5.11}$$

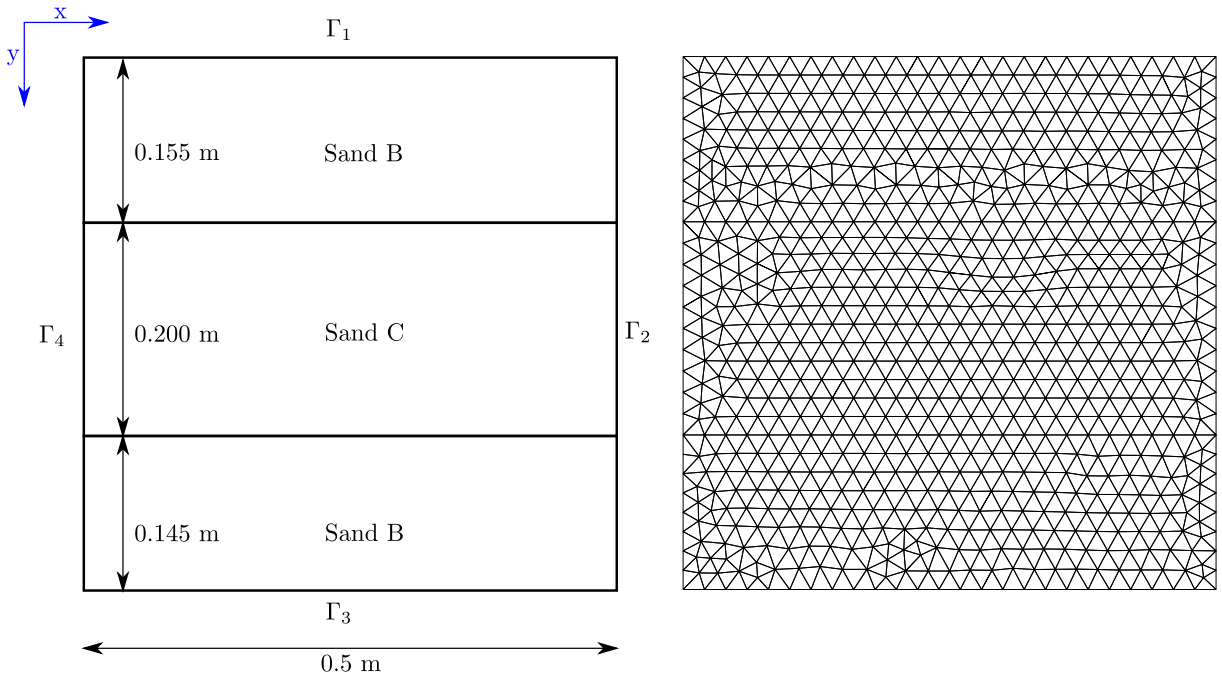


Figure 5.4: Computational domain (left) and the coarsest triangular mesh consisting of 1506 elements (right) for the benchmark problem in heterogeneous porous media described in Section 5.2.

Similarly to the McWhorter-Sunada problem in Section 5.1, the capillary pressure and the non-wetting phase pressure are selected as the primary unknowns in (3.1) i.e., $Z_1 = p_c$ and

Sand	ϕ [–]	K [m^2]	S_{rw} [–]	S_{rn} [–]	p_d [Pa]	λ^{BC} [–]
Sand B	0.4	$5.04 \cdot 10^{-10}$	0.08	0	370	3.86
Sand C	0.343	$5.168 \cdot 10^{-12}$	0.04	0	4605.8	2.857

Fluid	ρ [kg m^{-3}]	μ [Pa s^{-1}]
Water	1000	$1 \cdot 10^{-4}$
NAPL	1400	$1 \cdot 10^{-4}$

Table 5.7: Sand and fluid properties for the benchmark problem in heterogeneous porous media described in Section 5.2. Adapted from [62].

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt [s]
1	$1.34 \cdot 10^{-2}$	1 506	2 310	4 620	2
2	$6.99 \cdot 10^{-3}$	5 886	8 930	17 860	1
3	$3.45 \cdot 10^{-3}$	23 308	35 162	70 324	0.5

Table 5.8: Mesh properties for the heterogeneous benchmark problem described in Section 5.2 and time steps for each mesh.

$Z_2 = p_n$. In addition to the McWhorter-Sunada problem, this problem includes the effects of gravity. For the aforementioned choice of the primary variables, the problem is represented by the following non-zero coefficients in the general formulation (3.1)

$$\mathcal{N} = \begin{pmatrix} -\Phi\rho_w \frac{dS_w}{dp_c} & 0 \\ -\Phi\rho_n \frac{dS_w}{dp_c} & 0 \end{pmatrix}, \quad \mathcal{M} = \begin{pmatrix} \rho_w \frac{\lambda_w}{\lambda_t} \\ \rho_n \frac{\lambda_n}{\lambda_t} \end{pmatrix}, \quad \mathcal{D} = \begin{pmatrix} \lambda_t \mathbb{K} & -\lambda_t \mathbb{K} \\ 0 & \lambda_t \mathbb{K} \end{pmatrix}, \quad \mathcal{W} = \begin{pmatrix} -\rho_w \lambda_t \mathbb{K} \boldsymbol{g} \\ \rho_n \lambda_t \mathbb{K} \boldsymbol{g} \end{pmatrix}. \quad (5.12)$$

The reference numerical solution taken from [41] was obtained using the vertex-centered finite volume method in 1D on a very fine mesh and is in a good match with the results provided in [62]. Solving the problem in 1D allows for obtaining high spatial resolution and the 1D solution is sufficient for the comparison as the problem configuration shown in Figure 5.4 is essentially one-dimensional in the direction of the y axis.

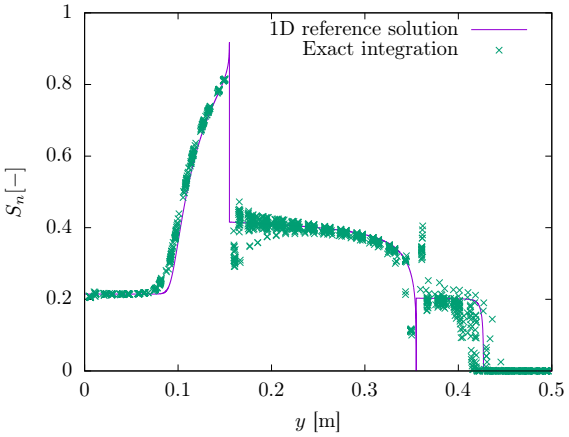
The solutions are compared at time $t = 1650$ s. A single cross-section through the center of the 2D domain is not suitable for the comparison between the 1D and 2D cases because these results are significantly affected by the alignment of the triangular mesh and the cross-section line. Therefore, to demonstrate the behavior of the solution in the vicinity of the interfaces, the superposed values from all elements of the 2D mesh are taken into account, with the y coordinate of their center as the 1D coordinate. To ensure that the observed artifacts are not caused by the mesh, the problem was solved on three meshes with properties given in Table 5.8. All these meshes were generated by `Gmsh` [49] and the computations were performed on a PC equipped with Intel Core i7-5820K, 3.6 GHz processor and 32 GB RAM.

5.2.1 Results

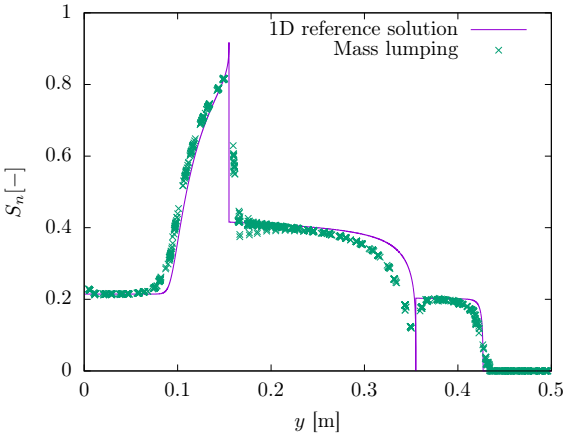
Numerical results for the EI variant of the method are compared to the 1D reference solution in Figures 5.5a, 5.5c, and 5.5e. The oscillations appear at both sand interfaces, and are more apparent in the case of flow from the fine to coarse sand. These oscillations are present on several mesh refinements, therefore, they cannot be attributed to the mesh properties. On all meshes, the 2D numerical solution significantly differs from the reference 1D solution. This indicates that the EI variant of the numerical method presented in Chapter 3 cannot correctly capture the flow through the material interfaces and, therefore, it is not suitable for simulating the two-phase flow in heterogeneous porous media.

Numerical results for the ML variant of the method are shown in Figures 5.5b, 5.5d, and 5.5f. In comparison with the EI variant, the oscillations at both material interfaces are significantly reduced and the numerical solution is in a good match with the reference solution. This indicates that the ML variant of the numerical method is necessary to correctly capture the flow through material interfaces and as such, it is employed in the rest of this work for solving problems in heterogeneous porous media.

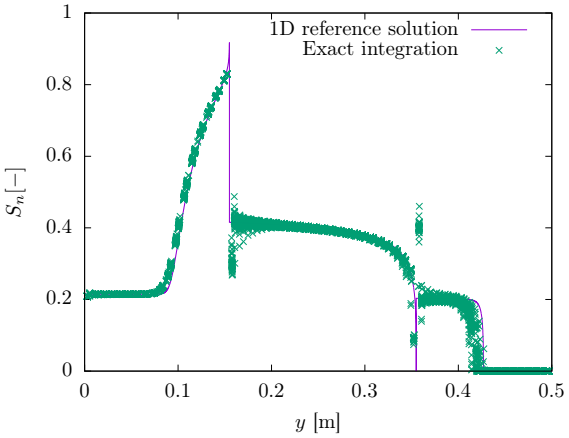
The application of the MHFEM-based solver to this problem was studied also in [41], where a similar behavior at material interfaces was observed. The approach proposed to eliminate the oscillations in [41] required to explicitly formulate the barrier effect condition as described in Section 2.3.8. The advantage of the approach of employing the mass-lumping technique used here is that no special treatment of the interface sides is needed. Additionally, this simplifies the implementation of the numerical scheme, especially in the case of more complex geometries and cases when the changes in flow direction occur frequently.



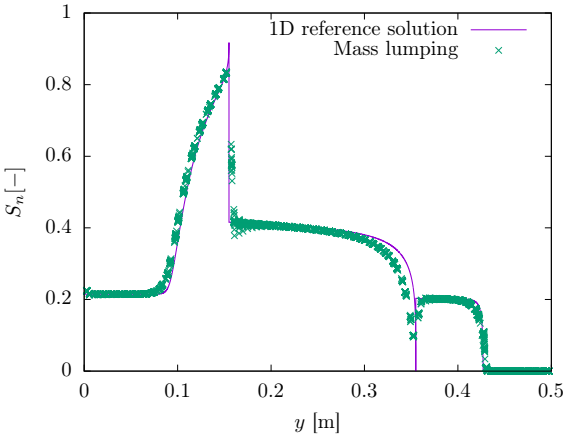
(a) 1 506 △, exact integration (EI).



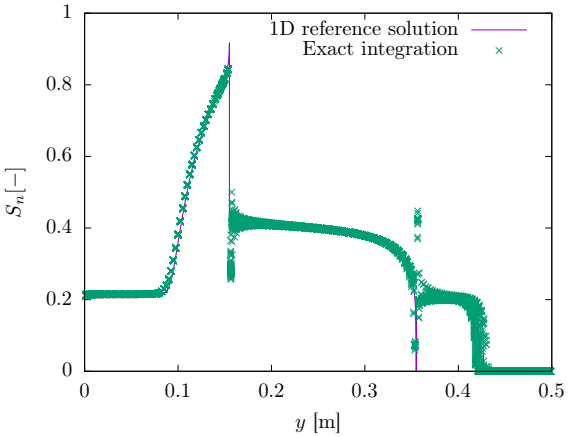
(b) 1 506 △, mass lumping (ML).



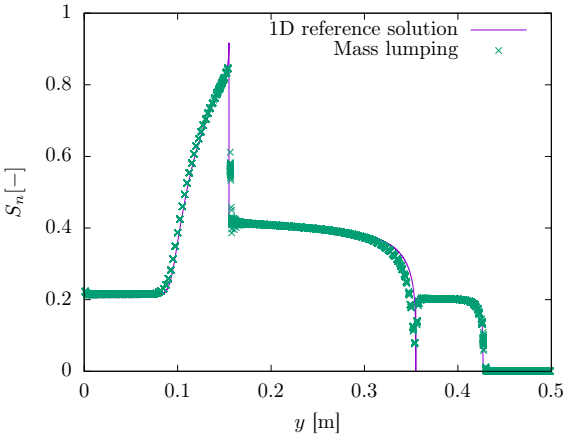
(c) 5 886 △, exact integration (EI).



(d) 5 886 △, mass lumping (ML).



(e) 23 308 △, exact integration (EI).



(f) 23 308 △, mass lumping (ML).

Figure 5.5: Comparison between the reference 1D solution and the EI and ML variants of the method on various meshes for the superposed solution of the benchmark problem in heterogeneous porous media described in Section 5.2.

5.3 Chapter summary

In this chapter, we verified the numerical scheme on benchmark problems of two-phase flow in porous media with a known solution.

In the homogeneous porous medium, the generalization of the McWhorter-Sunada problem in 2D and 3D with the semi-analytical solution was used. We performed a convergence study and showed that the numerical method is convergent with the first order of accuracy. These results hold for both approaches to computing the local mesh coefficients: exact integration and the mass lumping techniques, the latter leading to slightly larger errors of the numerical solution. The results of this convergence study for the EI variant of the method were published in [43]. This variant of the method is used in this work in the case of a homogeneous medium later in Chapter 6.

In the heterogeneous porous medium, we showed that using the exact integration approach to compute the entries of the local coefficient matrices leads to non-physical oscillations on the material interfaces. These oscillations are eliminated using the mass lumping technique. This variant of the numerical scheme gives good results without the need of modifying the general formulation of the numerical scheme to explicitly take into account the capillary barrier effects that was proposed earlier in [41]. The results of employing the mass-lumping technique on unstructured meshes were published in [102]. The ML variant of the method is used in the rest of this work when the problems in heterogeneous porous media are solved later in Chapters 7 and 8.

Parallel benchmarks 6

In this chapter, the performance of the parallel implementation presented in Chapter 4 is investigated. The serial implementation was used so far and we were interested mainly in the accuracy and convergence properties of the method and not in the computational times. However, for finer meshes, the serial computations become more time-consuming and problems of a certain size cannot be solved by the serial variant of the method on the available hardware. To overcome these limitations and speed up the computation on finer meshes, a parallel implementation based on the domain decomposition method is used as described in Chapter 4. The main goal in this chapter is to demonstrate that the parallel implementation preserves a good efficiency for the increasing number of CPU cores used. For this demonstration, both weak and strong scaling tests are carried out.

The results presented in this chapter are divided into two parts. Firstly, the parallel implementation is verified on the Laplace problem both in 2D and 3D in Section 6.2.

Then, the performance of the parallel implementation is demonstrated on the McWhorter-Sunada problem as in Section 5.1. The MHFEM discretization of this unsteady problem leads to the solution of a linear problem with a non-symmetric matrix at each time step. Therefore, in contrast with the Laplace problem, a sequence of linear problems needs to be solved to obtain a solution at a given time. The performance of the solver is studied on a solution of a single linear system in Section 6.3 referred to as the steady problem, where the matrix $\mathbb{A}$ obtained at a single selected time step is considered. This setting for suitable meshes allows for the comparison with the Laplace problem resulting in symmetric matrix $\mathbb{A}$.

Furthermore, various approaches to the solution of the whole sequence of linear problems needed to obtain the solution at a given time, referred to as the unsteady problem, are studied in Section 6.4. Here, we focus on using the information about the solution and preconditioner from previous time steps to speed up the solution of the linear problem at the current time step.

6.1 Computations setup

The benchmark problems used in this chapter are considered both in 2D and 3D on structured and unstructured meshes. The numbers of elements and sides for all the meshes used in this chapter are shown in Table 6.1.

The meshes are numbered from 1 to 9 for all the cases considered. For the MHFEM discretization employed here, the resulting linear problem is formulated using traces of the unknown functions, therefore, the degrees of freedom are located on the mesh sides and the number of sides of each mesh controls the size of the linear problem. Note that there is one degree of freedom per side for the Laplace problem and two per side for the McWhorter-Sunada problem. To allow for comparison among the meshes, each mesh is generated to get approximately the

same number of sides for both structured and unstructured cases, and for the scaling tests, the number of sides is approximately doubled between successive meshes. All the unstructured meshes are generated by `Gmsh` [49].

The weak and strong scaling tests for steady problems comprising the Laplace problem and single time step of the McWhorter-Sunada problem were performed on the `Salomon` cluster at the IT4Innovations National Supercomputing Center in Ostrava, Czech Republic. The computational nodes of `Salomon` were equipped with two 12-core Intel Xeon E5-2680v3, 2.5 GHz processors and 128 GB RAM. The computations of the unsteady problems were run on the `Helios` cluster at the Czech Technical University in Prague, Czech Republic. The computational nodes of `Helios` were equipped with two 16-core AMD EPYC 7281, 2.1GHz processors and 128 GB RAM.

Id	1	2	3	4	5	6	7	8	9
2D unstructured									
# Elements	227	454	903	1 806	3 611	7 222	14 437	28 878	57 743
# Sides*	341	682	1 699	2 710	5 419	10 836	21 661	43 324	86 624
2D structured									
# Elements	171	346	686	1 383	2 742	5 532	10 811	21 623	43 244
# Sides*	344	693	1 373	2 768	5 488	11 069	21 628	43 254	86 501
3D unstructured									
# Elements	164	305	684	1 345	2 593	5 262	10 528	21 314	42 666
# Sides*	337	625	1 392	2 730	5 247	10 629	21 220	42 890	85 762
3D structured									
# Elements	111	216	1 334	885	1 728	3 512	7 078	14 172	28 094
# Sides*	339	659	1 334	2 682	5 227	10 605	21 344	42 693	84 561

* The number of degrees of freedom is equal to the # Sides for the Laplace problem, whereas it is $2\times$ # Sides for the two-phase flow problem.

Table 6.1: Number (in thousands) of elements and sides for all the meshes used in the parallel benchmarks in Chapter 6.

As the problems in homogeneous media are considered only, the EI variant of the method is used in this chapter. The iterations of the linear solver are terminated when the relative norm of the residual of the Schur complement system (4.3a) is lower than a given tolerance ϵ

$$\frac{\left\| \mathbb{S}\boldsymbol{Z}_q^{\Gamma,k+1} - \hat{\boldsymbol{b}} \right\|}{\left\| \hat{\boldsymbol{b}} \right\|} < \epsilon, \tag{6.1}$$

where $\boldsymbol{Z}_q^{\Gamma,k+1}$ is the solution for the degrees of freedom located on the interface obtained in q-th iteration. The value of $\epsilon = 10^{-9}$ is selected.

6.1.1 Metrics to evaluate the parallel performance

Here, we recall that one-by-one mapping is used, where each subdomain is assigned to one `MPI` process which is mapped to a single CPU core, see Section 4.2. The performance in the weak and strong scaling test tests is evaluated using the following metrics.

For the weak scaling test, the size of the local problem, i.e., the number of degrees of freedom per subdomain, is kept constant while the number of subdomains increases, i.e., the mesh number $p+1$ is divided into twice as many subdomains than the mesh number p. The efficiency Eff_m for the weak scaling test using m cores is defined by

$$Eff_m = \frac{t_{n_{ref}}}{t_m} \cdot 100\%, \tag{6.2}$$

where $t_{n_{ref}}$ is the reference computational time using n_{ref} cores and t_m is the computational time using m cores. Here, $n_{ref} = 24$ is selected which corresponds to a single fully allocated node of `Salomon`. In the ideal case, the efficiency is 100%, i.e., the computational time of the twice larger problem divided into twice as many subdomains remains the same as the computational time of the original problem.

For the strong scaling test, the size of the overall problem is kept constant while the number of subdomains increases, i.e., a selected fixed mesh is divided into an increasing number of subdomains. The speed-up Sp_m for the strong scaling test using m cores is defined by

$$Sp_m = \frac{t_{n_{ref}}}{t_m}, \tag{6.3}$$

where $t_{n_{ref}}$ is the reference computational time using n_{ref} cores and t_m is the computational time using m cores. Here, in contrast to the weak scaling test, the value of n_{ref} is not the same for all the problems. It is selected as the lowest feasible number of cores on which the computation for the given problem can be run. The following values are selected $n_{ref} = 24$, $n_{ref} = 48$, and $n_{ref} = 96$ for the Laplace problem, 2D McWhorter-Sunada problem, and 3D McWhorter-Sunada problem, respectively. Computations using a lower number of cores cannot be run due to the memory requirements of the linear solvers for the local problems. In the ideal case, the speed-up using m cores is $\frac{m}{n_{ref}}$, i.e., the computational time using twice as many CPU cores is half of the original one. The case of a speed-up higher than the optimal value is called the *superlinear speed-up*.

6.2 Symmetric case

The Laplace problem used as the verification for the parallel implementation consists of a single elliptic partial differential equation ($n = 1$) for the unknown function Z_1 in the form

$$\Delta Z_1(\boldsymbol{x}) = 0, \qquad \forall \boldsymbol{x} \in \Omega, \tag{6.4}$$

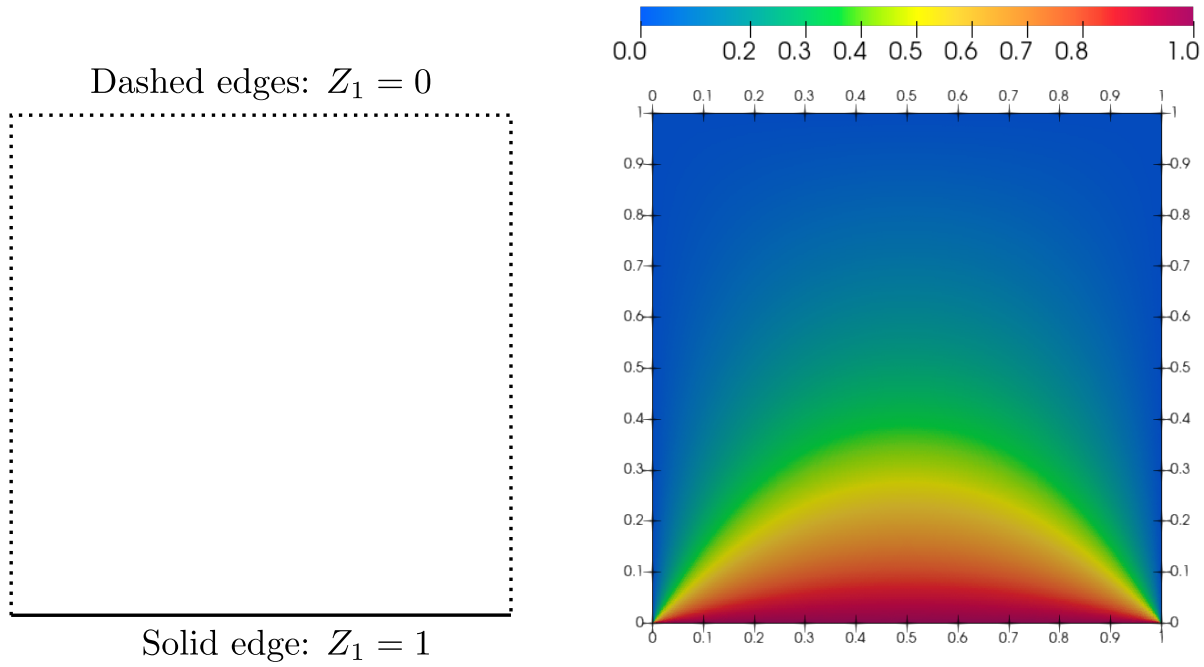
and is represented by the following non-zero coefficients in the general formulation of the problem (3.1),

$$\mathcal{M} = \left(1\right), \qquad \mathcal{D} = \left(\mathbb{I}\right). \tag{6.5}$$

This standard steady problem discretized by MHFEM results in a solution of a single sparse linear system with a symmetric matrix that needs to be solved to obtain the solution. Application of the BDDC method to linear problems with a symmetric positive definite matrix is largely

studied in the domain decomposition literature, for example [110]. The scaling tests of the implementation for this problem are presented here to verify that the implementation works according to expectations. More importantly, the computational times for the symmetric problems are used as reference ones to demonstrate the difference in complexity between the symmetric and non-symmetric problems of the same size arising from the discretization of different systems of governing equations.

The computational domains are a square with a 1 meter long side and a cube with a 1 meter long edge in 2D and 3D cases, respectively, with the following boundary conditions. In 2D, on three edges, the zero Dirichlet boundary condition $Z_1 = 0$ is prescribed. On the remaining edge, the Dirichlet boundary condition $Z_1 = 1$ is prescribed. In 3D, on five faces, the zero Dirichlet boundary condition $Z_1 = 0$ is prescribed. On the remaining face, the Dirichlet boundary condition $Z_1 = 1$ is prescribed. These boundary conditions in 2D and 3D are illustrated in Figures 6.1 and 6.2, respectively, where also the numerical solutions of the Laplace problem with these boundary conditions are shown.



(a) Boundary conditions.

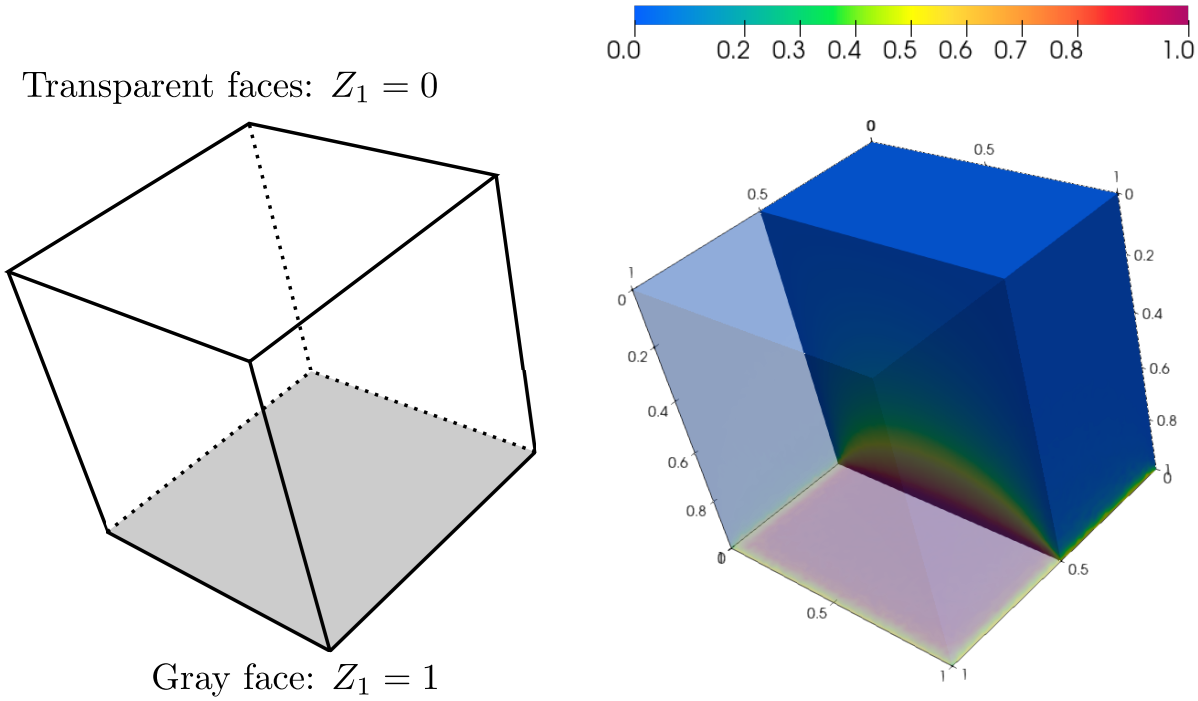
(b) Numerical solution.

Figure 6.1: Boundary conditions and numerical solution for the 2D Laplace problem considered in Section 6.2.

The problem is solved in 2D and 3D on both structured and unstructured meshes using all the meshes numbered from 1 to 9 in Table 6.1. The PCG iterative solver for the symmetric Schur complement matrix starts from the zero initial vector for all the computations in this section.

6.2.1 Weak scaling test

The number of subdomains in the weak scaling tests for each mesh is selected to get approximately 57 thousand degrees of freedom per subdomain. In addition to the overall computational time and the total number of PCG iterations needed for the convergence of the linear solver, we measure the times required to set up the BDDC method and to perform the PCG iterations separately. For the symmetric matrix $\mathbb{A}$, the setup comprises the following steps:



(a) Boundary conditions. (b) Numerical solution.

Figure 6.2: Boundary conditions and numerical solution for the 3D Laplace problem considered in Section 6.2.

1. Factorization of $\mathbb{A}_i^{II}$ needed to compute the action of the Schur complement matrix $\mathbb{S}_i$ in Algorithm 8.

2. Factorization of the matrices in (4.10) used for the local corrections.

3. Computation of the coarse basis functions $\mathbf{\Phi}_i$ in (4.11).

4. Assembly and factorization of the coarse problem matrix $\mathbb{S}_C$ in (4.19).

The computational times and the total number of iterations for the 2D unstructured, 3D unstructured, 2D structured, and 3D structured meshes are shown in Tables 6.2, 6.3, 6.4, and 6.5, respectively. In the case of structured meshes, both the division obtained by `METIS` and the structured division are considered.

The results presented here indicate that the implementation of the BDDC method applied to the linear problem with a symmetric matrix preserves a good efficiency of up to 768 subdomains and starts to deteriorate for 1536 subdomains. The number of PCG iterations is similar for the given size of the linear problem regardless of the mesh type and dimension of the problem. For all cases, there is an increase in the number of iterations with the first few mesh refinements and then the number of iterations remains almost constant.

6.2.2 Strong scaling test

A strong scaling test is performed on an unstructured mesh number 6 with approximately 11 million of unknowns both in 2D and 3D. The reference computation is run on a division into 24 subdomains which corresponds to one fully allocated node of the `Salomon` cluster. The local

Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	0.52	0.55	0.63	0.64	0.60	0.92	0.97	1.03	1.47
Iterative solution [s]	0.16	0.30	0.29	0.46	0.55	0.70	0.92	1.13	1.36
Solution total [s]	0.68	0.85	0.92	1.11	1.15	1.57	1.89	2.17	2.83
#PCG Iterations [-]	15	18	18	20	20	20	20	21	22
Efficiency $\boldsymbol{Eff}$ [%]	136	109	**100**	83	80	59	49	42	33

Table 6.2: 2D symmetric problem. Weak scaling test on unstructured meshes.

Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	2.80	3.73	4.57	4.73	4.61	4.87	4.92	5.80	7.55
Iterative solution [s]	0.52	0.56	0.74	0.97	1.25	1.51	1.74	2.03	16.94
Solution total [s]	3.31	4.29	5.31	5.70	5.86	6.38	6.67	7.82	24.49
#PCG Iterations [-]	17	17	19	19	19	21	22	23	23
Efficiency $\boldsymbol{Eff}$ [%]	160	124	**100**	93	91	83	80	68	22

Table 6.3: 3D symmetric problem. Weak scaling test on unstructured meshes.

matrices for the divisions into fewer subdomains become too large to be factorized by the `CHOLMOD` library due to excessive memory requirements.

Similarly to the weak scaling test in Section 6.2.1, we are interested in the computational times, the total number of PCG iterations, and the speed-up. The results are shown in Tables 6.6 and 6.7 for the 2D and 3D cases, respectively.

These results show a superlinear speed-up for the initial mesh refinements and, then, the speed-up significantly drops for 1536 subdomains.

This effect is mainly caused by the nonlinear complexity of the direct solvers. For a smaller number of subdomains, the local problems that are solved independently in parallel are large in comparison with the coarse problem and their solution takes most of the overall computational time. However, with the increasing number of subdomains, the local problems become smaller, and eventually the coarse problem that is solved in serial dominates. The number of PCG iterations remains almost constant for all cases during the whole strong scaling test.

The results presented in this section tend to confirm the previous observations, e.g., [107] that the scalability of the two-level BDDC method significantly worsens around a thousand subdomains/cores, motivating the use of the three-level BDDC method for higher core-counts. A detailed discussion of the results is provided together with the non-symmetric case later in Section 6.3.

Rectangles									
Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	0.56	0.68	0.64	0.75	0.76	0.93	0.88	1.34	1.53
Iterative solution [s]	0.15	0.29	0.38	0.54	0.62	0.73	0.78	1.17	1.24
Solution total [s]	0.71	0.97	1.02	1.29	1.37	1.66	1.67	2.51	2.77
#PCG Iterations [-]	8	13	18	19	19	19	19	19	19
Efficiency $\boldsymbol{Eff}$ [%]	143	105	**100**	79	74	61	61	41	37
`METIS`									
Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	0.49	0.61	0.64	0.71	0.74	0.84	0.81	0.88	1.28
Iterative solution [s]	0.21	0.35	0.39	0.54	0.63	0.72	0.93	1.06	1.31
Solution total [s]	0.70	0.98	1.03	1.26	1.37	1.56	1.74	1.95	2.59
#PCG Iterations [-]	16	17	19	21	20	20	20	20	21
Efficiency $\boldsymbol{Eff}$ [%]	147	105	**100**	82	76	66	60	53	40

Table 6.4: 2D symmetric problem. Weak scaling test on structured meshes.

6.3 Non-symmetric case: a steady problem

The generalization of the McWhorter-Sunada problem is used as the benchmark problem, with its MHFEM discretization resulting in the linear system with a non-symmetric matrix in (3.57). The problem is the same as the one used for the convergence study in Chapter 5. For a detailed description of the problem, material properties, computational domain, and boundary conditions, see Section 5.1.

In this section, we focus on a single time step of the problem and demonstrate the efficiency of the BDDC solver for a single non-symmetric matrix. All the results presented in this section are for the linear system obtained at the first time step of the McWhorter-Sunada problem and the BiCGStab iterations starting from the zero initial vector.

In Section 6.4, where the complete unsteady problem is solved, we demonstrate that the restriction to the first time step done in this section is justifiable to show the performance of the BDDC solver and that the linear problem at the first time step is a good representation of the linear problems arising during the solution of the unsteady McWhorter-Sunada problem. From the reference computation done in Section 6.4, it follows that the complexity of the linear problems at each time step remains almost the same as the problem at the first time step.

6.3.1 Weak scaling test

For non-symmetric matrices, we also start with a weak scaling test. The underlying McWhorter-Sunada problem consists of a system of two equations, therefore, there are two degrees of freedom

Cuboids									
Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	4.36	7.65	7.55	8.34	7.96	7.76	8.48	9.25	8.25
Iterative solution [s]	0.45	0.77	1.07	1.20	1.35	1.63	1.83	2.17	2.12
Solution total [s]	4.81	8.43	8.62	9.55	9.32	9.39	10.31	11.43	10.71
#PCG Iterations [-]	12	13	18	18	19	22	21	21	23
Efficiency $\boldsymbol{Eff}$ [%]	179	102	**100**	90	92	92	84	75	80
`METIS`									
Mesh Id	1	2	3	4	5	6	7	8	9
Subdomains	6	12	24	48	96	192	384	768	1536
Setup [s]	5.91	8.87	8.93	10.49	10.58	11.63	12.58	14.49	15.7
Iterative solution [s]	0.58	0.99	1.11	1.32	1.60	1.86	2.29	2.72	18.49
Solution total [s]	6.49	9.87	10.18	11.81	12.18	13.50	14.87	16.24	34.19
#PCG Iterations [-]	15	16	18	18	20	20	23	24	23
Efficiency $\boldsymbol{Eff}$ [%]	157	103	**100**	86	84	75	68	63	30

Table 6.5: 3D symmetric problem. Weak scaling test on structured meshes.

Subdomains	24	48	96	192	384	768	1536
Setup [s]	13.64	4.63	1.74	0.92	0.41	0.20	0.31
Iterations [s]	3.39	1.72	0.98	0.70	0.72	0.76	0.89
Solution total [s]	17.05	6.35	2.72	1.57	1.14	0.97	1.20
#PCG Iterations [-]	21	22	21	20	20	19	19
Speed-up $\boldsymbol{Sp}$ [-]	**1.00**	2.68 *(2)*	6.26 *(4)*	10.87 *(8)*	15.00 *(16)*	17.64 *(32)*	14.25 *(64)*

Table 6.6: 2D symmetric problem. Strong scaling test on unstructured mesh number 6. The optimal speed-up values are given in parentheses.

corresponding to each side. To obtain the same sizes of matrix $\mathbb{A}$ as for the Laplace problem, the meshes 1 to 8 are used only. Each mesh is divided into twice as many subdomains as for the same mesh in the Laplace problem to get approximately 57 thousand degrees of freedom per subdomain and, therefore, comparable sizes of the local problems.

Similarly to the symmetric case, in addition to the overall computational times and the total number of BiCGStab iterations, the computational times needed to set up the BDDC method, and to perform the BiCGStab iterations are measured separately. For the non-symmetric matrix

Subdomains	24	48	96	192	384	768	1536
Setup [s]	225.20	65.73	17.67	4.66	3.42	1.19	2.10
Iterations [s]	12.33	5.08	2.44	1.53	1.38	1.37	14.03
Solution total [s]	237.53	70.80	20.11	6.37	4.59	2.55	16.12
#PCG Iterations [-]	22	21	21	21	21	22	21
Speed-up Sp [-]	**1.00**	3.35 *(2)*	11.81 *(4)*	37.27 *(8)*	51.77 *(16)*	93.02 *(32)*	14.73 *(64)*

Table 6.7: 3D symmetric problem. Strong scaling test on unstructured mesh number 6. The optimal speed-up values are given in parentheses.

$\mathbb{A}$, the setup comprises the following steps:

1. Factorization of $\mathbb{A}_i^{II}$ needed to compute the action of the Schur complement matrix $\mathbb{S}_i$ in Algorithm 8.
2. Factorization of the matrices in (4.10) used for the local corrections.
3. Computation of the coarse basis functions $\mathbf{\Phi}_i$ in (4.11) and the adjoint coarse basis functions $\mathbf{\Phi}_i^*$ in (4.12).
4. Assembly and factorization of the coarse problem matrix $\mathbb{S}_C$ in (4.19).

The computational times and total number of iterations for the 2D unstructured, 3D unstructured, 2D structured, and 3D structured meshes are shown in Tables 6.8, 6.9, 6.10, and 6.11, respectively. In contrast to the unstructured meshes where only the division obtained by `METIS` is used for the structured meshes, we also consider the regular division into rectangular (2D) or cuboidal (3D) subdomains and compare the computational times and the total number of iterations for these two approaches in the same way as for the symmetric problem.

Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	5.75	6.43	6.42	6.58	6.70	6.91	7.69	7.59
Iterative solution [s]	0.70	0.80	0.86	1.10	1.26	1.42	2.14	14.03
Solution total [s]	6.47	7.24	7.30	7.69	8.00	8.33	9.85	21.63
#BiCGStab Iterations	10	10	11	11	12	11	12	11
Efficiency Eff [%]	112	**100**	99	94	91	87	74	33

Table 6.8: 2D non-symmetric problem. Weak scaling test on unstructured meshes.

The results presented here indicate that the efficiency of the implementation of the BDDC method applied to a problem with a non-symmetric matrix is comparable to the symmetric case. Good efficiency is preserved up to 768 subdomains and starts to deteriorate for 1536 subdomains. The numbers of BiCGStab iterations are similar for the given size of the linear problem regardless

Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	14.18	13.32	17.01	16.67	16.30	17.43	20.19	25.85
Iterative solution [s]	1.74	2.00	2.36	2.83	3.08	3.81	13.12	21.21
Solution total [s]	15.95	15.37	19.41	19.53	19.42	21.27	33.36	47.12
#BiCGStab Iterations	10	12	11	12	13	13	14	13
Efficiency $\boldsymbol{Eff}$ [%]	96	**100**	79	79	79	72	46	33

Table 6.9: 3D non-symmetric problem. Weak scaling test on unstructured meshes.

Rectangles								
Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	8.30	10.16	10.92	10.92	9.72	11.19	10.71	13.27
Iterative solution [s]	1.51	1.11	1.08	1.42	1.45	1.69	1.83	2.15
Solution total [s]	9.84	11.29	12.03	12.35	11.18	12.28	12.57	15.45
#BiCGStab Iterations	9	11	11	12	11	11	11	11
Efficiency $\boldsymbol{Eff}$ [%]	115	**100**	94	91	101	92	90	73
`METIS`								
Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	8.40	8.42	9.27	9.35	9.79	10.01	10.20	10.95
Iterative solution [s]	0.78	0.87	1.01	1.18	1.33	1.63	1.83	17.91
Solution total [s]	9.20	9.31	10.28	10.55	11.14	11.66	12.06	28.88
#BiCGStab Iterations [-]	9	10	10	10	10	11	11	11
Efficiency $\boldsymbol{Eff}$ [%]	101	**100**	91	88	84	80	77	32

Table 6.10: 2D non-symmetric problem. Weak scaling test on structured meshes.

of the mesh type and dimension of the problem. For all the cases, there is an increase in the number of iterations with the first few mesh refinements and then the number of iterations remains almost constant. The differences in the total number of iterations between the coarsest and the finest meshes are smaller than for the symmetric case.

The computational times for the problems with non-symmetric matrices are significantly longer than for the symmetric matrices of the same size. The weak scaling tests are designed to result in approximately the same size of local matrices for both symmetric and non-symmetric cases. However, for the McWhorter-Sunada problem, the matrices have twice as many non-zero

Cuboids								
Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	24.33	23.42	25.27	24.85	23.77	26.31	25.57	30.81
Iterative solution [s]	3.02	3.55	3.51	4.03	5.15	5.48	5.22	28.44
Solution total [s]	27.44	27.03	28.89	28.99	28.97	34.02	30.90	59.29
#BiCGStab Iterations	9	11	10	11	14	12	12	14
Efficiency $\boldsymbol{Eff}$ [%]	98	**100**	94	93	93	79	87	46
`METIS`								
Mesh Id	1	2	3	4	5	6	7	8
Subdomains	12	24	48	96	192	384	768	1536
Setup [s]	23.57	23.24	26.15	26.64	26.60	29.52	32.70	38.43
Iterative solution [s]	2.69	3.12	3.77	4.21	4.87	6.62	14.64	29.15
Solution total [s]	26.33	26.44	29.96	30.90	31.51	36.13	47.40	67.67
#BiCGStab Iterations [-]	9	10	11	11	12	14	14	16
Efficiency $\boldsymbol{Eff}$ [%]	100	**100**	88	86	84	73	56	39

Table 6.11: 3D non-symmetric problem. Weak scaling test on structured meshes.

entries on each line as for the Laplace problem due to being described by the system of two equations in the general formulation (3.1). In addition to the number of non-zero entries, both the local and coarse matrices are also non-symmetric. Therefore, the LU factorization needs to be used instead of the faster Cholesky factorization during the initialization step described in Section 4.5.1. Moreover, the non-symmetry of matrix $\mathbb{A}$ requires the computation of the adjoint coarse basis functions in the setup of the BDDC method.

For the 2D meshes, except for the finest one, the solution on structured meshes is slower than on the unstructured ones for the same number of degrees of freedom. For the structured meshes, the computations for the division produced by `METIS` are slightly faster than for the regular division. For the finest mesh with a significant drop in efficiency for all 2D meshes, the solution on the structured mesh becomes faster than on the unstructured one and the regular division of the structured mesh is faster than the `METIS` division.

Similarly to the symmetric case in Section 6.2, the solution of the 3D problem is significantly slower than the 2D one. The difference can be attributed to different structures of the matrices resulting from discretization in 2D and 3D and different numbers of non-zero entries in each matrix row that affect both the factorization of the matrices in the setup of the method and the backward and forward substitutions needed in each linear solver iteration. For the problem consisting of n equations in (3.1), in the 2D case, there are $\mathbf{5}n$ non-zero entries for triangular meshes and $\mathbf{7}n$ non-zero entries for rectangular meshes. In the 3D case, there are $\mathbf{7}n$ non-zero entries for tetrahedral meshes and $\mathbf{11}n$ non-zero entries for cuboidal meshes. Comparing Tables 6.8 and 6.9, the computation is about 3 times more costly in 3D than in 2D.

In contrast to the 2D case, in the 3D case, the difference between the structured and

unstructured meshes is more significant, favoring the unstructured meshes for the same number of degrees of freedom. Additionally, for the same structured mesh, the computational time for the regular division into cuboidal subdomains is shorter than for the division produced by `METIS`. This difference, similarly to the symmetric matrices, can be attributed to differences in the sizes of the interface and the size of the coarse problem. The mesh division produced by `METIS` minimizes the size of the interface, therefore, reduces `MPI` data transfers between the subdomains. However, it results in a geometrically more complex interface and, therefore, a significantly larger coarse problem, especially in 3D. The sizes of the coarse problems for the McWhorter-Sunada problem are shown in Table 6.12.

Subdomains	12	24	48	96	192	384	768	1536
Cuboids	40	92	208	448	944	1984	4096	8384
`METIS`	74	108	434	958	2174	4572	9706	20272

Table 6.12: Coarse problem size for the 3D McWhorter-Sunada problem on structured meshes.

6.3.2 Strong scaling test

The strong scaling test for the non-symmetric matrix is performed also on the unstructured mesh number 6, both in 2D and 3D. The reference computations are run on a division into 48 subdomains in 2D and 96 subdomains in 3D which correspond to two and four fully allocated nodes, respectively.

The computational times separately for the BDDC setup and BiCGStab iterations, total number of iterations, and speed-up are shown in Tables 6.13 and 6.14 for the 2D and 3D cases, respectively.

Subdomains	48	96	192	384	768	1536
Setup [s]	83.93	33.56	15.66	6.91	3.55	2.31
Iterations [s]	9.67	4.34	2.44	1.40	1.13	13.57
Solution total [s]	93.73	37.96	18.12	8.33	4.69	15.83
#BiCGStab Iterations [-]	13	12	12	11	11	11
Speed-up Sp [-]	**1.00**	2.47 *(2)*	5.17 *(4)*	11.25 *(8)*	19.99 *(16)*	5.92 *(32)*

Table 6.13: 2D non-symmetric strong scaling test on unstructured mesh number 6. The optimal speed-up values are given in parentheses.

The results in Tables 6.13 and 6.14 again suggest a superlinear speed-up for the first few increases in the number of subdomains due to the non-linear complexity of the `UMFPACK` solver together with the fact that for a small number of subdomains, the solution of the local problems dominates the overall computational time. With the increasing number of subdomains, the local problems become smaller and more time is spent on the solution of the coarse problem that is not parallelized in the implementation. As a result, the efficiency gradually decreases. From the comparison of the computational times for the division into 768 and 1536 subdomains, it can be

Subdomains	96	192	384	768	1536
Setup [s]	140.98	43.18	17.42	10.45	12.78
Iterations [s]	20.71	8.38	3.79	10.40	15.90
Solution total [s]	162.14	51.72	21.27	20.87	28.69
#BiCGStab Iterations [-]	14	13	13	13	14
Speed-up Sp [-]	**1.00**	3.13 *(2)*	7.62 *(4)*	7.77 *(8)*	5.65 *(16)*

Table 6.14: 3D non-symmetric strong scaling test on unstructured mesh number 6. The optimal speed-up values are given in parentheses.

seen that for a large number of subdomains, the computational time even starts to increase as the serial solution of the coarse problem becomes the dominant step of the solution. Similarly to the symmetric case, regardless of the computational times, the number of BiCGStab iterations remains almost constant for all the divisions into subdomains.

6.4 Non-symmetric case: an unsteady problem

In this section, the application of the BDDC method to the solution of a sequence of linear problems arising at each time step of the solution of the unsteady McWhorter-Sunada benchmark problem is investigated.

The performance of the BDDC method on a such single problem was studied in Section 6.3 where we stated that the first time step problem is a good representation of the complete sequence of problems. Here, several approaches to the solution of the whole sequence are compared. We start with the solution of each problem of the sequence as an independent one and proceed to use more information both about the iterative solution and the matrix $\mathbb{A}$ from the previous time steps.

The computations in this section are run on 3D mesh number 6 with properties given in Table 6.1 divided into 384 subdomains which is the same division that was used for the scaling tests in Section 6.3 with the final time $t_{fin} = 20000$ s. The final time is the same as in the convergence study in Section 5.1.1, however, because of a different mesh resolution and purpose of this study, a fixed time step $\Delta t = 10$ s is selected. Note that the computational times for the first time step in this section differ from those given in Section 6.3, because the computations were run on the `Helios` cluster while the scaling tests were run on the `Salomon` cluster.

6.4.1 Reference computation

The first approach to the solution of the sequence of linear problems resulting from the semi-implicit time discretization of the unsteady problem is to solve them as independent ones. In this approach at each time step, matrix $\mathbb{A}$ is assembled, its the preconditioner is computed and the BiCGStab iterations start from the zero initial vector. This is the same procedure that was used for the solution of the first time step problem in Section 6.3.

The computational times and the numbers of BiCGStab iterations at each time step for this approach are shown in Figure 6.3 and the overall computational time is shown in Table 6.15. The results in Figure 6.3 show that the time needed to set up the BDDC method varies only

slightly during all time steps (less than 3%). The BiCGStab method in most of the time steps converges in 13 iterations with a minimum of 12 and a maximum of 14 iterations. This small fluctuation in the number of iterations results in a slightly higher variance (less than 10%) in the computational times of the BiCGStab iterations.

In the following sections, we show how the performance of the method presented in this reference computation can be improved using the information from previous time steps. Both the iterative solution and the information about $\mathbb{A}$ from the previous time steps can be used and these approaches are discussed in Section 6.4.2 and Section 6.4.3, respectively.

The results of this reference computation of the sequence of independent problems justify the approach taken in Section 6.3 to demonstrate the performance of the BDDC solver. The results presented in Figure 6.3 show that the first time step is a good representation of the complete unsteady problem as the complexity of the linear problems arising during the computation varies only slightly.

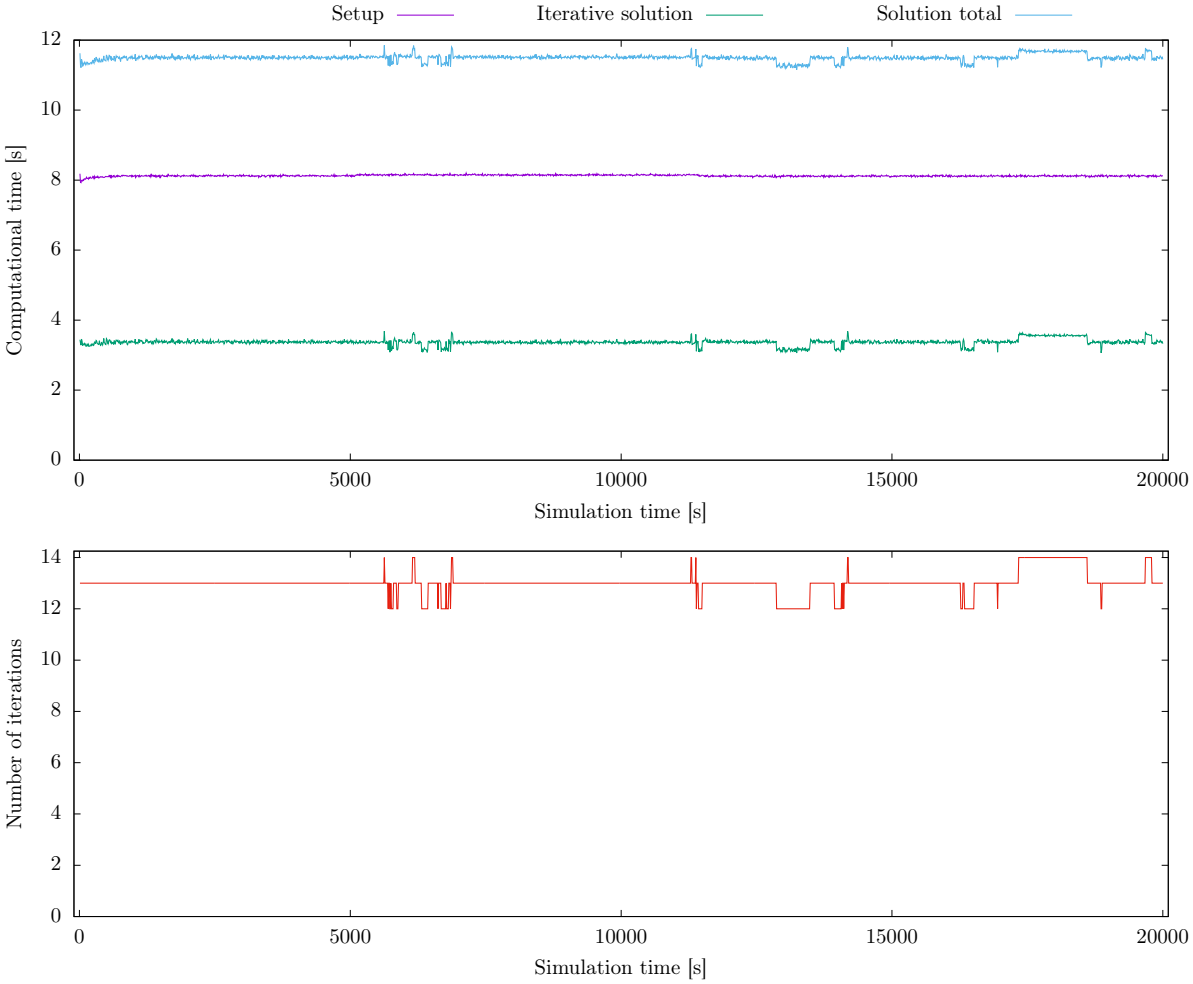


Figure 6.3: Computational times for setup, iterative solution, and overall solution (top), and number of BiCGStab iterations (bottom) for each time step ($\Delta t = 10\,\mathrm{s}$) of the unsteady problem. Iterations start from the zero initial vector and the preconditioner is computed in each time step.

6.4.2 Initial vector from previous time step

In this section, the solution from the previous time step is used as the initial vector for the BiCGStab iterations at the current time step in contrast with the reference computation in Section 6.4.1 where the BiCGStab iterations started from the zero initial vector. The only exception is the first time step where the zero initial vector is still used. We assume that the solutions of the McWhorter-Sunada problem in two consecutive time steps do not differ significantly and, therefore, the solution from the previous time step could be a good initial guess of the solution at the current time step.

Results for this approach are shown in Figure 6.4. The number of the BiCGStab iterations at all the time steps after the first one is significantly reduced in comparison with the case of BiCGStab iterations starting from the zero initial vector shown in Figure 6.3. The summary of results in Table 6.15 shows that the overall number of iterations is reduced by more than 50 %, however, the reduction in the computational time is only about 15 %. This disparity follows from the properties of the BDDC method for this problem, where most of the computational time is spent to set up the BDDC method as indicated by the detailed overview of computational times in Section 6.3.

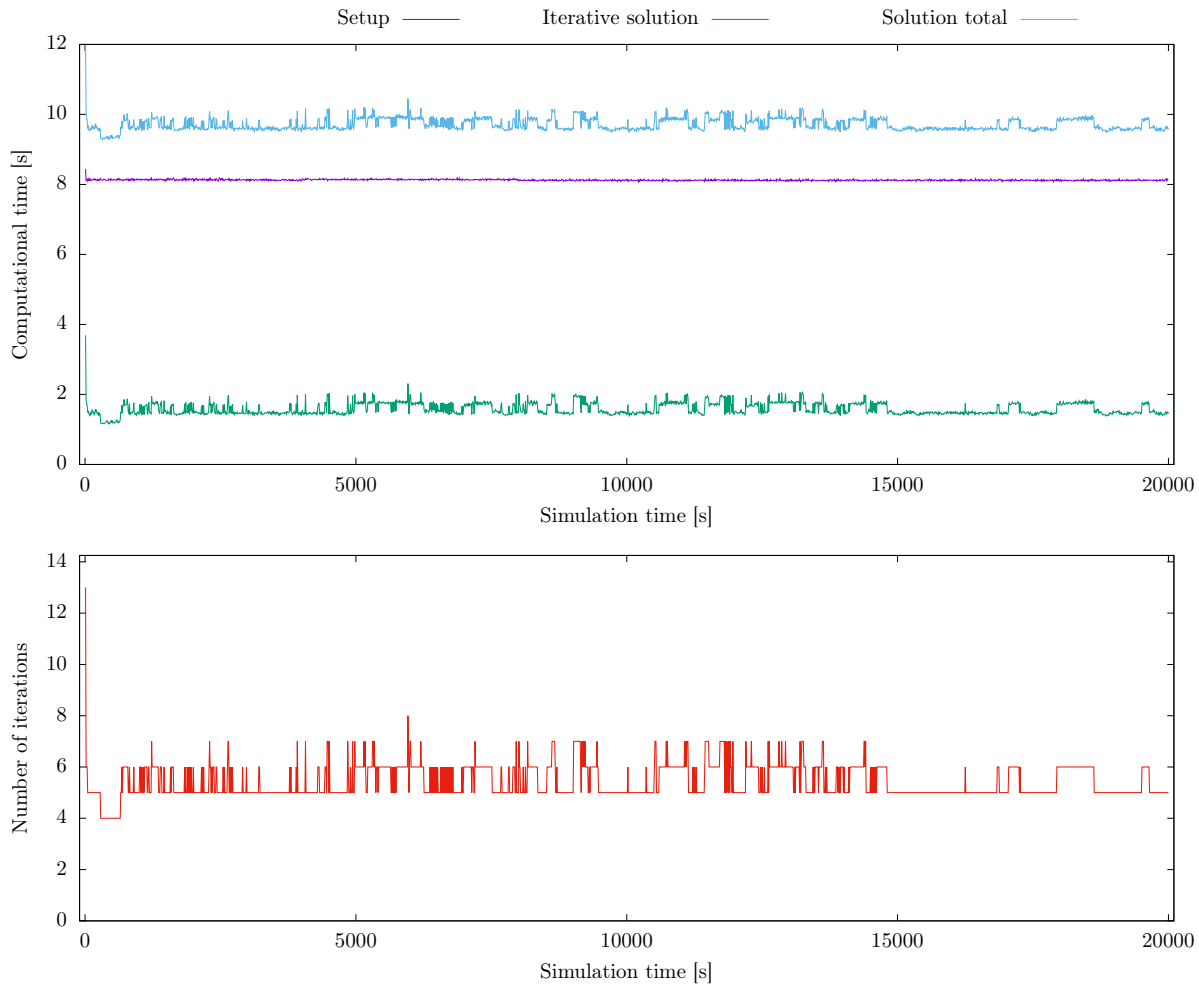


Figure 6.4: Computational times for setup, iterative solution, and overall solution (top), and number of BiCGStab iterations (bottom) for each time step ($\Delta t = 10\,\mathrm{s}$) of the unsteady problem. Iterations start from the solution from the previous time level and the preconditioner is computed in each time step.

6.4.3 Reusing the preconditioner

As shown in Section 6.4.2, using the solution from the previous time step as the initial vector reduces the number of iterations, however, the iterations are not the most time-consuming operation of the solution. To achieve a more significant reduction in the computation time, we focus on the setup of the BDDC method which takes larger portion of the overall computational time.

The steps needed to set up the BDDC method are described in Section 4.5.1. The factorization of $\mathbb{A}_i^{II}$ at each time step cannot be eliminated as the action of the Schur complement matrix needs to be computed exactly, however, we can avoid computing the preconditioner at each time step. Under the assumption of matrices $\mathbb{A}$ not changing significantly between the consecutive time steps, the same preconditioner can be used for several time steps without a significant reduction in preconditioner quality, instead of computing it at each time step. The quality of the preconditioner is evaluated using simple heuristics based on the number of BiCGStab iterations required at the given time step. The preconditioner is used until the number of iterations exceeds a given trigger value. The trigger values of 5, 10, 15, 20, and 25 are tested and the results are compared.

The overall computational times, the total number of BiCGStab iterations, and the number of preconditioner computations in Table 6.15 show that reusing the preconditioner can for certain trigger values significantly reduce the overall computational time. Although the lowest number of BiCGStab iterations is reached for the preconditioner computed at each time step, the shortest overall computational time is reached for the trigger value of 10, reducing the time of the reference computation by more than 40 %. For this trigger value, only 68 preconditioner computations are needed in comparison with 2000 for the reference computation in Section 6.4.1. Savings on the preconditioner computations are larger than the additional time spent on approximately 50 % more BiCGStab iterations required. However, for higher trigger values, the overall computational time starts to increase. For the trigger value of 25, the time needed for the additional BiCGStab iterations overweights the savings of only seven preconditioner computations and the computational time is even longer than for the case of the preconditioner updated in each time step.

In Figure 6.5, the detailed results for the trigger value of 10 show that the time needed to set up the method at the time steps when the preconditioner is updated is significantly larger than at the time steps when the already computed preconditioner is reused. The factorization of matrices $\mathbb{A}_i^{II}$ still needs to be done during the setup of the BDDC method at each time step, therefore, the time needed for the setup is never reduced to zero. The numbers of iterations in Figure 6.5 show a drop to a value of 4 up to time $t = 650\,\mathrm{s}$. This drop, observed only at the beginning of the computation, can be attributed to the fact that the head of the solution in this time interval is located only within the interior of a single subdomain and the interface degrees of freedom are almost unaffected.

6.5 Chapter summary

The weak and strong scaling tests presented in this chapter show that for both the standard problem with a symmetric matrix and also the more complex problem with a non-symmetric matrix the parallel implementation preserves good efficiency for division up to 768 domains for both 2D and 3D problems on structured and unstructured meshes. The efficiency of the BDDC method then deteriorates when the number of subdomains is increased to 1536. The drop in efficiency is more significant in 3D. The scaling tests for the symmetric matrices tend to confirm the previous results reported e.g. in [107] that the efficient application of the two-level BDDC

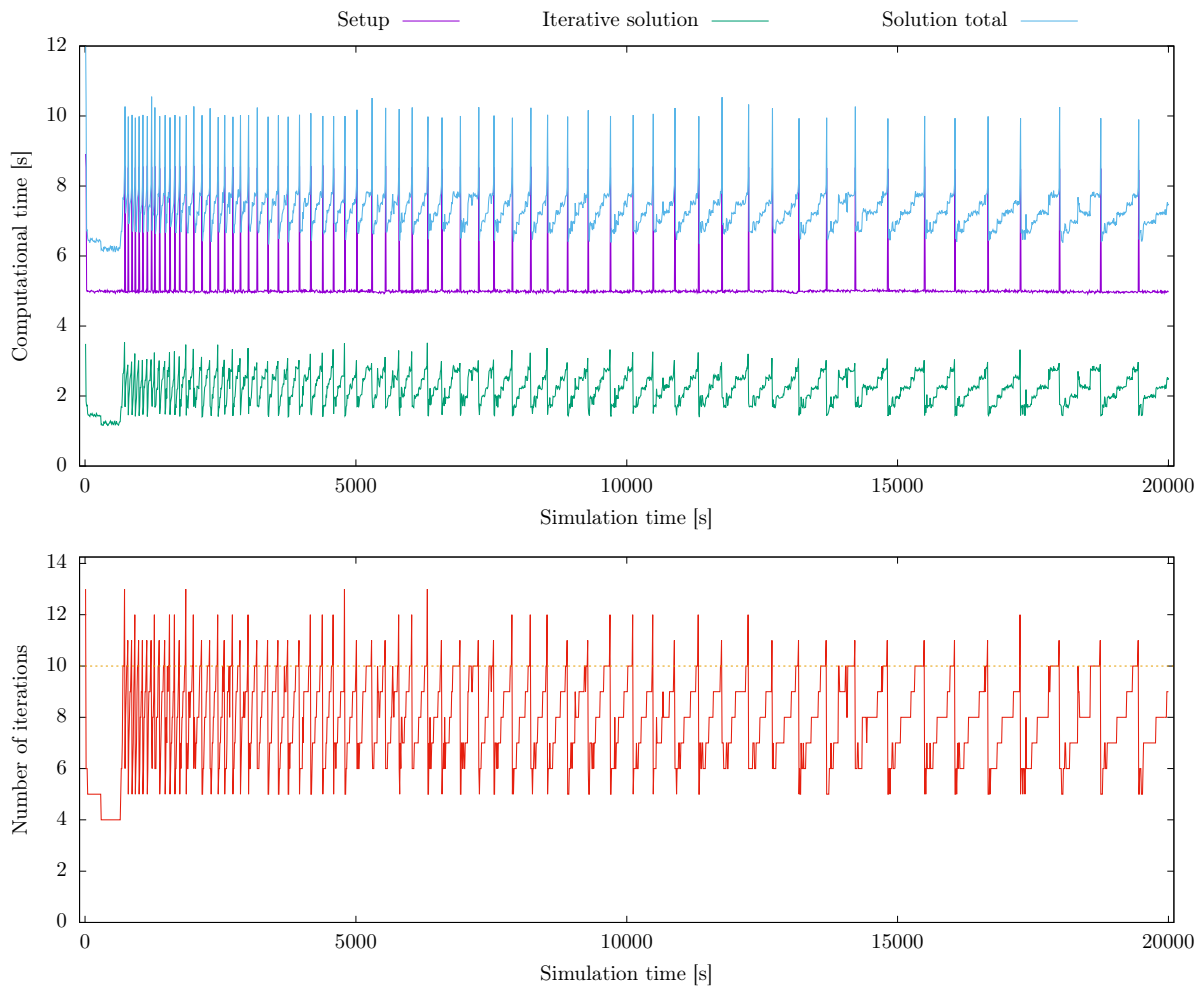


Figure 6.5: Computational times for setup, iterative solution, and overall solution (top), and number of BiCGStab iterations (bottom) for each time step ($\Delta t = 10\,\mathrm{s}$) of the unsteady problem. Iterations starting from the previous time level and with preconditioner used until the number of BiCGStab iterations exceeds 10 (dashed line).

method is limited to approximately a thousand subdomains and the performance significantly worsens with increasing the number of subdomains further. The results for the non-symmetric matrices show that the performance of the BDDC method for the non-symmetric problem is comparable to the performance on standard problems with a symmetric matrix.

For all the problems considered here, the superlinear speed-up is observed in the strong scaling test for the first few increases in the number of subdomains. Obtaining the speed-up higher than the optimal value can be attributed to the nonlinear complexity of the direct solvers used for the local problems together with the fact that the local problems are the most computationally demanding part of the problem for the lower number of subdomains.

For the same number of degrees of freedom, the computations in 2D are significantly faster than in 3D. The difference between 2D and 3D problems is caused by the different matrix structures for the 2D and 3D problems and also by more non-zero entries in each matrix row in the case of 3D problems. The non-symmetry of the matrices adds additional complexity. The LU factorization needs to be used instead of the faster Cholesky one for both local and coarse problems and also adjoint coarse basis functions are needed. The selected benchmark problem resulting in a non-symmetric matrix also arises from the system of two equations and, therefore,

Method	Computational time [min]	#BiCGStab iterations	#Preconditioner computations
Zero initial vector	387	26 006	2 000
Previous time step as initial vector	327	**10 848**	2 000
Reused preconditioner (5)	293	11 340	1 164
Reused preconditioner (10)	**247**	15 737	68
Reused preconditioner (15)	283	21 856	29
Reused preconditioner (20)	293	27 443	16
Reused preconditioner (25)	329	36 609	**7**

Table 6.15: Overall computational times, total number of iterations, and number of preconditioner computations for the McWhorter-Sunada problem in 3D on mesh 6 for time $t \in [0, 20000]$ seconds and various approaches to the solution of an unsteady problem with time step $\Delta t = 10\,\mathrm{s}$.

leads to twice as many non-zero entries in each matrix row. As a result, for the same number of degrees of freedom, the computations for this non-symmetric case are about three times slower than for the symmetric one.

In addition to the scaling tests on a single matrix demonstrating the properties of the BDDC method we also presented the application of the BDDC method to the sequence of linear problems arising from the spatio-temporal discretization of an unsteady problem of two-phase flow in porous media. In the reference approach of the problems solved as independent ones, the complexity of the problem varies only slightly: the computational time and the number of iterations remain almost the same for all the problems of the sequence. The computational time of this simple approach can be significantly reduced using the information from the previous time steps.

The total number of iterations is reduced by more than 50% using the solution from the previous time step as the initial vector for the BiCGStab iterations. However, a more significant reduction in the computational time is achieved by using the same preconditioner for several consecutive time steps, until the number of iterations exceeds a given trigger value instead of computing it at each time step. Although it leads to an increase in the total number of BiCGStab iterations, the savings of not computing the preconditioner at each time step overweight the losses caused by additional BiCGStab iterations. These two approaches combined result in up to 40% reduction of the computational time for the trigger value of 10 iterations for the selected benchmark problem.

Results from this chapter were published in [104].

Effects of mass transfer on attenuation of leaked carbon dioxide in shallow aquifers

7

In the case of leakage of the sequestered CO_2 from a deep aquifer, it is important to understand the processes controlling the transport of CO_2 to the surface. Due to the scale of the problem and significantly different conditions between the deep aquifer and the shallow subsurface, this complex problem is usually divided into simpler local ones. This division allows for focusing on certain processes in more detail than would be possible for the complete problem.

In this chapter, we focus on the shallow subsurface and study the transport and phase transitions of CO_2. Under the conditions in shallow aquifers, the originally supercritical CO_2 injected into the deep aquifer can reach the shallow zone dissolved in water. As the hydrostatic pressure decreases, the solubility limit of CO_2 in water also decreases and the CO_2 originally dissolved in water can evolve as an independent gaseous phase. Gas can accumulate under low-permeable layers forming temporal reservoirs [63] and when the water flow regime changes and clean water enters this reservoir, the entrapped gas can dissolve back into water and is transported away from the accumulation zone.

The processes related to the multiphase CO_2 evolution were thoroughly studied in recent years. However, there is still a gap in understanding of these processes in particular in the interactions between water and gas within the pores. This work is aimed to help to fill that gap using the data from highly controlled laboratory experiments and confront them with the results of a numerical model which employs various conceptualizations of mass transfer.

The experiments in this chapter are described in chronological order to explain the limitations of earlier experiments and the motivation for more complex experiments.

1. Experiment A, in detail described in Section 7.2, was carried out in a rather small 2D tank with a two-dimensional flow field. Only three saturation sensors were installed in the tank and the distances between the injection port and the sensors were small to observe the dynamic of exsolution and dissolution processes. These limitations lead to the need to carry out larger-scale and more complex experiments.

2. Experiment B, in detail described in Section 7.3, consisted of a high but narrow rectangular column. The height of the column allowed for the installation of more saturation sensors and larger distances between the injection port and the sensors to better study the dynamics of exsolution and dissolution processes. However, due to the dimensions of the tank, the flow field was predominantly one-dimensional in a vertical direction, which is not the case

of the flow expected in the field in the shallow subsurface. Based on the flow field, this experiment is also referred to as the quasi-1D case.

3. Experiment C, in detail described in Section 7.4, consisted of a large 2D tank. In contrast to the column in Experiment B, the dimensions of the tank and position of low permeable layers allowed for a complex two-dimensional flow field which is a more accurate representation of the flow regime in the field. Moreover, the distances between the injection port and saturation sensors were larger than in Experiment A and more sensors were used allowing for a more detailed study of the dynamics of the exsolution and dissolution processes under these flow conditions.

All three experiments consisted of the following three main stages:

1. Once the stationary flow field was established in the heterogeneously packed sand tank, the injection of CO_2-saturated water started through a selected port. The concentration of dissolved gas was higher than the saturated concentration in the tank allowing for the evolution of the gas phase within the tank.

2. The injection of CO_2-saturated water stopped and the evolution of both dissolved and gaseous CO_2 was observed. Emphasis was given to the accumulation of gaseous CO_2 under low-permeable sand layers set up in the tank.

3. Clean water was kept flowing through the tank and the dissolution of accumulated CO_2 was observed. In some experiments, the additional injection of clean water was introduced.

7.1 Model formulation

We investigate a two-phase compositional system consisting of gaseous CO_2 and water with dissolved CO_2. For clarity and to emphasize that the non-wetting phase is gas and the wetting phase is liquid, we use the subscripts g and ℓ, respectively, and refer to the phases as gas and liquid instead of the general notation of the wetting and non-wetting phases introduced in Section 2.3. For the purpose of this study, the liquid phase consists of two components: pure water and dissolved CO_2, gaseous phase consists of a single component: CO_2. The physical properties of the fluids are given in Table 7.1.

Similarly to the benchmark problems in Chapter 5, the capillary pressure and the gas phase pressure are selected as the primary unknowns for the two-phase flow: $Z_1 = p_c$, $Z_2 = p_g$. In addition to these two unknowns, the mass fraction of CO_2 dissolved in water is selected as the third primary unknown in this compositional system: $Z_3 = X$. For simplicity, we omit the superscript κ and subscript ℓ in the equation for the component transport (2.41) as only one component is tracked in the wetting phase only.

The governing equations of this system are then represented by the following non-zero

coefficients in the general formulation (3.1)

$$\mathcal{N} = \begin{pmatrix} -\phi\rho_\ell \frac{dS_\ell}{dp_c} & 0 & 0 \\ -\phi\rho_g \frac{dS_\ell}{dp_c} & \phi S_g \frac{d\rho_g}{dp_g} & 0 \\ 0 & 0 & \phi S_\ell \rho_\ell \end{pmatrix}, \qquad \mathcal{U} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \rho_\ell \boldsymbol{u}_\ell \end{pmatrix}, \tag{7.1}$$

$$\mathcal{M} = \begin{pmatrix} \rho_\ell \frac{\lambda_\ell}{\lambda_t} \\ \rho_g \frac{\lambda_g}{\lambda_t} \\ \rho_\ell \end{pmatrix}, \qquad \mathcal{D} = \begin{pmatrix} \lambda_t \mathbb{K} & -\lambda_t \mathbb{K} & 0 \\ 0 & \lambda_t \mathbb{K} & 0 \\ 0 & 0 & \tau_\ell \Phi S_\ell D_\ell \mathbb{I} \end{pmatrix}, \tag{7.2}$$

$$\mathcal{W} = \begin{pmatrix} -\lambda_t \rho_\ell \mathbb{K} \boldsymbol{g} \\ \lambda_t \rho_g \mathbb{K} \boldsymbol{g} \\ \mathbf{0} \end{pmatrix}, \qquad \mathcal{F} = \begin{pmatrix} -f_\ell \\ f_g \\ f_X - X f_\ell \end{pmatrix}. \tag{7.3}$$

The dissolution and exsolution of CO_2 are represented by the kinetic mass transfer model that was in the general form given by (2.44), here for CO_2 the mass transfer is considered as

$$-f_g = f_\ell = f_X = k(C_s - X\rho_\ell), \tag{7.4}$$

where k is the effective mass transfer coefficient introduced in Section 2.5.1. C_s is the saturated concentration of CO_2 given by Henry's law computed using parameters given in Table 7.2, see Section 2.5.2 for details. Its dependence on temperature will be discussed for individual experiments in the corresponding sections. The current concentration of the dissolved CO_2 required by (2.44) is computed from primary unknown as $C = X\rho_\ell$. In this study, we do not use the empirical relations for coefficient k in terms of dimensionless quantities discussed in Section 2.5.1 as the goal of this work is not to validate the model with the selected relation. We consider the lumped mass transfer coefficient k as a parameter and demonstrate the impact of changes in this coefficient on the numerical results and compare it experimentally.

Two studied processes: gas exsolution and dissolution are represented by (7.4). In each mesh element, only one of these two processes can take place at a given time based on the sign of f_X, i.e., whether the current concentration is higher or lower than the saturated concentration C_s. In addition, we conceptualize that the mass transfer coefficient k can be generally different for the exsolution and dissolution processes and the coefficients are then denoted by k_{exs} and k_{dis}, respectively.

The case of $C_s = X\rho_\ell$, i.e., $f_X = 0$, corresponds to the equilibrium state. In the kinetic model, this state is reached after a certain time that is controlled by the mass transfer coefficient k. In the equilibrium model, this state is assumed to be reached immediately. As was proposed in Section 2.5.2, we represent the equilibrium model in the computations by selecting a high enough mass transfer coefficient k. The equilibrium and kinetic models for various mass transfer coefficients are compared for the experiments investigated in this chapter and the differences for various dimensions of the problem and flow field patterns are discussed.

7.1.1 Dispersion

The tensor $\mathbb{D}_\alpha^\kappa$ in (2.43) describes diffusion and dispersion that both control the transport of the dissolved component. However, for scenarios considered in this work, dispersion is not the driving mechanism of the studied processes. In addition, the experiments were not designed to investigate the impact of mechanical dispersion. Therefore, to avoid introducing an additional

Fluid	ρ $[\mathrm{kg\,m^{-3}}]$	μ $[\mathrm{Pa\,s^{-1}}]$	M $\left[\mathrm{kg\,mol^{-1}}\right]$
Water (H_2O)	997.78	$9.72 \cdot 10^{-4}$	$1.802 \cdot 10^{4}$
Carbon dioxide (CO_2)	1.98	$1.48 \cdot 10^{-5}$	$4.401 \cdot 10^{4}$

Table 7.1: Fluid properties for the experiments in Chapters 7 and 8.

Solute	Solvent	T_{ref} [K]	$K_{H,ref}$ $\left[\mathrm{mol\,m^3\,Pa^{-1}}\right]$	C_H [K]
Carbon dioxide (CO_2)	Water (H_2O)	298.15	$3.35 \cdot 10^{-4}$	2400

Table 7.2: Henry's law constants for the experiments in Chapter 7. Adapted from [99].

uncertainty, we decided to omit the dispersion in our model, i.e., $a_L = a_T = 0$ in (2.43) is set for all the computations in this chapter. The free molecular diffusion of CO_2 in water is $D = 1.92 \cdot 10^{-9}\ \mathrm{m^2\,s^{-1}}$.

7.1.2 Implementation remarks

The main objective of this chapter requires running many independent computations with different parameters for which, the serial implementation of the numerical scheme described in Chapter 4 is used. These computations were run on a PC equipped with Intel Core i7-5820K, 3.6 GHz processor, and 32 GB RAM. The only exception is the larger-scale experiment in Section 7.4 where the results obtained on a rather coarse mesh used for the parameter sensitivity study are compared to the computations on finer meshes where the parallel implementation described in Chapter 4 is employed. These computations were run on the `Helios` cluster at the Czech Technical University in Prague, Czech Republic. The computational nodes of `Helios` are equipped with two 16-core AMD EPYC 7281, 2.1GHz processors and 128 GB RAM. As the problems in heterogeneous porous media are solved in this chapter, the ML variant of the numerical method is employed. All computational meshes used in this chapter were generated by `Gmsh` [49].

7.2 Experiment A

We start with the first experiment of the three carried out to study the CO_2 evolution in a subsurface. The design of this experiment was presented in [87], however, the data from this paper are not used here. This study uses the data from the subsequent experiments carried out using the same sand tank with different settings described here in detail.

The main goal of the computational study is to find the optimal values of mass transfer coefficients for exsolution and dissolution k_{exs} and k_{dis}, respectively. Moreover, we want to demonstrate the impact of changes in the mass transfer coefficients on the results and show that when a certain value of coefficients is reached, increasing it over this value has a negligible effect. This can be viewed as the transition from kinetic to the local equilibrium model.

7.2.1 Experiment and model setup

The setup of Experiment A and the unstructured triangular mesh used in the computations are shown in Figure 7.1. The mesh properties are given in Table 7.3. The left- and right-hand sides of the tank are connected to constant head devices, their positions control the background flow through the tank. 3.81 cm wide layers in the vicinity of the connections to these constant head devices are filled with gravel (Granusil #8) to uniformly distribute the hydraulic head along the vertical boundary of the tank. The rest of the tank is packed with coarser (Granusil #20/30) and finer (Granusil #110/250) sands, the finer one forming a low-permeable zone in the upper part of the tank, allowing for a vertical flow along both sides. The properties of gravel and both types of Granusil sand are given in Table 7.4. Three gas saturation sensors (Ports A, B, and C) are installed below the fine sand block to measure the accumulation of gas and two CO_2-saturated water injection ports (Ports E, F) are installed in the lower part of the tank. Moreover, the temperature sensor was installed in port D, however, for Experiment A, the temperature fluctuations both during and between individual experiment runs were negligible. Therefore, the system is considered isothermal and the value of saturated concentration C_s is computed for the constant temperature of 25°C.

Two setups are selected for comparison with the numerical results in this section. In the first experiment run, the constant head devices on both sides were positioned at the same height. This configuration results in a negligible background flow through the tank and therefore, this setup is referred to as the static case. In this setup, the water saturated with CO_2 was injected into the tank through port E for 23.5 hours with the injection rate of $2\ \mathrm{ml\,min^{-1}}$. Then clean water was injected through the same port for 23 hours with the injection rate of $4\ \mathrm{ml\,min^{-1}}$.

In the second experiment run, the hydraulic head on the right-hand side of the tank was positioned $0.5\ \mathrm{cm}$ higher than on the left-hand side resulting in background flow from right to left. The water flow rate at the outflow was measured. These measured values were used to calibrate the boundary conditions in the model to match the outflow rate. In this setup, water saturated with CO_2 was injected into the tank through port F for 47.4 hours with the injection rate of $2\ \mathrm{ml\,min^{-1}}$. Then the clean water was injected through port E for 30 hours with the injection rate of $4\ \mathrm{ml\,min^{-1}}$.

In both cases, the injected water was saturated by CO_2 at the overpressure value of $10\,\mathrm{kPa}$.

Initial and boundary conditions

Initially, the tank was filled with pure water: $X = 0$, $p_c = p_d$, and $p_g = p_c + p_\ell$, where the hydrostatic profile for p_ℓ is prescribed. The following boundary conditions were used:

Γ_S, Γ_E, Γ_N — For both static and background flow cases, zero Neumann boundary conditions are prescribed for all the unknowns.

Γ_3, Γ_4, Γ_5, Γ_6 — $p_c = p_d$ and $p_g = p_c + p_\ell$ with hydrostatic profile for p_ℓ are prescribed. In the case of the background flow, the values of p_ℓ on the right-hand side (Γ_3, Γ_4) are increased by $5\ \mathrm{Pa}$ from the hydrostatic profile. This value was selected to match the measured outflow rate through the left-hand side outflow segments Γ_5 and Γ_6. When the part of the boundary serves as an inflow or outflow values $X = 0$ or $\nabla X \cdot \boldsymbol{n} = 0$ are prescribed for the dissolved gas mass fraction, respectively.

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt [s]
1	$5.55 \cdot 10^{-3}$	5 494	8 211	24 633	5

Table 7.3: Mesh properties for Experiment A described in Section 7.2.

Γ_1, Γ_2 When the port is not active, a zero Neumann boundary condition is prescribed for all the unknowns. When water is injected through the port, the Neumann boundary condition for $\boldsymbol{u}_\ell$ is prescribed to match the overall inflow rate through the port and $\boldsymbol{u}_g \cdot \boldsymbol{n} = 0$ is prescribed for gas. In the case of CO_2-saturated water, the Dirichlet boundary condition for X is prescribed. Its value is computed from the overpressure value of $10\,\mathrm{kPa}$ and temperature $25\,°\mathrm{C}$. In the case of clean water, $X = 0$ is prescribed.

All computations are run with a fixed time step $\Delta t = 5$ s.

The critical gas saturation introduced in Section 2.3.7 can play an important role in correctly capturing the behavior of the exsolved gas in the initially fully water-saturated porous media. However, the setup of Experiment A with all the gas saturation sensors installed directly below the low-permeable layer does not allow for investigating the impact of the critical gas saturation S_c on the results. All computations in this section were carried out with the value of $S_c = 0.2$ which is in agreement with the following experiment in Section 7.3 with more saturation sensors positioned further away from the low-permeable regions where the detailed discussion about finding the value of S_c is provided.

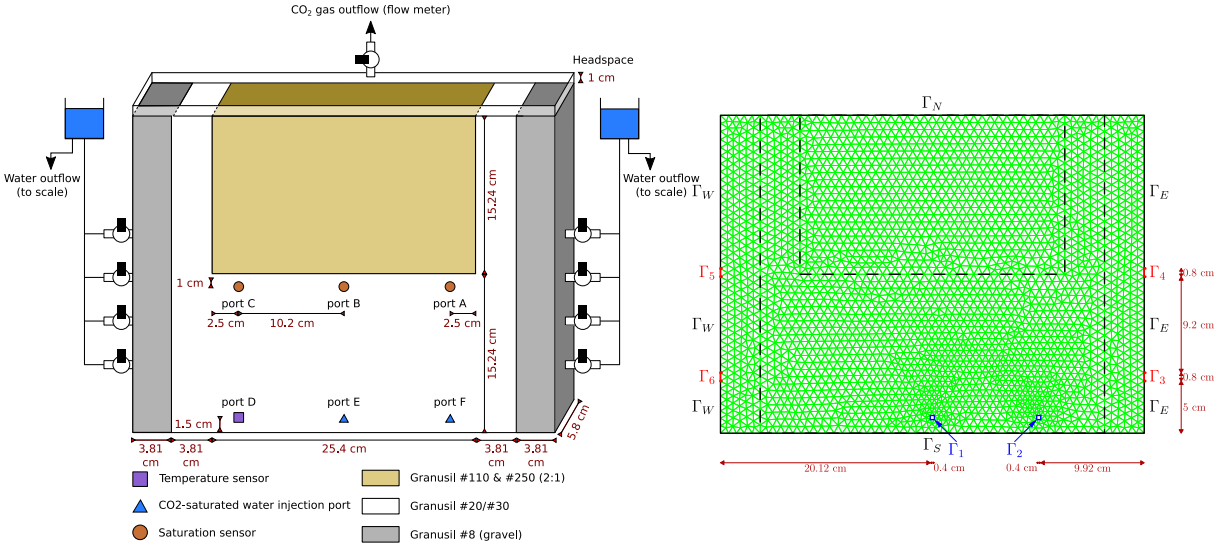


Figure 7.1: Setup of Experiment A described in Section 7.2 adapted from [87] and the computational domain. The properties of the numerical mesh are given in Table 7.3.

7.2.2 Discussion of results

The comparison of CO_2 saturation values S_g in Ports A, B, and C between the numerical results for the selected values of k and experimental data for the static and background flow experiments are shown in Figure 7.2. These numerical results for various values of k show that the best match with experimental data is obtained for rather large values of k in the range of 0.1 to 0.5 s^{-1}. In addition, the simulated results almost do not change when the value of the mass

Sand	ϕ [–]	K [m^2]	S_{rw} [–]	S_{rn} [–]	p_d [Pa]	λ^{BC} [–]
Granusil #8	0.41	1.00×10^{-9}	0.084	0	600	4.725
Granusil #20/30	0.32	2.30×10^{-10}	0.084	0	1200	7.33
Granusil #110/250	0.35	6.36×10^{-14}	0.170	0	8027	5.35

Table 7.4: Sand properties for Experiment A in Section 7.2. Adapted from [85, 88].

transfer coefficient exceeds a certain threshold value that corresponds to the best match with the experimental data. This indicates that the exsolution and dissolution processes are in the case of Experiment A fast enough that can be represented by the (near-) equilibrium approach and these findings are the same for both static and background flow cases. The comparison of the water velocity along the horizontal cross-section through the tank at the level of the injection port in Figure 7.5 shows that the flow fields in the tank are similar and the results of the study indicate that the flow regimes in this experiment do not significantly affect the CO_2 mass transfer rates.

The distribution of CO_2 saturation S_g and the mass fraction of dissolved CO_2 in water X in the whole tank for the value of $k = 0.5$ s^{-1} in selected times are shown in Figures 7.3 and 7.4 for the static and background flow experiments, respectively. The block of fine sand works as an obstacle to the flow and the gaseous CO_2 accumulates below this block, spreads laterally, and eventually, due to the buoyancy, rises to the top of the tank along the low-permeable region.

The comparison of measured data and numerical results indicates that the mass transfer in both these scenarios can be modeled using the equilibrium or near equilibrium approach. However, Experiment A provided only a limited number of measurements in the tank, all of them located very close to the heterogeneity. Therefore, we cannot draw a general conclusion using only this experiment and more complex experiments with more sensors located also in the regions of the tank further from the heterogeneity are needed. This limitation of Experiment A motivated to carry out Experiment B and Experiment C which are investigated in the following Sections 7.3 and 7.4, respectively.

7.3 Experiment B

In this section, a quasi-1D rectangular column packed with a layer of fine sand with dimensions and heterogeneity configuration leading to a predominantly vertical flow field is considered. The main goal of the study carried out for this column experiment is similar to Experiment A in Section 7.2: to find the optimal values of the mass transfer coefficients k_{exs}, k_{dis} for the exsolution and dissolution, respectively. In addition, we are also interested in the value of critical gas saturation S_c that in this experiment plays an important role.

7.3.1 Experiment and model setup

Six experiments were carried out using the same sand tank with the layer of sand mimicking the apex of an anticlinal geological feature. The shape of individual sand regions, positions at which the gas saturation was measured, and the numerical mesh are shown in Figure 7.6. Properties of the sands are summarized in Table 7.8 and the mesh parameters are listed in Table 7.7. The mesh was rather coarse with local refinement in the vicinity of the material interface. The spatial resolution of the mesh was selected to allow for many computations with various model

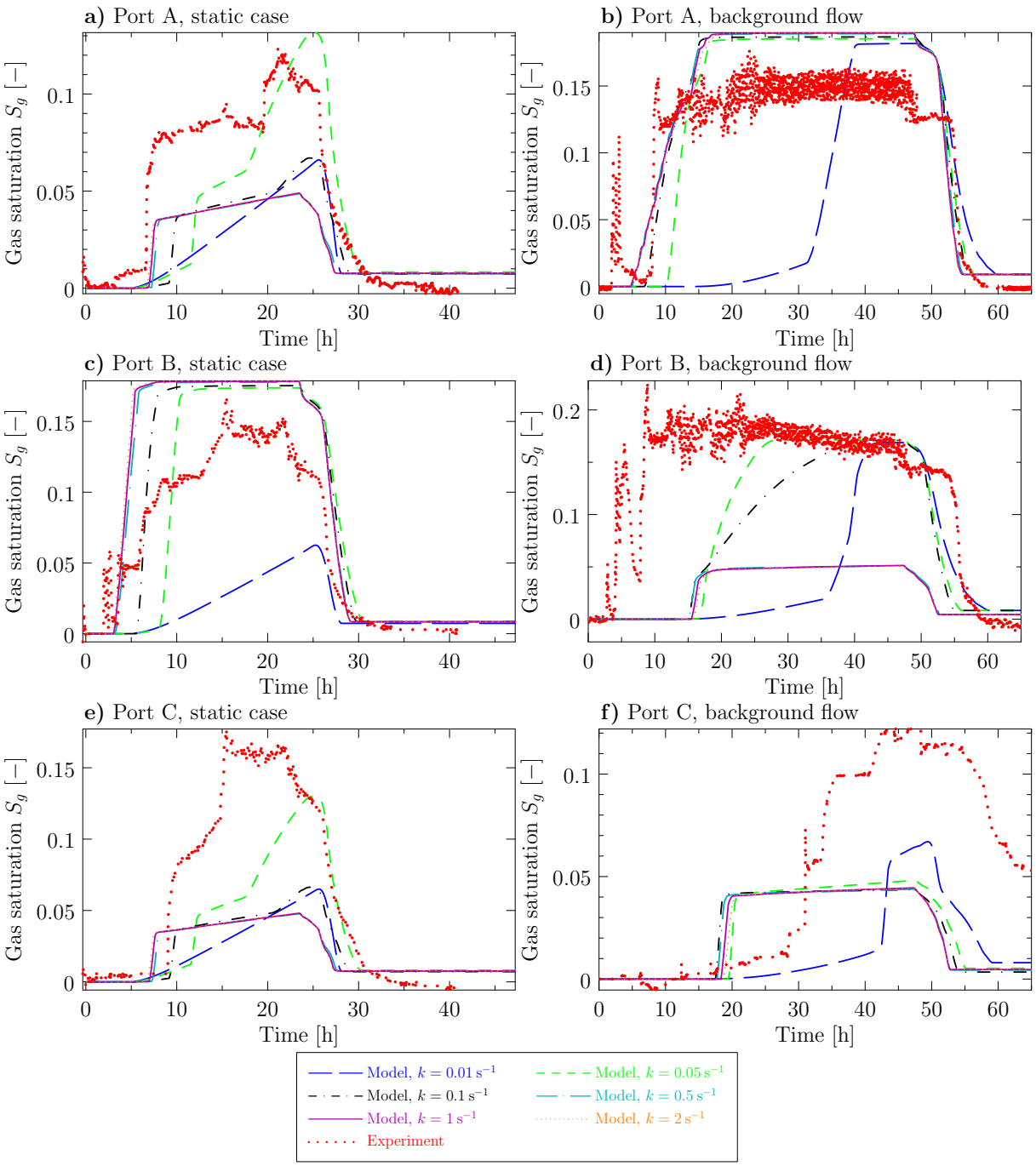


Figure 7.2: Gas saturation for Experiment A described in Section 7.2 for (a) the static case and (b) the case with a background flow.

parameters in a reasonable time, and at the same time, to capture the effects investigated in this setup. This configuration remained the same for all six column experiments.

Each column experiment consisted of CO_2-saturated water injection through the port located at the bottom of the tank for a given period of time with a constant injection rate. A portion of the dissolved CO_2 exsolved into the gas phase, migrated upwards, accumulated under the coarse-fine sand interface, and eventually penetrated the fine sand region. The injection of CO_2-saturated water was followed by the injection of clean water through the same port with

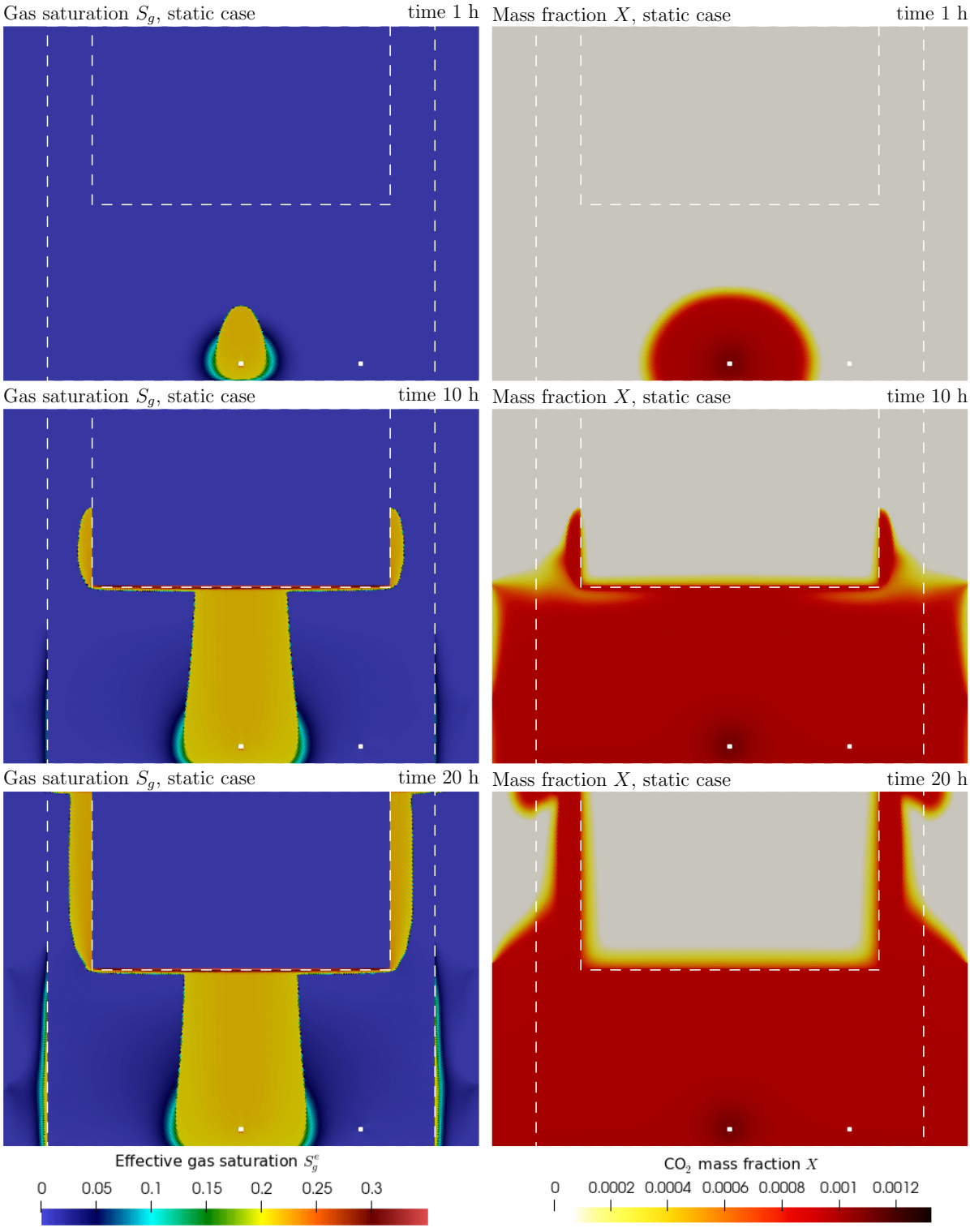


Figure 7.3: Spatial distribution of S_g (left) and X (right) for Experiment A described in Section 7.2: the static case at 1 h, 10 h, and 20 h (from top to bottom), computed using $k = 0.5\ \mathrm{s}^{-1}$ and $S_c = 0.2$.

varying flow rates. The gaseous CO_2 present in the tank dissolved into the water and was transported upwards. The experiment ended when no gas was present in the tank.

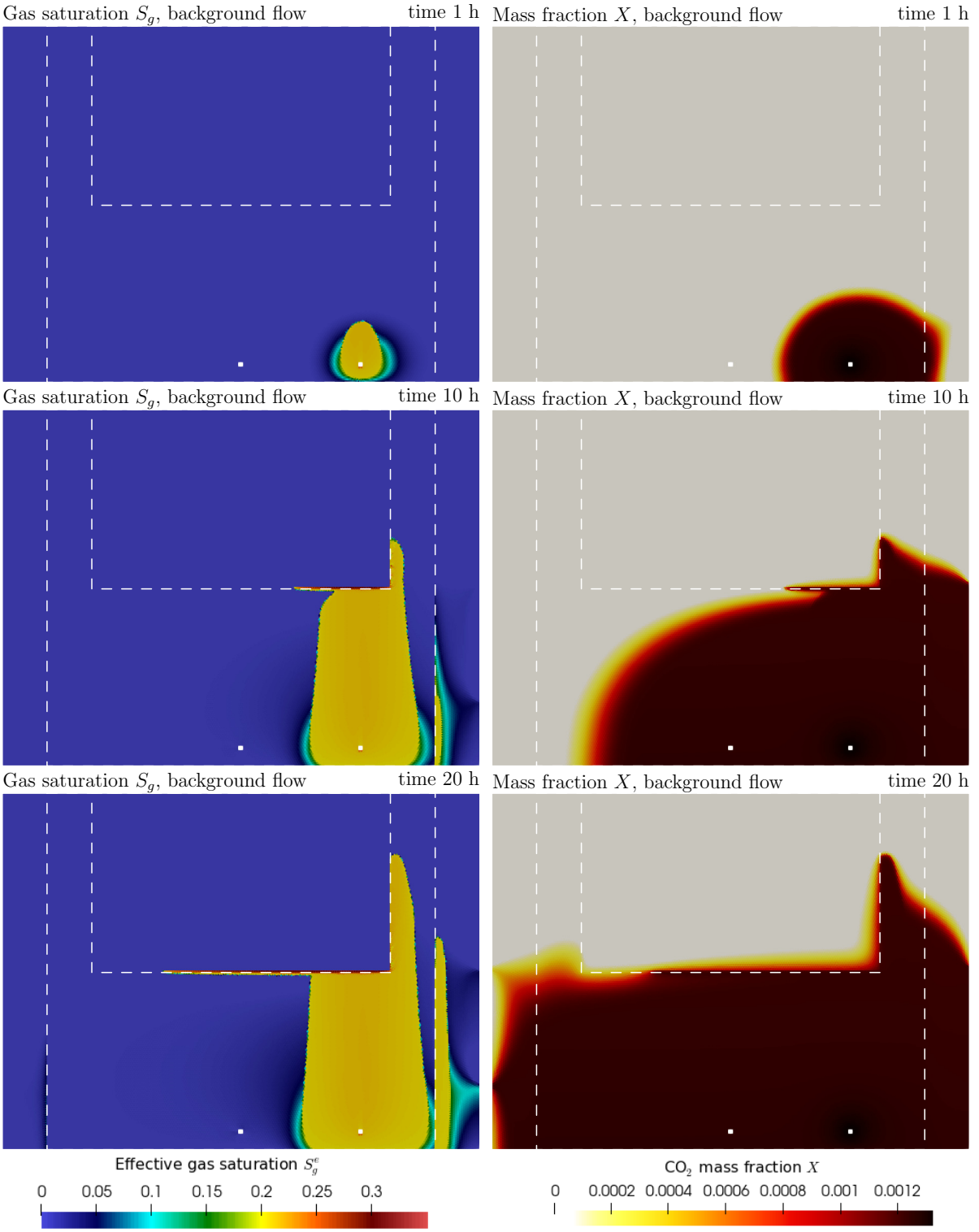
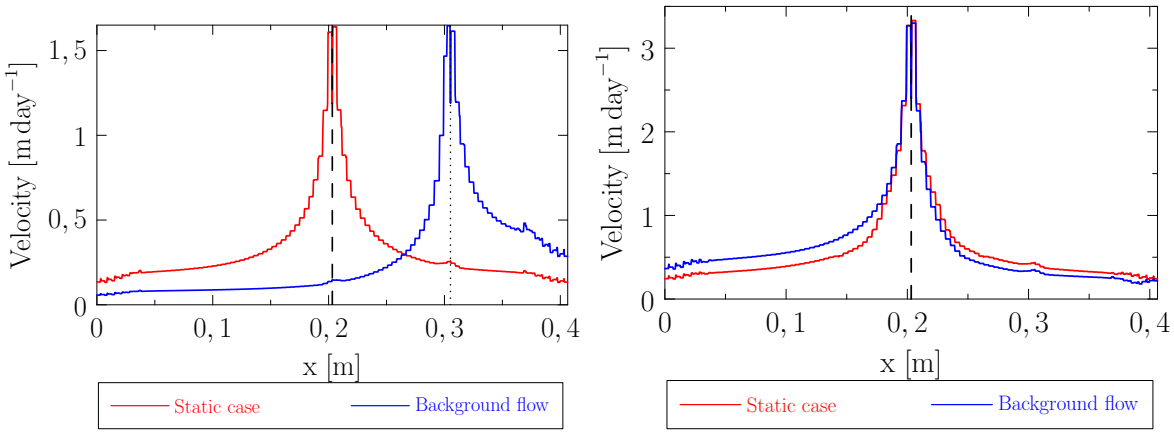


Figure 7.4: Spatial distribution of S_g (left) and X (right) for Experiment A described in Section 7.2: the background flow case at 1 h, 10 h, and 20 h (from top to bottom), computed using $k = 0.5\ \mathrm{s}^{-1}$ and $S_c = 0.2$.

The experiments varied in:

- the concentration of dissolved CO_2 characterized by the overpressure at which the water



(a) Start of the experiment.

(b) Start of the clean water injection.

Figure 7.5: Water velocity profile over the x-axis at the level of the injection port for Experiment A described in Section 7.2. The positions of the injection ports are depicted using the dashed and dotted lines.

was saturated by CO_2,

- the inflow rates of both CO_2-saturated and clean water,
- the length of injection periods for individual flow rates.

The two-letter notation of the experiments is introduced. The first letter characterizes the overpressure at which the water was saturated with CO_2: L = low (12 kPa), M = medium (20 kPa), and H = high (30 kPa). The second letter denotes the flow rate of the CO_2-saturated water in the first stage of the experiment: S = slow ($4\ \mathrm{ml\ min^{-1}}$) and F = fast ($40\ \mathrm{ml\ min^{-1}}$). The settings of these six experiments is summarized in Tables 7.5 and 7.6. Similarly to Experiment A, the temperature was monitored during the experiments. The readings show only very small fluctuations during all the experiments. Therefore, the constant temperature of $21\,^{\circ}\mathrm{C}$ was assumed for the whole Experiment B. The final time t_{fin} of each computation is selected the same as the final time of the corresponding experiment as listed in Table 7.5.

Initial and boundary conditions

The initial conditions are the same for all six experiments: $X = 0$, $p_c = p_d$, and $p_g = p_c + p_\ell$, where the hydrostatic pressure profile for p_ℓ was prescribed. These conditions represent the setup at the start of the experiment: the tank contains only pure water with no flow through the tank. The following boundary conditions were prescribed:

Γ_{in} During the CO_2-saturated water injection, the Neumann boundary condition for the water inflow velocity $\boldsymbol{u}_\ell$ and the Dirichlet boundary condition for the CO_2 mass fraction X computed from the values in Table 7.5 are prescribed. During the clean water injection, the Neumann boundary condition for the water inflow velocity $\boldsymbol{u}_\ell$ computed to correspond to the injection schedule given in Table 7.6 for each experiment together with the Dirichlet boundary condition $X = 0$ are prescribed. The one-minute breaks between the individual injections, i.e., rows in Table 7.6, needed to set up the next injection in the experiment, are represented by the no-flow Neumann boundary conditions for all the primary unknowns similarly to Γ_b. The injection port acts as a wall during these breaks.

Γ_w, Γ_b Zero Neumann boundary condition is prescribed for both phases as well as for the dissolved CO_2.

Γ_t The Dirichlet boundary conditions for both phases $p_c = p_d$, $p_g = p_{ref}$, and $\nabla X \cdot \boldsymbol{n} = 0$ for the dissolved gas are prescribed with $p_{ref} = 82$ kPa as the reference atmospheric pressure in the laboratory.

The computations were carried out with constant time steps: $\Delta t^L = 5$ s, $\Delta t^M = 2$ s, and $\Delta t^H = 1$ s in the case of low, medium, and high overpressure experiments, respectively.

Id.	Overpressure [kPa]	Inflow rate [ml min^{-1}]	Injection period [h]	Experiment duration [h]
B-LS	12	4	104.80	150
B-LF	12	40	10.67	70
B-MS	20	4	92.62	186
B-MF	20	40	10.45	73
B-HS	30	4	98.76	212
B-HF	30	40	10.20	76

Table 7.5: Settings of Experiment B described in Section 7.3. The material properties are listed in Table 7.8. The experiment duration in the last column is selected as the final time t_{fin} of the simulations.

For the purpose of this study, each column experiment was divided into three stages: exsolution, accumulation, and dissolution. This division into consecutive and almost independent stages allowed for splitting the problem of finding the values of the unknown model parameters k_{exs}, k_{dis}, and S_c into separate tasks. The stages are not completely isolated as the exsolution and accumulation both occur during the CO_2-saturated water injection and dissolution and exsolution processes can take place at the same time in different positions in the column. But, in each stage, the selected process is dominant and can be used to determine one of the model parameters as will be discussed in detail in the following text.

7.3.2 Stage 1: exsolution

During the exsolution stage, the CO_2-saturated water is injected into the tank, CO_2 exsolves, and the gas phase appears in the tank. The height at which the gas is detected differs between the experiments. For the medium and high overpressure experiments, the gaseous CO_2 appears at the bottom of the column and is detected in Ports 22 - 24. For the low overpressure experiments, the gaseous CO_2 first appears approximately in the middle of the column height close to the heterogeneity.

When the gas is detected, the gas saturation grows and, after a certain period of time, the measured saturation values start to oscillate around a constant value in the ports at the bottom of the tank as shown in Figure 7.7. These fluctuations that are more significant for the slow experiment runs are caused by the changes in the gas distribution at the pore scale and for the slow experiment runs, there is more time for this diffusive redistribution to occur.

Id.	B-LS	B-LF	B-MS	B-MF	B-HS	B-HF
duration [h]	12.15	12.44	2.80	14.34	2.87	15.70
inflow rate [ml min^{-1}]	6	8	32	6	32	6
duration [h]	2.53	2.58	16.83	2.35	14.53	3.03
inflow rate [ml min^{-1}]	32	32	6	32	8	32
duration [h]	2.80	3.28	2.67	2.62	2.7	2.98
inflow rate [ml min^{-1}]	32	32	32	32	32	32
duration [h]	3.15	2.25	3.07	2.95	2.67	3.22
inflow rate [ml min^{-1}]	32	32	32	32	32	32
duration [h]	15.57	14.91	2.3	16.67	2.78	15.67
inflow rate [ml min^{-1}]	6	6	32	6	32	6
duration [h]	2.98	2.93	15.95	3.17	12.68	2.6
inflow rate [ml min^{-1}]	32	32	6	32	8	32
duration [h]	5.60	2.45	3.17	3.65	3.93	2.57
inflow rate [ml min^{-1}]	16	32	32	32	32	32
duration [h]	-	2.92	3.02	16.47	2.73	1.7
inflow rate [ml min^{-1}]	-	32	32	6	32	32
duration [h]	-	14.6	1.82	-	3.30	2.87
inflow rate [ml min^{-1}]	-	6	32	-	32	32
duration [h]	-	-	15.28	-	14.733	12.183
inflow rate [ml min^{-1}]	-	-	6	-	6	8
duration [h]	-	-	3.28	-	4.40	2.05
inflow rate [ml min^{-1}]	-	-	32	-	16	32
duration [h]	-	-	3.30	-	44.92	-
inflow rate [ml min^{-1}]	-	-	32	-	2	-
duration [h]	-	-	1.17	-	-	-
inflow rate [ml min^{-1}]	-	-	32	-	-	-
duration [h]	-	-	17.22	-	-	-
inflow rate [ml min^{-1}]	-	-	6	-	-	-

Table 7.6: Clean water injection schedule for Experiment B described in Section 7.3.

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt^L [s]	Δt^M [s]	Δt^H [s]
1	$3.02 \cdot 10^{-2}$	2 552	3 915	11 745	5	2	1

Table 7.7: Mesh properties for Experiment B described in Section 7.3.

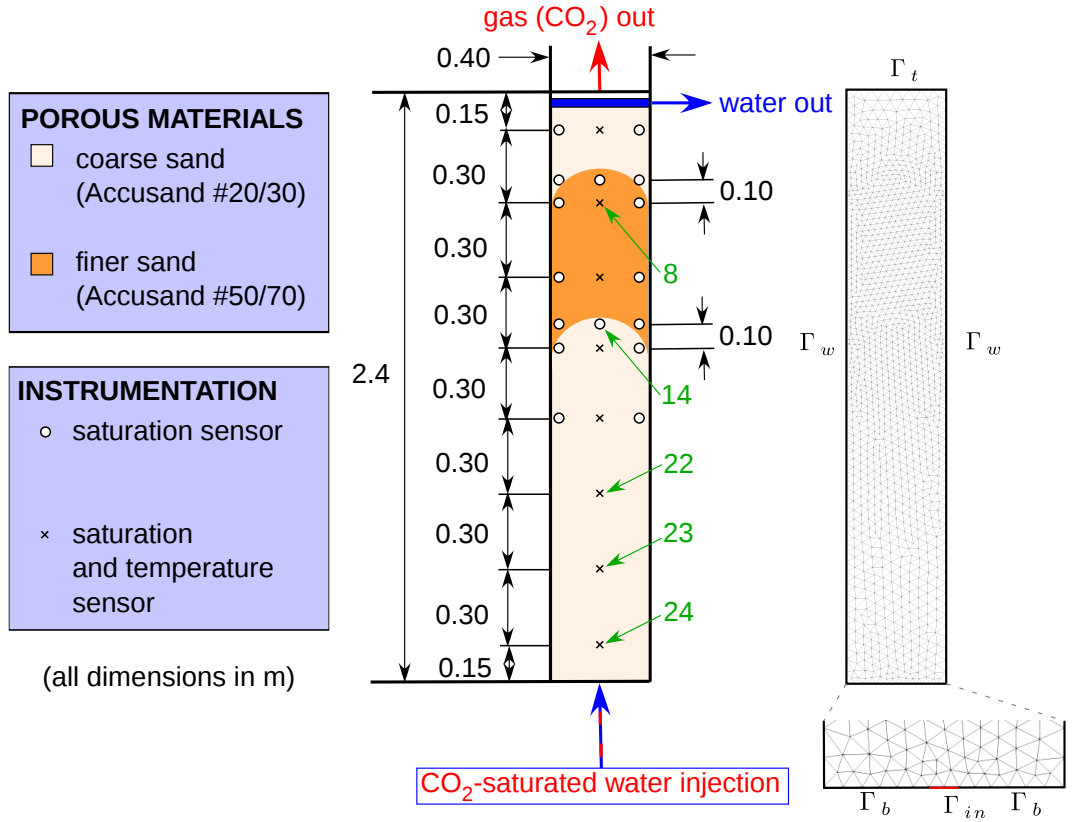


Figure 7.6: Configuration of Experiment B described in Section 7.3 and the triangular mesh used in the numerical simulations with parameters given in Table 7.7.

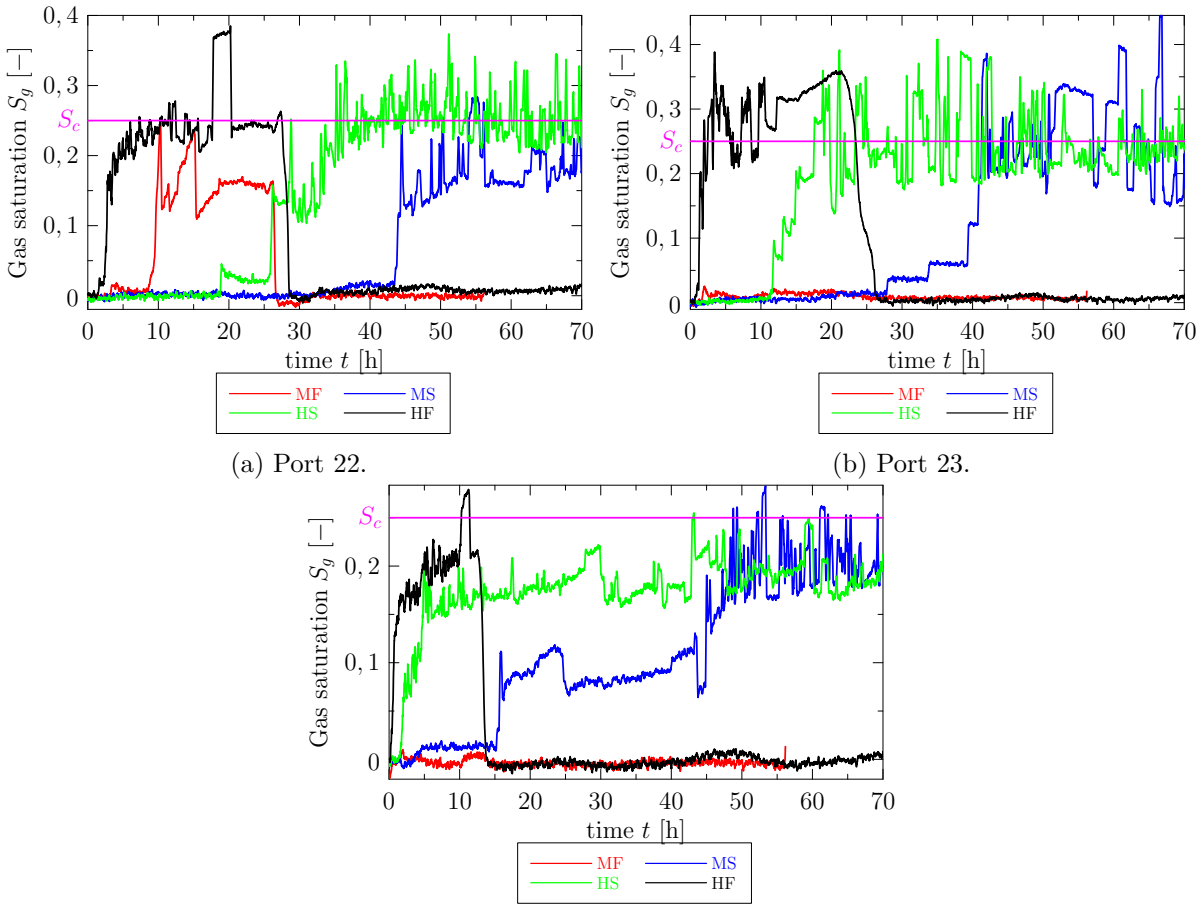
Sand	ϕ [−]	K [m^2]	S_{rw} [−]	S_{rn} [−]	p_d [Pa]	λ^{BC} [−]
Accusand #20/30	0.32	$2.3 \cdot 10^{-10}$	0.084	0	1200	7.33
Accusand #50/70	0.34	$3.0 \cdot 10^{-11}$	0.207	0	3400	16.9

Table 7.8: Sand properties for Experiment B in Section 7.3. Adapted from [85, 88].

These plateau values around which the measured values fluctuate represent the fraction of the gas phase that remains immobile in the porous media and can be directly interpreted as the value of critical saturation S_c. The measured values for the medium and high overpressure experiments are consistent and the value of the critical gas saturation is estimated to be $S_c = 0.25$. This value is also in agreement with the value reported in [90].

The impact of the value of S_c on the numerical results for experiment HS is demonstrated in Figure 7.8, where the results of the computations for the critical gas saturation values 0.05, 0.1, 0.15, 0.2, 0.25, 0.3 are compared. This comparison indicates that the value $S_c = 0.25$ selected based on the measured data is a sufficiently good approximation of the experimental data.

Note that the procedure to obtain the value of the critical gas saturation directly from the experimental data described above requires the non-zero gas saturation detected in the homogeneous region where the amount of trapped gas is controlled by its value. Therefore, it cannot be used for the low overpressure experiments B-LF and B-LS because, in these runs, CO_2



(a) Port 22.

(b) Port 23.

(c) Port 24.

Figure 7.7: Medium and high overpressure Experiment B described in Section 7.3, measured gas saturation and highlighted value of critical gas saturation $S_c = 0.25$.

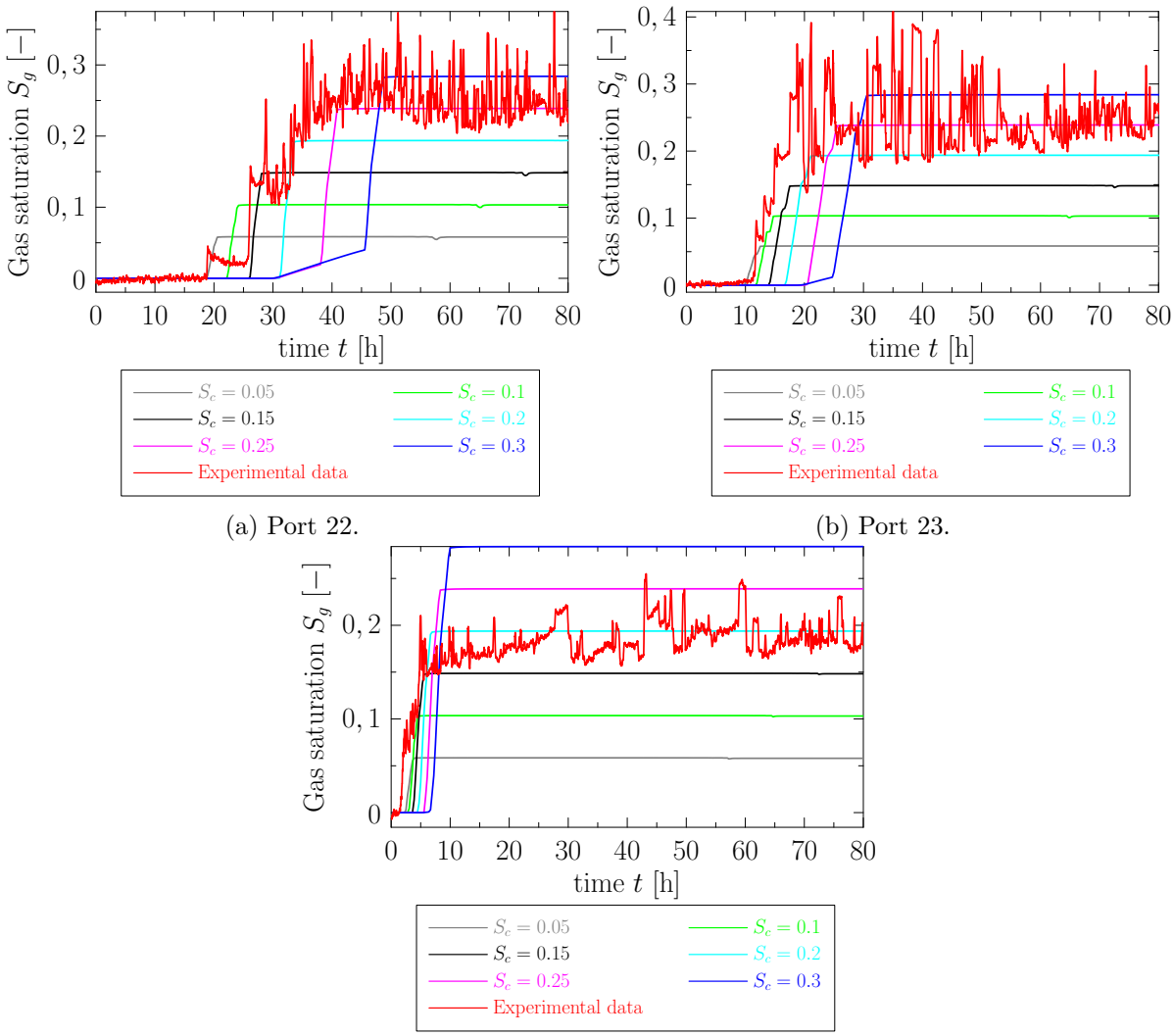
exsolves in the middle section of the tank close to the layer of the fine sand where the capillary barrier effect is responsible for the gas trapping and the gas saturation values much higher than 0.25 are detected.

7.3.3 Stage 2: accumulation

The accumulation stage of the experiments partially overlaps with the exsolution one. The injection of the CO_2-saturated water continues and gaseous CO_2 that exceeds the value of S_c migrates upwards and accumulates below the layer of fine sand that acts as a capillary barrier. As a result of differences in the entry pressures, the gas saturation reaches up to almost 0.9 which is the highest value detected in Port 14 directly below the fine sand layer. Due to high gas saturation values reached in Port 14, readings from this port are selected to demonstrate the dynamics of the CO_2 exsolution and accumulation processes in this section.

For each of the six experiment runs, a sensitivity study was carried out to determine the value of k_{exs} to match the experimental data. As the goal was not only to find the optimal value of k_{exs} but also to demonstrate the exsolution dynamics for a wide range of mass transfer coefficients, the computations were run for a series of values of k_{exs} with a selected step of 0.01 [s^{-1}].

From these computations, the optimal values of k_{exs} were selected to best fit the measured values of gas saturation S_g in all the ports during the accumulation stage. This best fit to the



(a) Port 22.

(b) Port 23.

(c) Port 24.

Figure 7.8: Experiment B-HS described in Section 7.3, gas saturation S_g for various values of critical gas saturation and for large mass transfer rate coefficients $k_{exs} = k_{dis} = 5\ \mathrm{s}^{-1}$.

experimental data is considered in terms of the sum of Euclidean norms of the difference between the computed and the measured value of S_g for all the ports.

The results of this procedure indicate that rather large values of k_{exs} are needed to result in enough dissolved CO_2 to exsolve in the tank to match the experimental data. In addition, a certain threshold value of k_{exs} denoted by k^*_{exs} exists for which the amount of exsolved CO_2 does not change when the k_{exs} is increased over this threshold, i.e., the numerical results are the same for k^*_{exs} and for all $k_{exs} > k^*_{exs}$. For the HS experiment, this behavior is illustrated in Figure 7.9 and similar effects can be observed also in the remaining column experiments for the same threshold value of $k^*_{exs} = 5\ \mathrm{s}^{-1}$. These findings on the exsolution dynamics indicate that a near-equilibrium rather than a rate-limited mass transfer is observed in the exsolution stage of all column experiments.

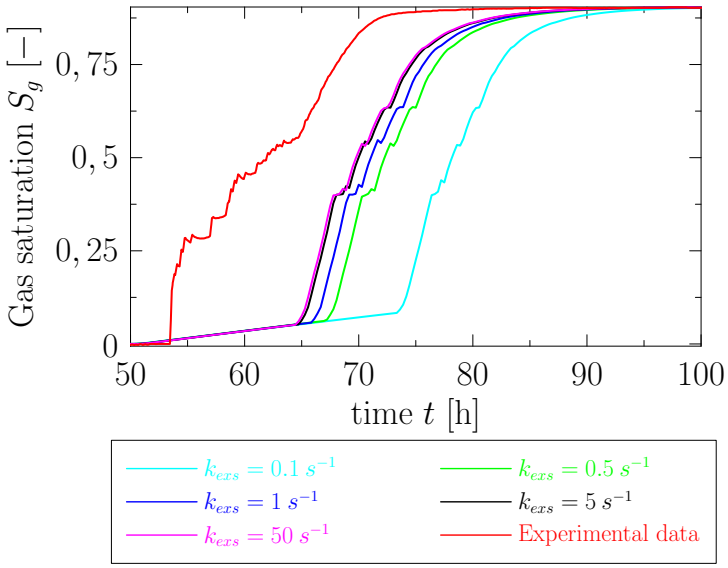


Figure 7.9: Experiment B-HS described in Section 7.3, gas saturation in Port 14 for various values of mass transfer coefficient for exsolution k_{exs}.

7.3.4 Stage 3: dissolution

The CO_2-saturated water injection is followed by clean water injection in the dissolution stage of the experiment. During this injection (with the schedule for each experiment given in Table 7.6) the gaseous CO_2 present in the tank dissolves into the clean water and is transported upwards. The dynamics of the dissolution process observed on the measured values of the gas saturation S_g in the ports show that the process is, similarly to the exsolution, rapid. The analogous parameter sensitivity study for the mass transfer coefficient for dissolution k_{dis} was carried out to find its optimal values. The only difference from the study presented in Section 7.3.3 was that the time period of the clean water injection was considered for the comparison between the measured and computed values of S_g.

The findings of this study are similar to the exsolution one. A substantially large values of $k_{dis} > k^*_{dis}$ with $k^*_{dis} = 5\ \mathrm{s}^{-1}$ as the threshold value are needed to capture the dissolution rate observed in the experiments. For the B-HS experiment, the dynamics of the dissolution process for various values of k_{dis} (both higher and lower than the threshold value k^*_{dis}) is illustrated in Figure 7.10 using Port 14. The same behavior is again observed for all the column experiments indicating that the CO_2 dissolution can be also considered as the near-equilibrium process.

7.3.5 Discussion of results

The best match with the experimental data was obtained for the the following parameters: $S_c = 0.25$, $k_{exs} = k_{dis} = 5\ \mathrm{s}^{-1}$. The overall comparison between the experimental data and numerical results for all six column experiments are shown in Figure 7.11 - 7.13.

After the calibration to match the measured gas saturation data in all the ports, the model can be used to obtain more information about the multiphase CO_2 evolution both by comparing the numerical results to the experimental data but also by comparing the numerical results for the different setups listed in Table 7.5.

First, in Figure 7.14 we compare the gas saturation profiles along the vertical axis of the tank at the end of the injection period of each experiment. The shapes of the saturation profiles indicate that the overpressure at which the water was saturated by CO_2 is the main controlling

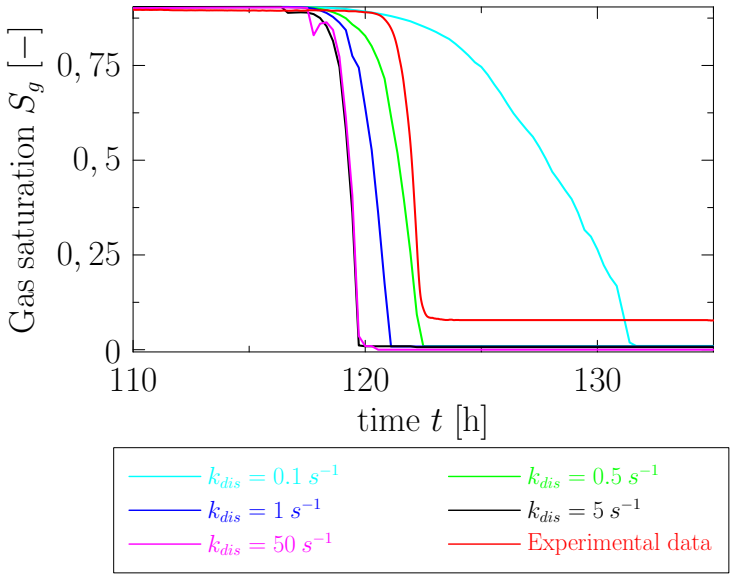


Figure 7.10: Experiment B-HS described in Section 7.3, gas saturation in Port 14 for various values of mass transfer coefficient for dissolution k_{dis}.

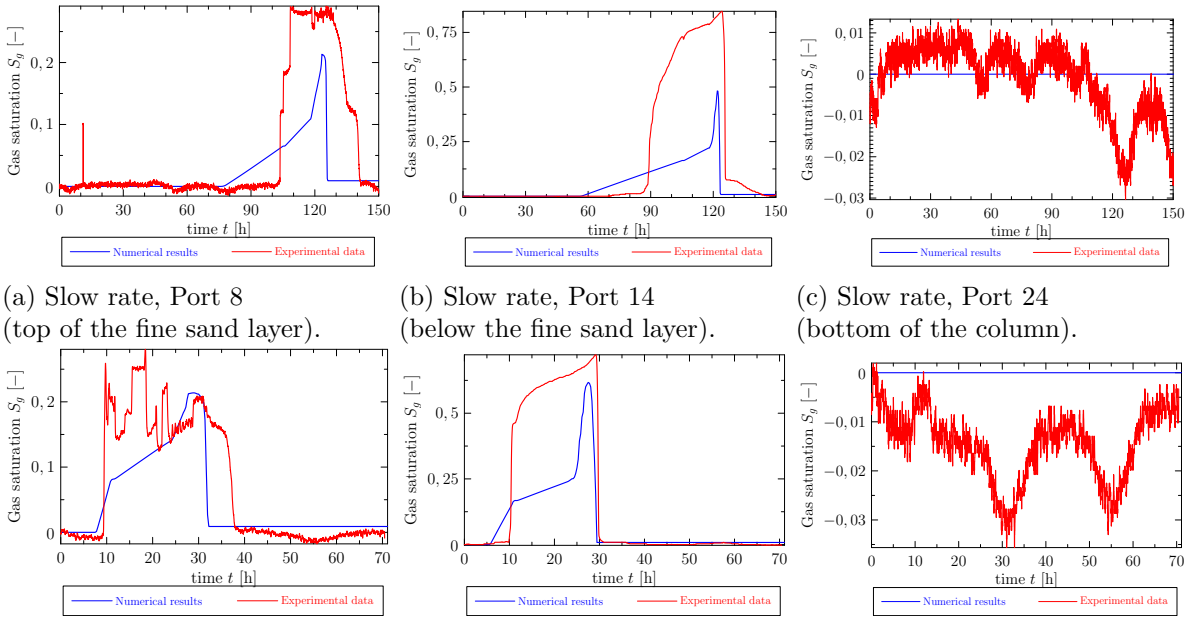
parameter of the depth at which the CO_2 first exsolves and the flow rate has a negligible role as the vertical profiles are very similar for both slow (**4 ml min^{-1}**) and fast (**40 ml min^{-1}**) inflow rates.

The accumulation of gaseous CO_2 below the fine sand layer observed in Port 14 located in the vicinity of the material interface is shown in Figures 7.11, 7.12, and 7.13 where high gas saturation values are detected. In this region, the gas saturation values are not determined by the value of the critical gas saturation as in the case of the ports at the bottom of the tank, instead, the capillary barrier described in Section 2.3.8 plays the key role [86] and enough gaseous CO_2 must be present in the tank to accumulate to the amount observed in the experiments.

The impact of the flow transition from fine to coarse sand can be observed in Port 8 located in the vicinity of this type of material interface. In this case, both numerical results and measured values of the gas saturation do not show accumulation to the extent observed in Port 14. Instead, a behavior similar to the ports at the bottom of the tank investigated in the exsolution stage in Section 7.3.2 is observed: after the initial growth, the gas saturation fluctuates around a certain value determined by the value of the critical gas saturation S_c. The measured value indicates that its value for the fine sand is the same as for the coarse one obtained in Stage 1 in Section 7.3.2.

The computed water flow velocities along the vertical axis of the tank are shown in Figure 7.15a for the selected times of B-HS and B-HF experiments. The vertical profiles at the start of the experiment when the tank is filled with water only are shown in Figure 7.15a. The slow and fast experiments with the CO_2-saturated water injection rates of **4 ml min^{-1}** and **40 ml min^{-1}**, respectively, cover the ranges of injection rates observed in the experiments. The high overpressure experiments were selected for this demonstration because of the highest amount of gas dissolved in the water and, consequently, the highest amount of exsolved gas present in the tank limiting the water flow. Therefore, for these two experiments, there will be the largest difference in velocities between the start of the experiment and the end of the CO_2-saturated water injection which is shown in Figure 7.15b. The comparison between the start and end of the CO_2-saturated water injection shows a significant drop in velocity magnitude in regions with high gas saturation. While the injection rate of the CO_2-saturated water remained constant, the injection rate of the clean water varied according to the schedule given in Table 7.6. However, the first clean water

injection of B-HS and B-HF experiments: **32 ml min^{-1}** and **6 ml min^{-1}**, respectively, covers the range of flow rates observed during the clean water injection for all the experiments. The only exception is the end of the HS experiment, however, there was almost no gas present in the tank at this stage of the experiment. The velocities at the start of the clean water injection for these two experiments are shown in Figure 7.15c. The corresponding gas saturation profiles are the same as for the end of the CO_2-saturated water injection shown in Figure 7.14. The x-axis range is chosen with respect to the velocity magnitude in the tank, therefore, the highest velocity magnitude in the vicinity of the injection port is out of the range in Figure 7.15, the maximal value of **13.16 m day^{-1}** is reached during the **40 ml min^{-1}** injection rate.



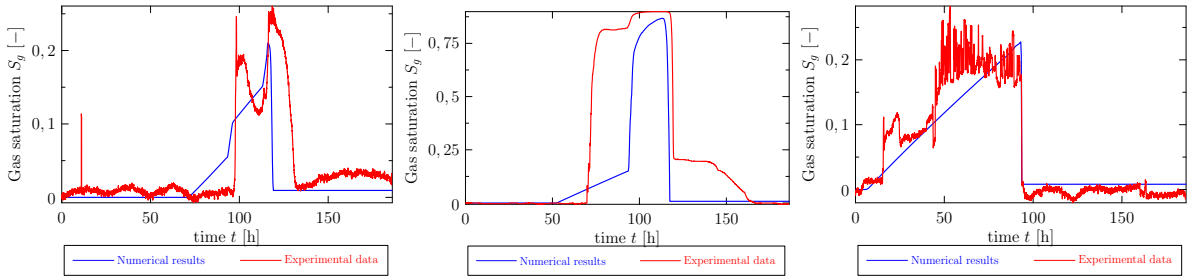
(a) Slow rate, Port 8 (top of the fine sand layer).
(b) Slow rate, Port 14 (below the fine sand layer).
(c) Slow rate, Port 24 (bottom of the column).
(d) Fast rate, Port 8 (top of the fine sand layer).
(e) Fast rate, Port 14 (below the fine sand layer).
(f) Fast rate, Port 24 (bottom of the column).

Figure 7.11: Low overpressure column experiments B-LS and B-LF described in Section 7.3, gas saturation S_g in selected ports.

Similarly to Experiment A, the computational study in the quasi-1D case showed that the numerical model is capable to capture the dynamics of the exsolution and dissolution processes observed in the experiments.

The optimal values of k_{exs} and k_{dis} were determined to be the same and equal (or larger) to the threshold value of **5 s^{-1}**, for which further increase of the coefficients has a negligible effect on the numerical results.

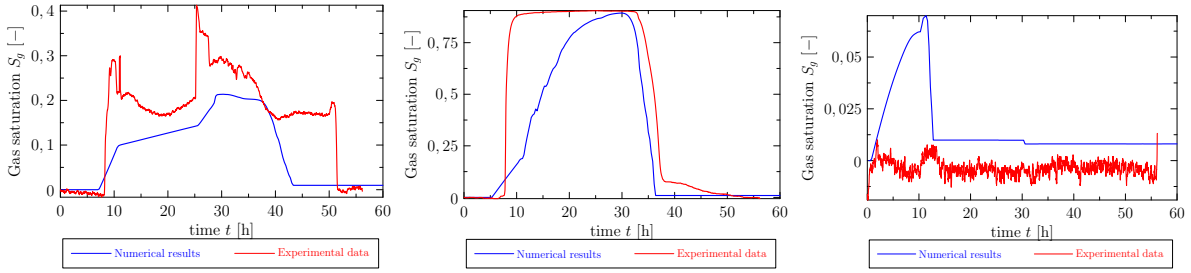
Small negative values of gas saturation were measured during the experiment. Although the negative values were obtained for all the experiments when there was no gas present in the vicinity of the gas saturation sensor, due to various ranges on the plots showing the saturation values, they are apparent only in the case when no gas is detected by the sensor for the whole duration of the experiment, for example, Port 24 in the low overpressure experiments. These negative saturation values are attributed to the post-processing calibration technique with the error of the gas saturation measurements of approximately **±0.05**. The negative readings reported here are within the measurement error.



(a) Slow rate, Port 8 (top of the fine sand layer).

(b) Slow rate, Port 14 (below the fine sand layer).

(c) Slow rate, Port 24 (bottom of the column).

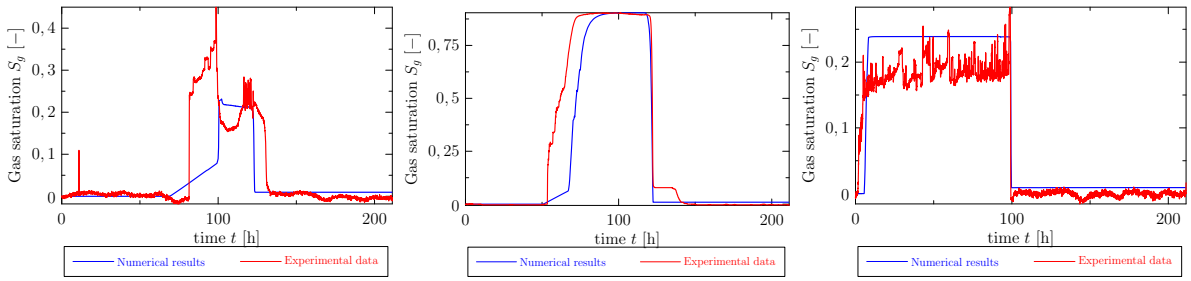


(d) Fast rate, Port 8 (top of the fine sand layer).

(e) Fast rate, Port 14 (below the fine sand layer).

(f) Fast rate, Port 24 (bottom of the column).

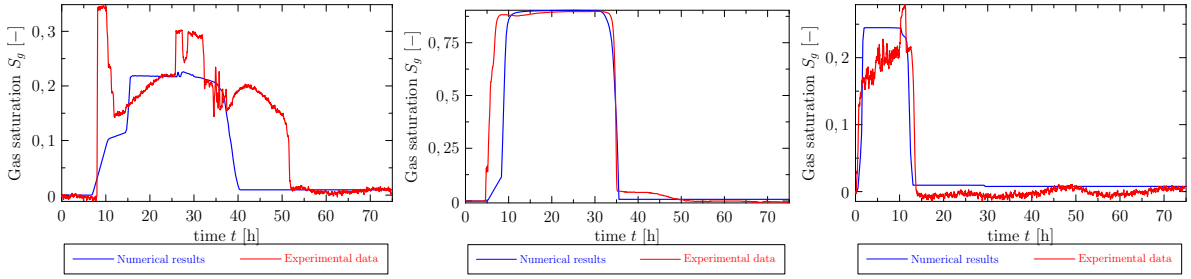
Figure 7.12: Medium overpressure column experiments B-MS and B-MF described in Section 7.3, gas saturation S_g in selected ports.



(a) Slow rate, Port 8 (top of the fine sand layer).

(b) Slow rate, Port 14 (below the fine sand layer).

(c) Slow rate, Port 24 (bottom of the column).

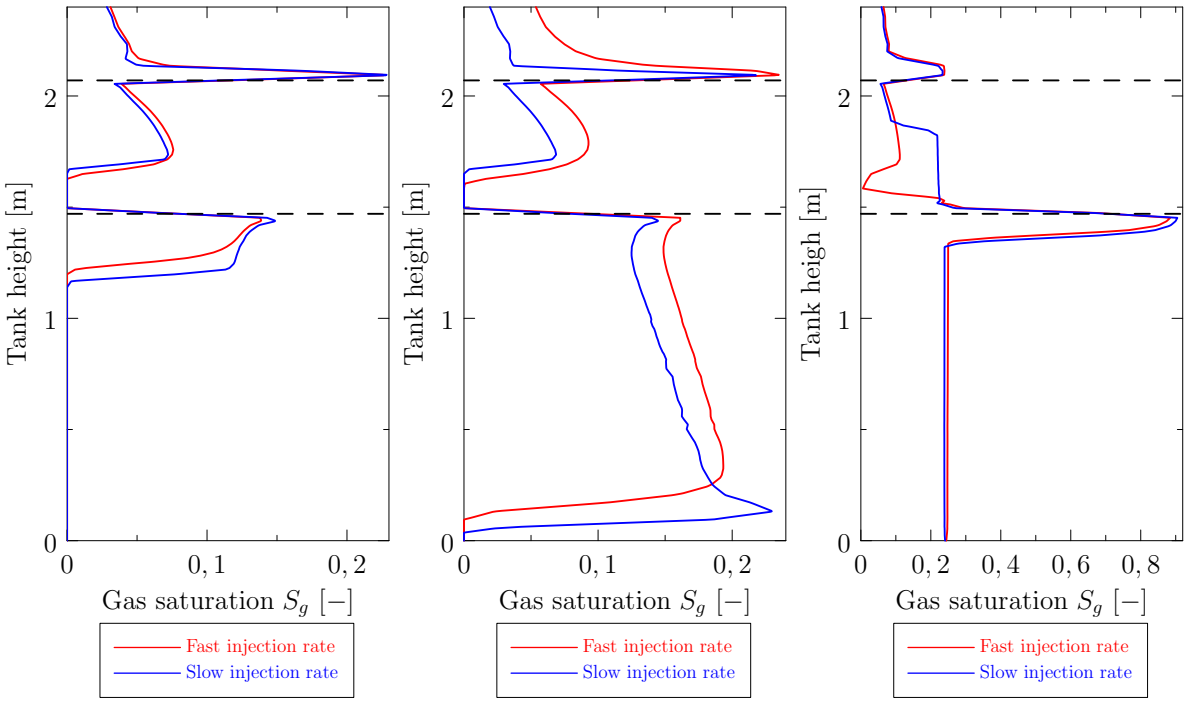


(d) Fast rate, Port 8 (top of the fine sand layer).

(e) Fast rate, Port 14 (below the fine sand layer).

(f) Fast rate, Port 24 (bottom of the column).

Figure 7.13: High overpressure column experiments B-HS and B-HF described in Section 7.3, gas saturation S_g in selected ports.



(a) Low overpressure. (b) Medium overpressure. (c) High overpressure.

Figure 7.14: Gas saturation distribution over the y-axis (tank height) of the tank at the end of the injection period for Experiment B described in Section 7.3. The material interfaces are depicted using the dashed lines.

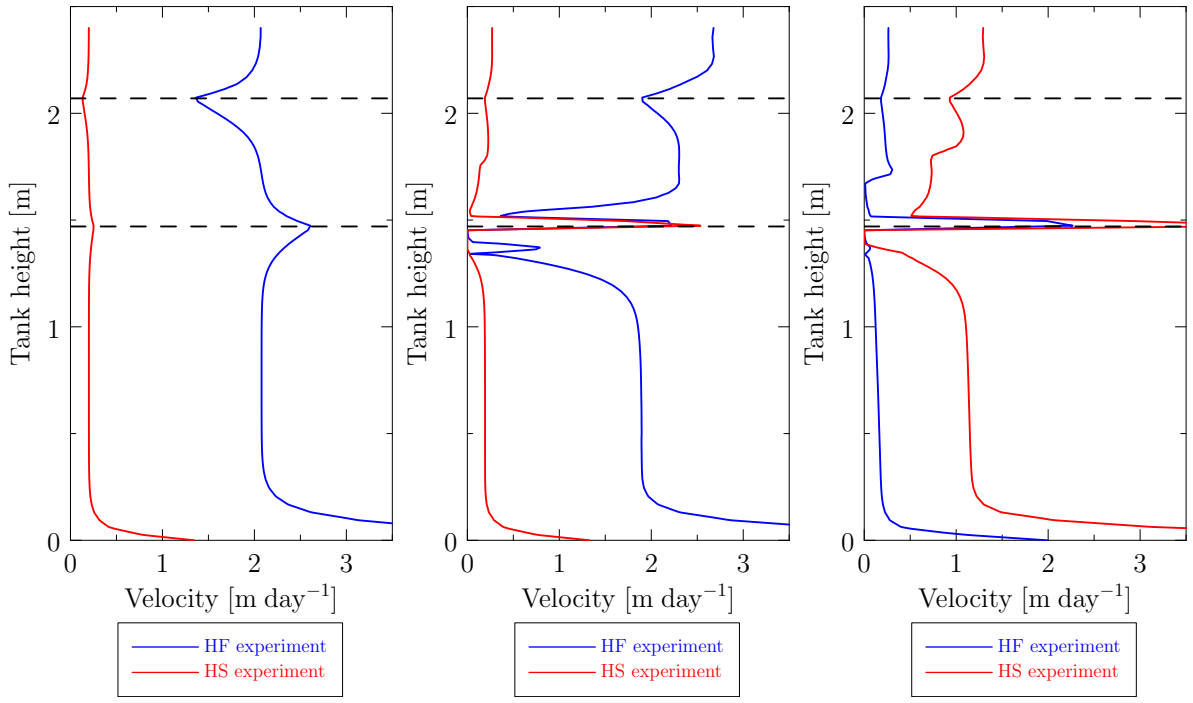
7.4 Experiment C

In this experiment, the multiphase CO_2 evolution is investigated in the larger laboratory-scale tank with the two different packing configurations allowing for a two-dimensional flow field. For more details about Experiment C, see [88]. Similarly to the previous experiments, the optimal values of the model parameters are determined using the measured gas saturation data, and then, the numerical results are used to obtain more information about the CO_2 evolution that would be possible from the experimental data only. The main goal is to demonstrate the difference in the dynamics of the exsolution and dissolution processes between the case of the larger-scale two-dimensional flow field and the previous two experiments presented in Sections 7.2 and 7.3. In one of the runs of Experiment C, the isothermal assumption was violated allowing for studying the effects of varying temperature on the CO_2 evolution.

In addition to the computational study to find the optimal parameter values for this larger-scale problem, we also consider high-resolution computations. Once the optimal values of the model parameters are determined, the computation for these parameters is run on two finer meshes and the results are compared to the coarse mesh used for the sensitivity study.

7.4.1 Experiment and model setup

The dimensions of the sand regions, the positions of the gas saturation sensors, and the numerical mesh used for the parameter sensitivity study are shown in Figure 7.16. The mesh parameters are listed in Table 7.9. The mesh resolution is selected to allow for running many simulations with different parameters in a reasonable time, and at the same time, the resolution is sufficient to capture the CO_2 mass transfer and transport processes. The gravel and clay layers are for



(a) Start of the experiment. (b) End of the injection period. (c) Start of the clean water injection.

Figure 7.15: Water velocity profile over the y-axis (tank height) in the middle of the tank for the B-HS and B-HF experiments described in Section 7.3. The material interfaces are depicted using the dashed lines.

simplicity not included in the model and the interior of the tank is considered only. The main role of these blocks is to uniformly distribute the pressure and split the outflow into several parts. In the model, the same settings are obtained by prescribing suitable boundary conditions.

The experiment started by connecting constant head devices to the inflow and outflow sides of the tank and positioning them to establish a lateral background flow of water through the tank. Once the flow in the tank was established, water saturated with CO_2 at **13 kPa** overpressure was injected through the port at the bottom of the tank with the injection rate of **11.2 ml min^{-1}** for a certain period of time.

The dissolved CO_2 plume spread in the tank and its shape was controlled by the combination of the background flow and the injection rate. The overpressure was high enough for a portion of CO_2 to exsolve and migrate through the tank as a gas phase. After the injection of CO_2-saturated water stopped, the experiment continued with the background flow only. The gas present in the tank dissolved into the clean water and, together with the original dissolved CO_2 plume, it was transported further to the outflow side of the tank.

Experiment C consisted of two runs, which differed in the type of the fine sand and the length of the injection period. In the low contrast experiment run, the fine sand was Accusand #40/50 which was only slightly finer than the surrounding coarse sand Granusil #20/30. In the high contrast experiment run, the fine sand was Unimin #110/250 which is much finer than the one used in the rest of the tank. Parameters of these three types of sand are listed in Table 7.10. The CO_2-saturated water injection period lasted for 1.88 days in the high contrast experiment and for 2.43 days in the low contrast experiment. In this study, we consider only the first ten days of both experiments where the dynamics of the dissolution and exsolution processes can be investigated. Positions of the injection port and the gas saturation sensors were the same for

Mesh ID	h [m]	# Elements	# Sides*	# DOF	Δt [s]
Coarse	$3.48 \cdot 10^{-2}$	5 638	8 578	25 734	5
Medium	$7.41 \cdot 10^{-3}$	134 640	202 551	607 653	10
Fine	$3.54 \cdot 10^{-3}$	534 984	803 656	2 410 968	4

Table 7.9: Mesh properties for Experiment C described in Section 7.4.

both experiments and the background water flow through the tank was very similar for both runs.

In addition to the differences in the setup, these two experiments differed in the ambient temperature during the experiment runs. For the duration of the low contrast experiment, the temperature was almost constant. The difference between the lowest and highest temperature during the 10 days of the experiment considered in this work was less than $2\ °C$ as shown in Figure 7.17. However, temperature fluctuations on a daily basis were observed during the high contrast experiment. These fluctuations shown in Figure 7.17 were caused by the air conditioning issues in the CESEP experimental facility and the difference between the lowest and highest temperature within a day exceeded $10\ °C$.

The fluctuations in the temperature were not planned for the experiments that were supposed to be run as isothermal. Therefore, the tank was not insulated and the heat fluxes and thermal conductivity of the materials were not measured. Hence, due to the lack of information needed to specify the material properties and prescribe the appropriate initial and boundary conditions, the energy balance equation is not included in the mathematical model. Instead, a simplified approach is employed, where only the temporal violation of the isothermal assumption is considered, i.e., at a given time, the temperature is considered constant in the whole tank. This assumption is justifiable because the tank is narrow compared to its other dimensions and its walls are not insulated. The only parameter that is considered as a function of temperature is the saturated concentration of CO_2 dissolved in water C_s given by (2.49).

Sand	ϕ [−]	K [m^2]	S_{rw} [−]	S_{rn} [−]	p_d [Pa]	λ^{BC} [−]
Granusil #20/30	0.41	1.21×10^{-10}	0.10	0	1580	5.79
Accusand #40/50	0.42	5.23×10^{-11}	0.07	0	1940	4.09
Unimin #110/250	0.35	6.39×10^{-14}	0.17	0	8100	5.35

Table 7.10: Sand properties for Experiment C in Section 7.4. Adapted form [85, 88].

Initial and boundary conditions

The initial conditions are given as follows. At the start of the computation, the tank contains only pure water with no background flow: $X = 0$, $p_c = p_d$, and $p_g = p_c + p_\ell$ where the hydrostatic profile for p_ℓ was prescribed. For both low and high contrast experiments, the final time is $t_{fin} = 10$ days and the fixed time step $\Delta t = 5$ s is used.

The following boundary conditions are prescribed:

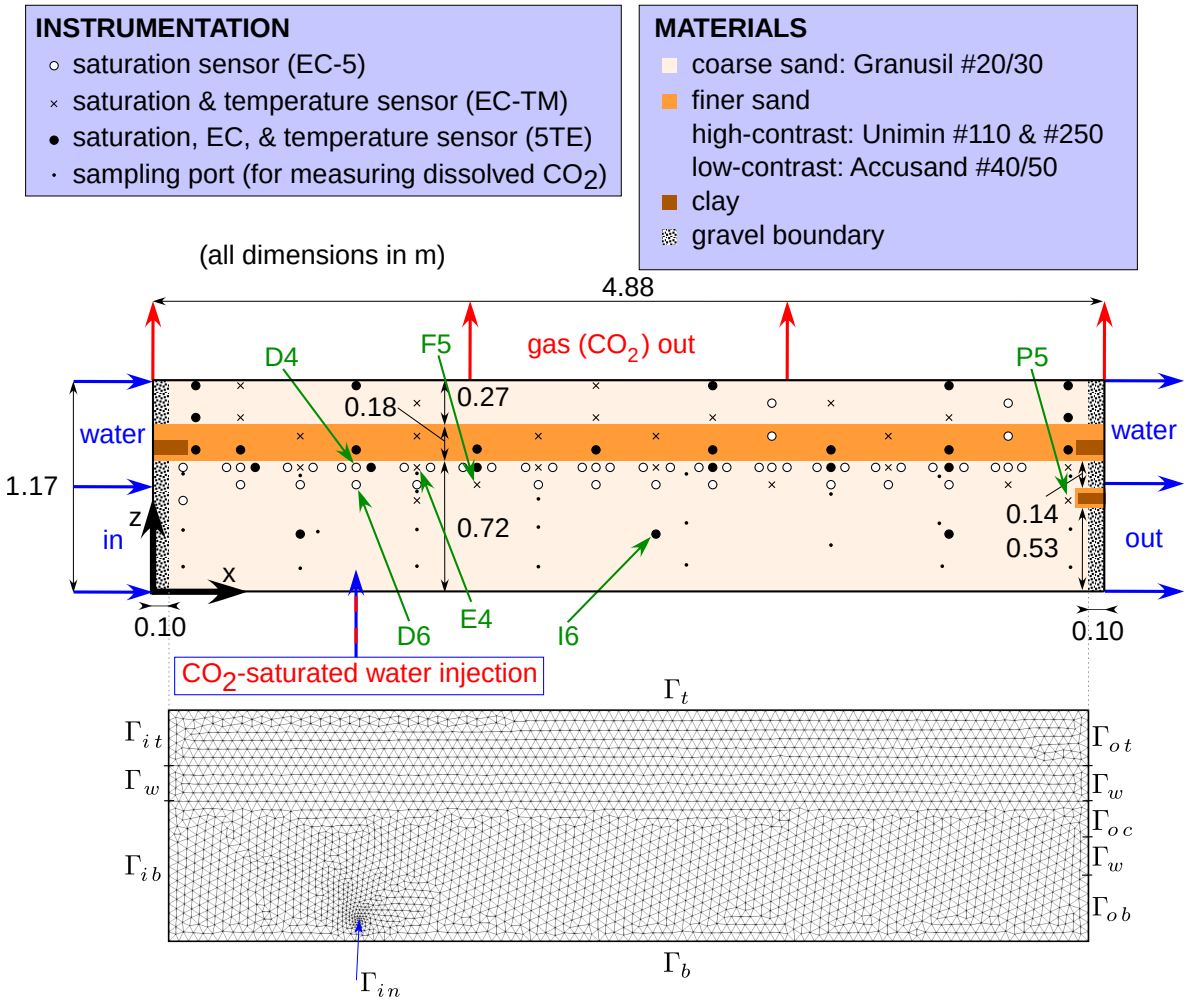


Figure 7.16: Configuration of Experiment C described in Section 7.4 adapted from [88] and the triangular mesh of 5638 elements used in numerical simulations. EC stands for the electric conductivity sensor and EC-5, EC-TM, and 5TE denote the specific type of a sensor, for more details, we refer the reader to [88].

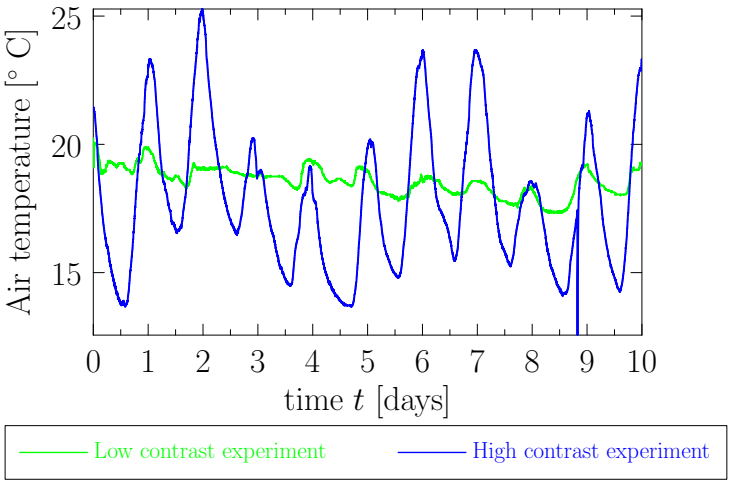


Figure 7.17: Air temperature during Experiment C described in Section 7.4.

Γ_w, Γ_b — No-flow boundary condition is prescribed for both phases and for the dissolved CO_2 mass fraction.

Γ_{in} — No-flow condition for the gas phase is prescribed for the whole duration of the experiment. When the CO_2-saturated water is injected, the Neumann boundary condition for water velocity is prescribed with $\boldsymbol{u}_\ell$ computed to match the injection rate and the Dirichlet boundary condition for X is prescribed computed from the overpressure and temperature as described in Section 7.4.1. For the rest of the experiment, the no-flow boundary condition is prescribed also for water and the dissolved CO_2 mass fraction.

Γ_{it}, Γ_{ib} — $X = 0$, $p_c = p_d$, and $p_g = p_c + p_\ell$ where the hydrostatic profile for water pressure p_ℓ corresponding to the constant head device on the left-hand side of the tank are prescribed.

Γ_{ot}, Γ_{oc}, Γ_{ob} — The Neumann boundary conditions for the water velocities are prescribed with $\boldsymbol{u}_\ell$ given by the outflow rates measured in [88]. For the other two unknowns, $\nabla X \cdot \boldsymbol{n} = 0$ and the no-flow boundary condition for the gas phase are prescribed. This approach was selected to reproduce the flow field in the tank and the water outflow rates that in the experiment varied although the water head was kept constant. This effect can be attributed to the partial clogging of the outflow during the experiment.

Γ_t — The gas pressure $p_g = p_{ref}$ is kept constant, where $p_{ref} = 82$ kPa is the reference atmospheric pressure in the laboratory, $\nabla X \cdot \boldsymbol{n} = 0$, and the no-flow boundary condition for water are prescribed.

Experiment C is for the purpose of the computational study divided into two stages: 1) exsolution and 2) dissolution. The design of the experiment allowed for almost isolating the dissolution and exsolution processes and, therefore, the unknown mass transfer coefficients k_{exs} and k_{dis} can be determined independently.

7.4.2 Critical gas saturation

In contrast to Experiment B described in Section 7.3, there is no separate stage that can be used to determine the critical gas saturation S_c. There are no gas saturation sensors in the vicinity of the injection port and higher gas saturation values comparable to the lower ports in Experiment B are detected in the ports directly below the low permeable layer only, where the amount of CO_2 is controlled by the capillary barrier effect rather than the value of critical gas saturation. Overall, the amount of gas present in the tank is lower as indicated also by the measured gas saturation in the lower homogeneous section of the tank, where the maximal values around 0.1 are detected which indicates that the role of the critical gas saturation is not as significant as in the case of the column experiment. The critical gas saturation gives a threshold value above which the gas phase becomes mobile. However, in this experiment, the main mechanism controlling the gas evolution in a given place is the flow of water with dissolved CO_2 and exsolution at the position rather than the flow of mobile gas. Therefore, the only conclusion that can be drawn about the value of critical gas saturation S_g in this case is that the value is at least 0.1.

7.4.3 Stage 1: exsolution

The exsolution stage covers the injection of CO_2-saturated water into the tank and the exsolution of CO_2 in the vicinity of the injection port. The rest of CO_2 that remains dissolved in water is transported by the background flow in the downstream direction and can exsolve further away from the injection port. The gas phase is well detected in Ports D4, D6, E4, and F3 as shown in Figure 7.18.

In this section, we focus on the low contrast experiment, where the temperature fluctuations were negligible and, therefore, the isothermal model is used. The procedure to find the optimal value of the mass transfer coefficient k_{exs} and to demonstrate the behavior for a wide range of values was similar to the one used in Experiment B. Numerical realizations were computed for a series of values of k_{exs} in this case with a selected step of 0.001 [s^{-1}]. Then, in all the ports, the computed values of S_g were compared to the measured ones and the difference between them was evaluated using the Euclidean norm.

For this experiment, a different behavior than for the previous two is observed. When the value of k_{exs} increases, the amount of gas detected in downstream ports decreases. For values of k_{exs} comparable to those in the previous experiments representing the near-equilibrium approach, CO_2 exsolves rapidly in the vicinity of the injection port and migrates upwards as can be seen in the computed values in Port D4 in Figure 7.19. As a result, less CO_2 remains dissolved in water flowing downstream and little or almost no gas is detected in the downstream ports as shown in Figure 7.19 for the selected Ports E4 and F5. The lower amount of gas present in the downstream section of the tank can be also seen in Figure 7.20 where the comparison between the gas saturation results for the near-equilibrium: $k_{exs} = 0.1\ s^{-1}$ and rate-limited kinetic: $k_{exs} = 0.005\ s^{-1}$ models is shown for a section of the tank around the injection port.

For both experiments, the optimal value of $k_{exs} = 0.005\ s^{-1}$ that minimized the difference between experimental and computed values of S_g was determined. The necessary modifications of the model for the high contrast experiment are discussed later in Section 7.4.4. In contrast to Experiment A and Experiment B, significantly lower values, far from the values representing the local equilibrium approach of the mass transfer coefficients, must be used in order to capture the gas evolution correctly as illustrated in Figure 7.19.

7.4.4 Thermal effects

In Section 7.4.3, the low contrast experiment was investigated using the isothermal model because the temperature fluctuations during the experiment run were negligible. However, for the high contrast experiment, the temperature fluctuations were more significant and, without taking into account the influence of temperature, it is impossible to find a suitable value of mass transfer coefficient for exsolution k_{exs} to match the experimental data.

The difference between the isothermal approach with saturated concentration C_s computed for the averaged temperature of $19\ °C$ that is the same for both experiments and the non-isothermal approach when C_s is computed using the measured temperature value is illustrated in Figures 7.21 and 7.22 for low and high contrast experiments, respectively.

When the constant temperature is used, the gas distribution appears to be similar for both low and high contrast experiments. The amount of gas present in the tank is not high enough for the more significant barrier effect of the fine sand in the high contrast experiment to play an important role. Since the temperature fluctuations in the low contrast experiment shown in Figure 7.17 were small, the results in Figure 7.21 confirm that the difference between a constant and varying temperature is small and thus justifies the isothermal approach. However, there is a significant difference between the constant and varying temperature results for the high contrast experiment shown in Figure 7.22. In the case of varying temperature, there is more gas present

in the tank and the barrier effect of the fine sand layer plays a more important role. In addition, there is a larger downstream spreading of the gas plume. Therefore, the varying temperature model is considered to match the experimental data for the high contrast experiment.

7.4.5 Stage 2: dissolution

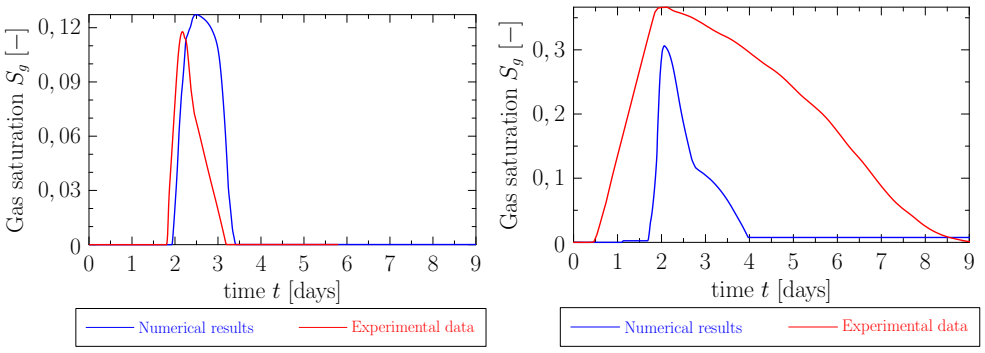
The mass transfer coefficient for the dissolution was found using an analogous procedure. Based on the findings of the exsolution process, the saturated concentration C_s was considered to be a function of temperature in the case of the high contrast experiment. The optimal value of $k_{dis} = 0.002\ \mathrm{s}^{-1}$ was obtained for both low and high contrast experiments. The comparison of numerical results and experimental data for the selected ports shown in Figure 7.18 indicates that the numerical model is capable of capturing the gas saturation evolution sufficiently well.

7.4.6 Discussion of results

The water flow velocity profiles along the horizontal cross-sections at the level of the injection port ($y = 0.1$ m) and below the fine sand layer ($y = 0.7$ m) in the selected times are shown in Figure 7.23. In contrast to the column experiment, there were not that significant changes in the flow rates, therefore, the profiles at the start of the experiment and after 48 hours, when the gas accumulation below the fine sand layer is observed, are selected for comparison. Gas saturation S_g at these times is shown in Figures 7.20 and 7.22 for the low and high contrast experiments, respectively.

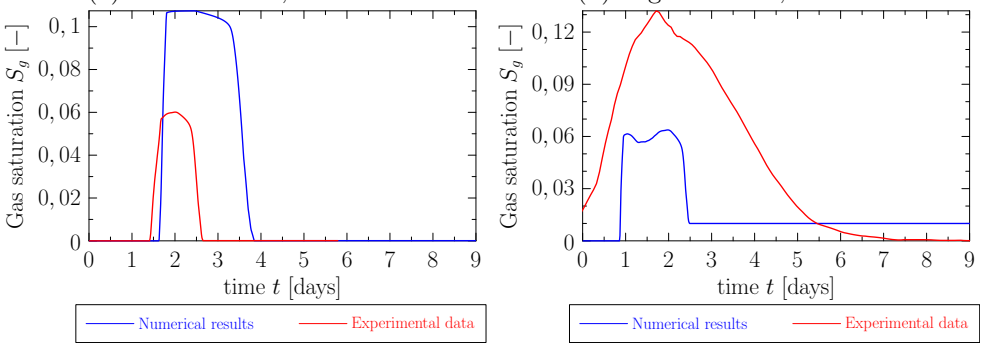
At the start of the experiment, the velocity profiles are similar for both runs and it can be seen that the type of fine sand does not have a significant impact on the flow field in these zones of the tank. After 48 hours, the water velocity below the heterogeneity is lower for the high contrast experiment than for the low contrast one, which can be attributed to a more significant gas accumulation. The difference at the level of the injection port is caused by a slightly different length of the injection period. For the low contrast experiment, the CO_2-saturated water is still injected after 48 hours, however, in the case of the high contrast experiment, the injection already stopped. The y-axis range is chosen with respect to the velocity magnitude in the tank, therefore, the maximal value of $2.85\ \mathrm{m\ day}^{-1}$ in the vicinity of the injection port is out of the range in Figure 7.23.

The numerical results show that the equilibrium mass transfer model is not sufficient for the scenarios of Experiment C. Its employment leads to overprediction of the exsolution rate in the vicinity of the injection port where the two-dimensional flow field is present. The kinetic model with the specific values of mass transfer coefficients is needed to capture the gas saturation evolution observed in the experiment. In addition, the optimal values of coefficients k_{exs} and k_{dis} found for this experiment indicate that the exsolution is approximately 2.5 times faster than the dissolution. These findings tend to confirm that the exsolution and dissolution processes differ and emphasize the importance of employing the kinetic mass transfer model that can take this difference into account. In general, the mass transfer rate coefficients k_{exs} and k_{dis} are quite low in comparison with the previous two experiments which, together with numerical results for higher coefficient values, mean that the mass transfer is rate-limited in this case. However, due to the available experimental data, it is beyond the scope of this work to rigorously investigate the mass transfer coefficients and establish a suitable expression as a function of flow and porous media properties such as relations proposed in Section 2.5.1. Further research is needed to extend the understanding of the multiphase evolution of fluids in porous media.



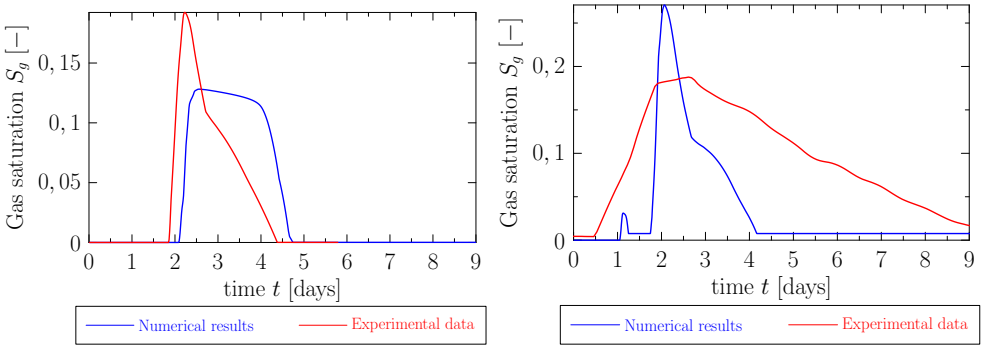
(a) Low contrast, Port D4.

(b) High contrast, Port D4.



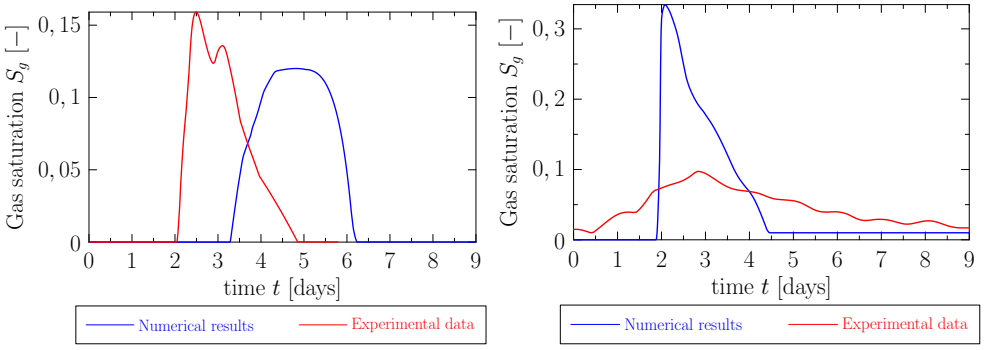
(c) Low contrast, Port D6.

(d) High contrast, Port D6.



(e) Low contrast, Port E4.

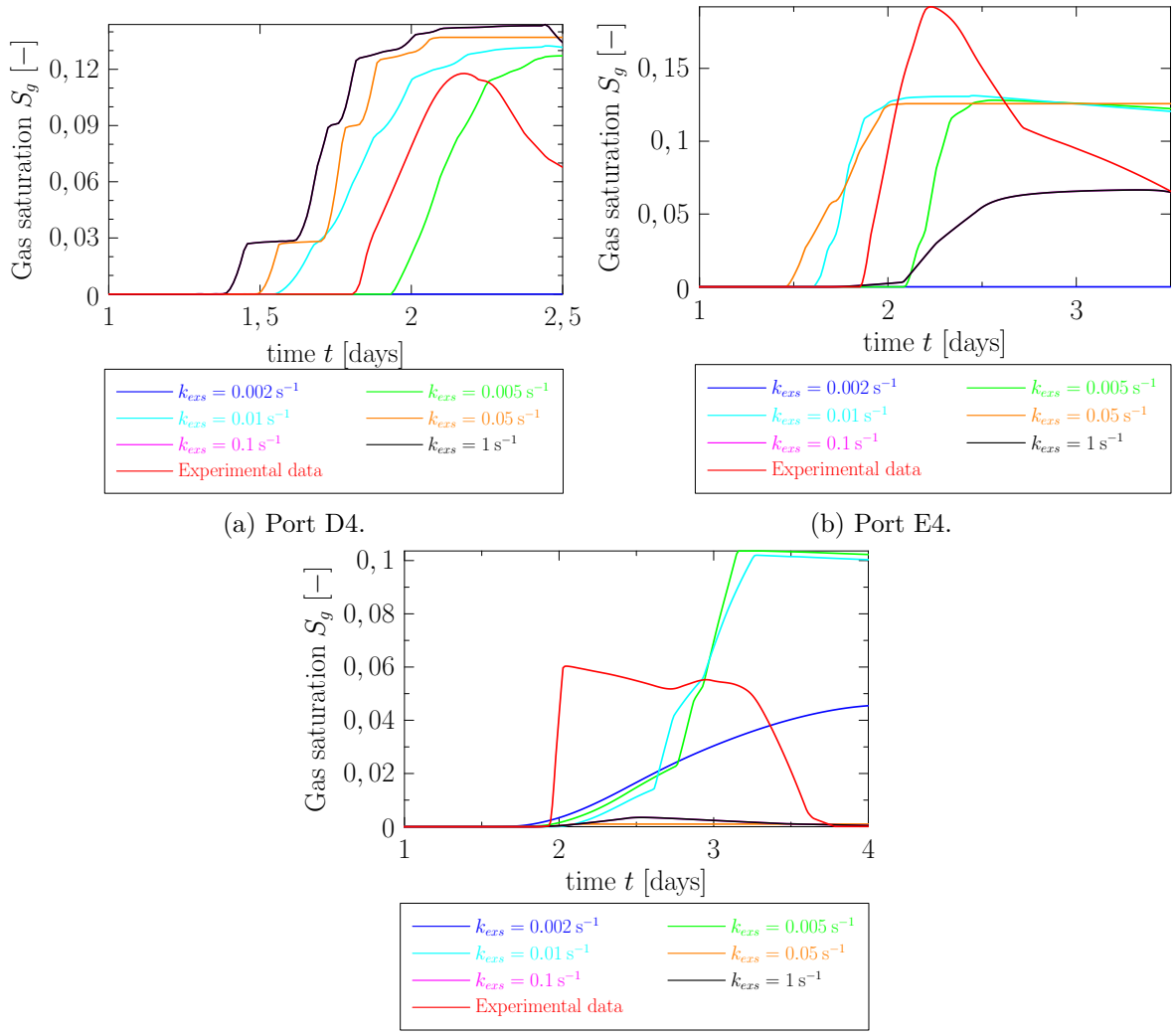
(f) High contrast, Port E4.



(g) Low contrast, Port F3.

(h) High contrast, Port F3.

Figure 7.18: Experiment C described in Section 7.4, gas saturation S_g, selected ports for the low and high contrast cases.



(a) Port D4.

(b) Port E4.

(c) Port F5.

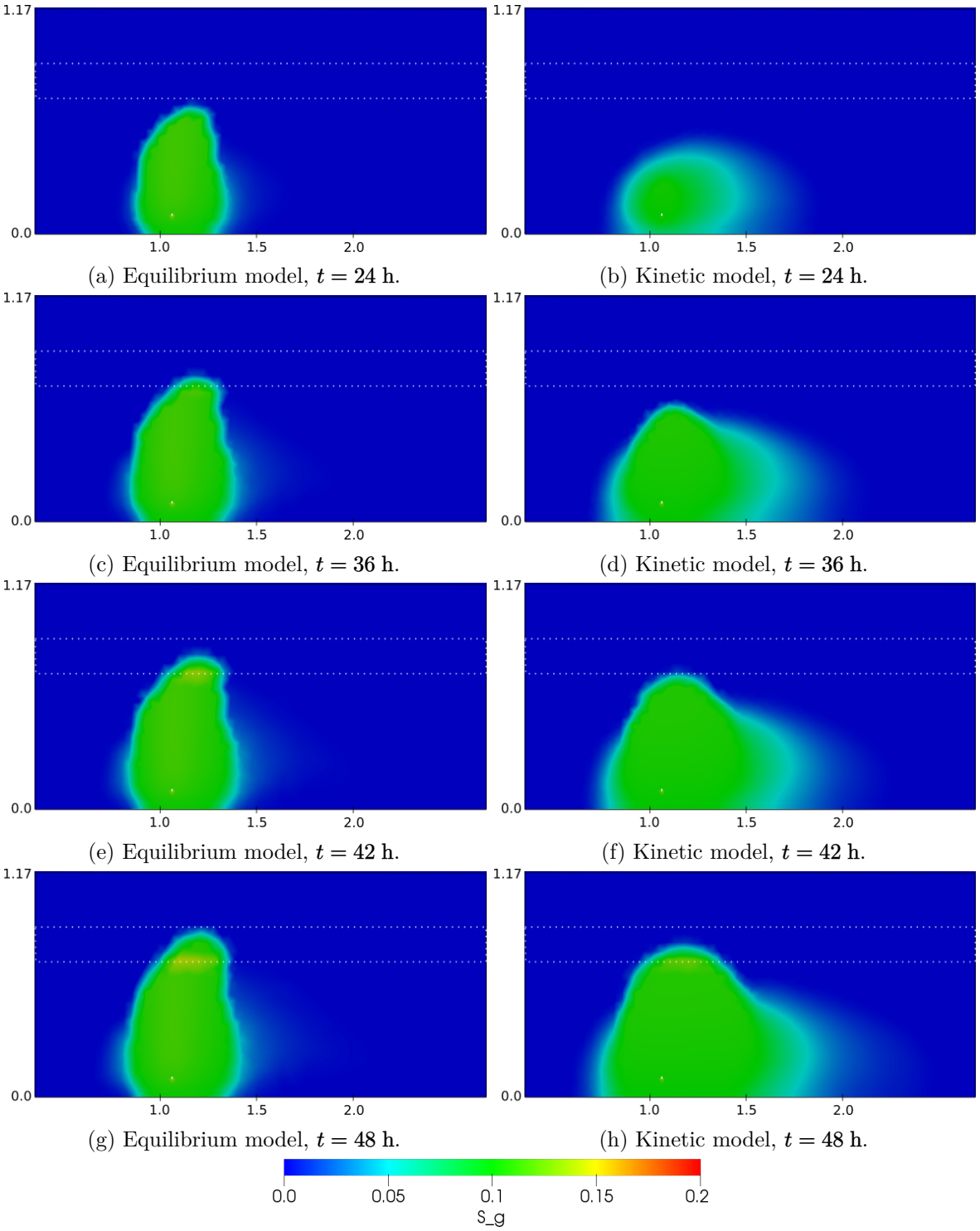
Figure 7.19: Experiment C described in Section 7.4, gas saturation S_g, various values of the mass transfer coefficient for exsolution.

7.4.7 High resolution computation

For the computations on the medium and fine mesh with properties given in Table 7.9, the following strategy proposed to solve the unsteady problems in Section 6.4.3 was employed: the solution from the previous time step was used as the initial vector for the BiCGStab iterations at the current time step and the preconditioner was reused until the trigger value of 10 iterations was exceeded.

The medium and fine meshes were divided into subdomains to get approximately 38 thousand degrees of freedom per subdomain. This corresponds to 16 and 64 subdomains for medium and fine mesh, respectively. In contrast to the sensitivity study, where for higher mass transfer coefficients k_{dis} and k_{exs} shorter time steps were enforced, here the computations for the coefficients $k_{exs} = 0.005\ \mathrm{s}^{-1}$ and $k_{dis} = 0.002\ \mathrm{s}^{-1}$ are carried out with longer time steps: $\Delta t = 10$ s and $\Delta t = 4$ s for medium and coarse mesh, respectively. Computation times, total number of BiCGStab iterations, and the number of preconditioner computations for both low and high contrast experiments are shown in Table 7.11.

The comparison of numerical results on all three meshes considered for this problem is shown



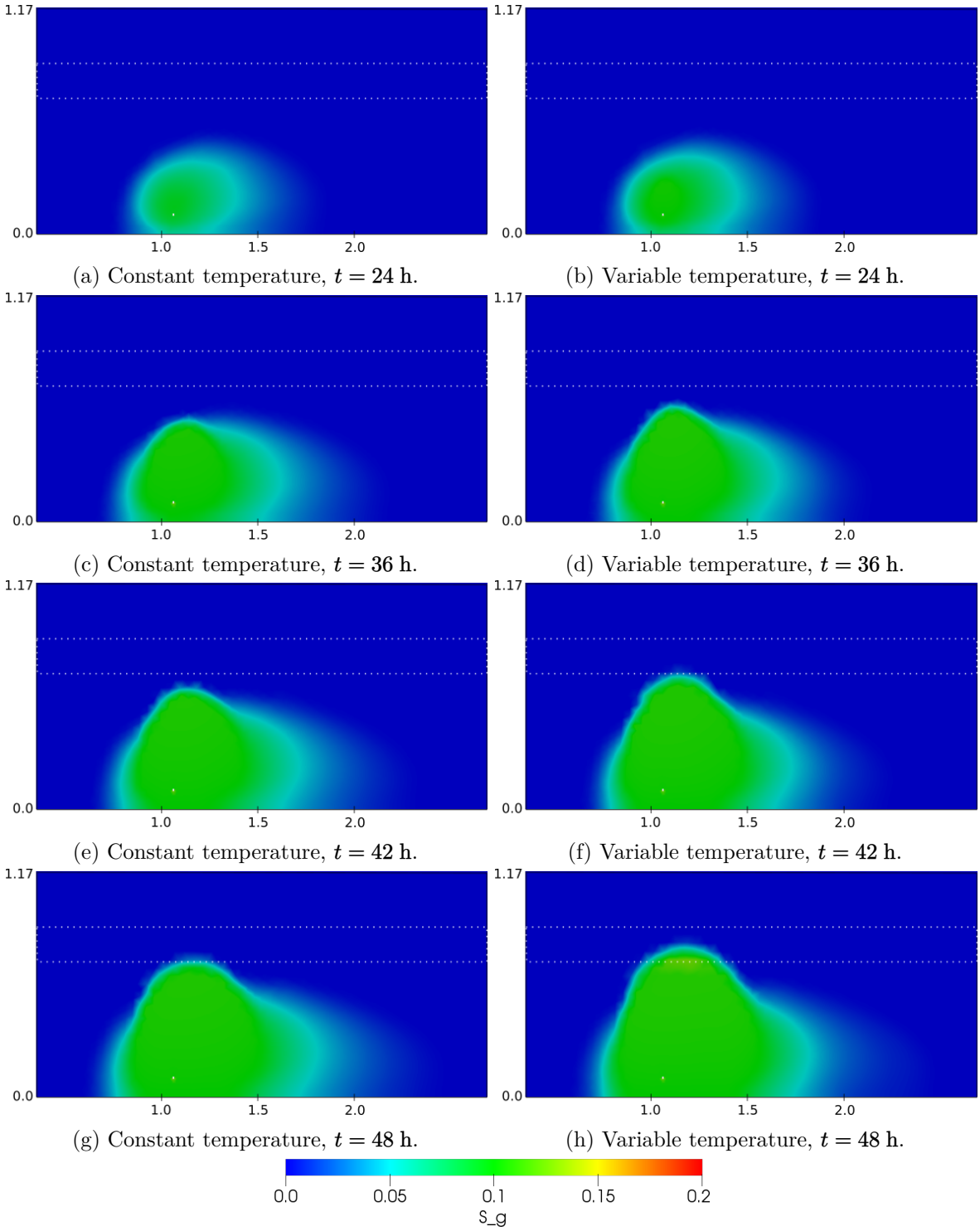
(a) Equilibrium model, $t = 24$ h. (b) Kinetic model, $t = 24$ h.

(c) Equilibrium model, $t = 36$ h. (d) Kinetic model, $t = 36$ h.

(e) Equilibrium model, $t = 42$ h. (f) Kinetic model, $t = 42$ h.

(g) Equilibrium model, $t = 48$ h. (h) Kinetic model, $t = 48$ h.

Figure 7.20: The gas saturation evolution using the near equilibrium (left) and rate-limited kinetic (right) mass transfer models for the low contrast Experiment C described in Section 7.4.



(a) Constant temperature, $t = 24$ h.

(b) Variable temperature, $t = 24$ h.

(c) Constant temperature, $t = 36$ h.

(d) Variable temperature, $t = 36$ h.

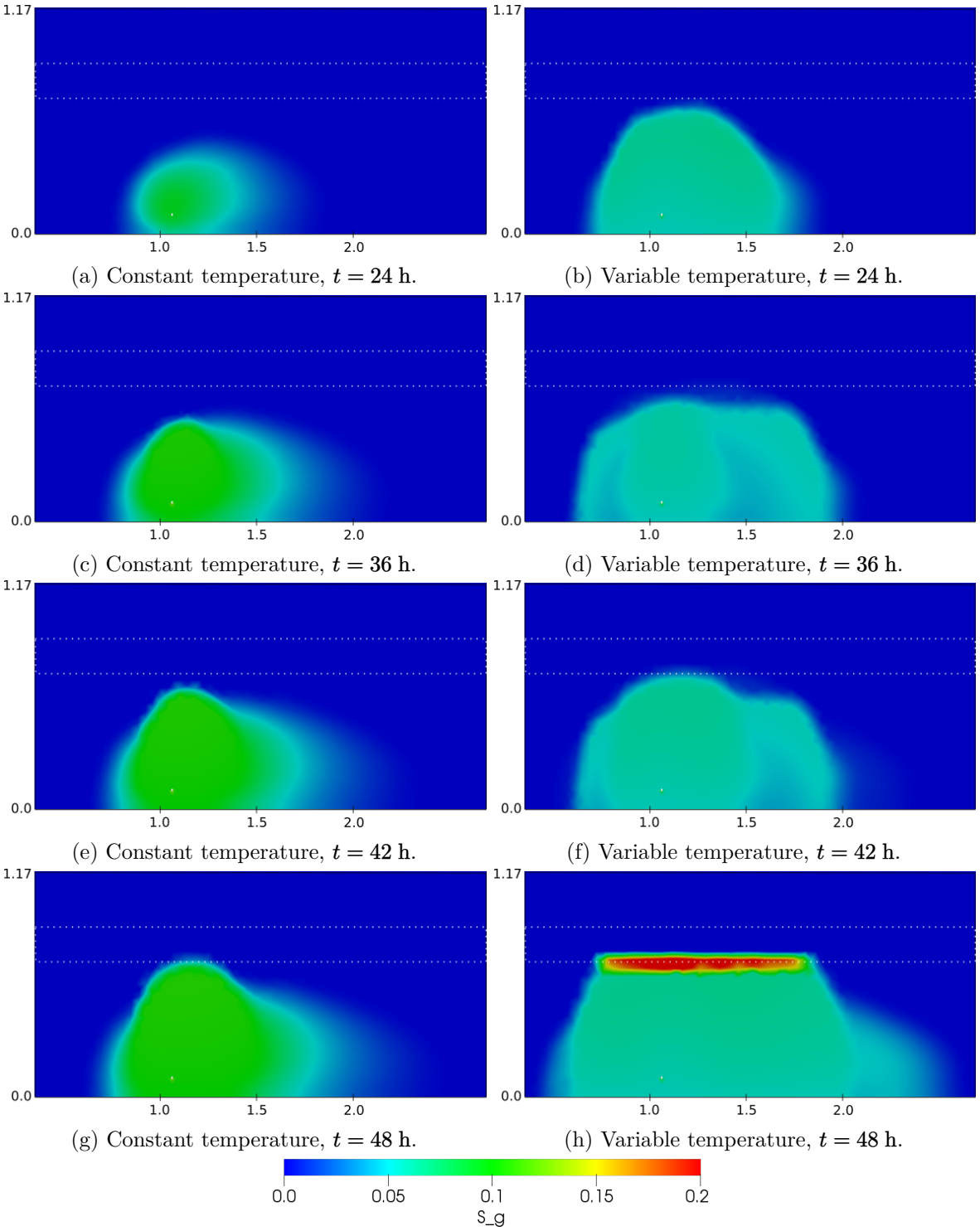
(e) Constant temperature, $t = 42$ h.

(f) Variable temperature, $t = 42$ h.

(g) Constant temperature, $t = 48$ h.

(h) Variable temperature, $t = 48$ h.

Figure 7.21: Gas saturation S_g, comparison of isothermal (left) and non-isothermal (right) models for low contrast Experiment C described in Section 7.4.



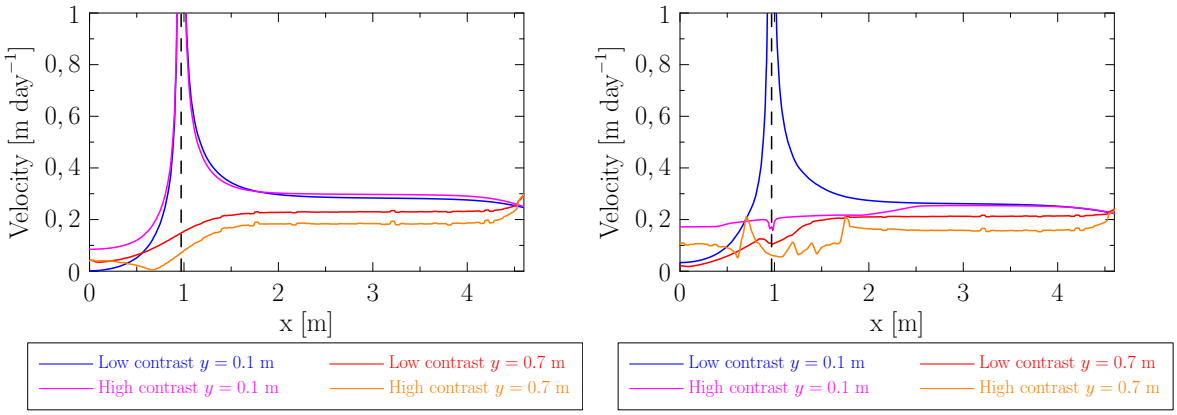
(a) Constant temperature, $t = 24$ h. (b) Variable temperature, $t = 24$ h.

(c) Constant temperature, $t = 36$ h. (d) Variable temperature, $t = 36$ h.

(e) Constant temperature, $t = 42$ h. (f) Variable temperature, $t = 42$ h.

(g) Constant temperature, $t = 48$ h. (h) Variable temperature, $t = 48$ h.

Figure 7.22: Gas saturation S_g, comparison of isothermal (left) and non-isothermal (right) models for high contrast Experiment C described in Section 7.4.



(a) Start of the experiment.

(b) $t = 48$ h.

Figure 7.23: Water velocity profile over the x-axis at the level of the injection port ($y = 0.1$ m) and below the fine sand layer ($y = 0.7$ m) for Experiment C described in Section 7.4. The position of the injection port is depicted using the dashed line.

Mesh resolution	Medium		Fine	
Experiment	Low contrast	High contrast	Low contrast	High contrast
# Subdomains	16		64	
# Time steps	86 400 ($\Delta t = 10$ s)		216 000 ($\Delta t = 4$ s)	
# BiCGStab iterations	570 177	490 202	1 380 437	1 129 254
Avg. per time step	6.6	5.7	6.4	5.2
# Preconditioner computations	23	21	51	48
Computational time [h]	16.9	21.0	49.3	62.8

Table 7.11: Statistics for high-resolution computations for Experiment C described in Section 7.4.

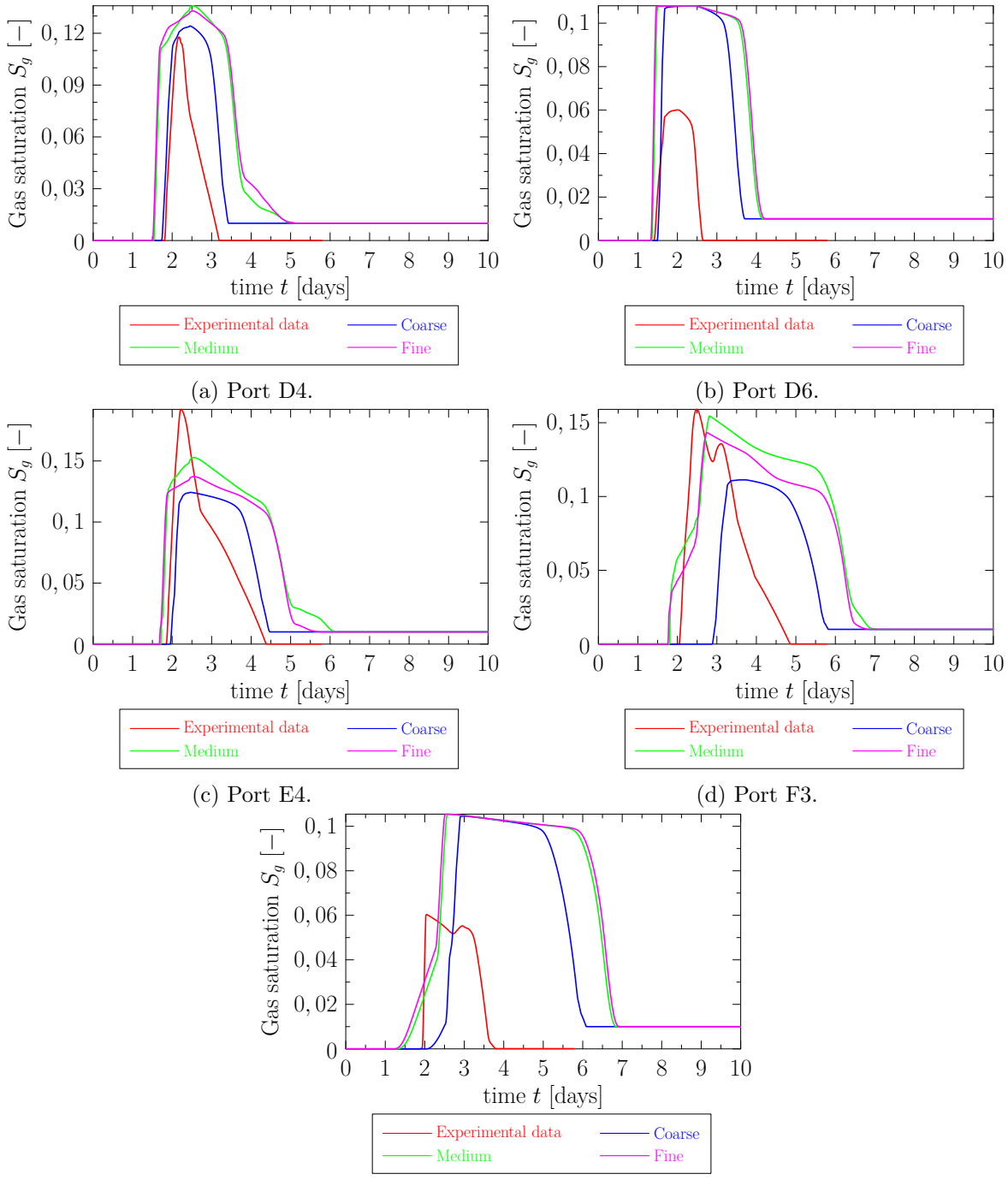
in Figures 7.24 and 7.25 for low and high contrast experiments, respectively.

The results for finer meshes do not differ significantly which indicates the numerical convergence of the solution. This justifies the approach taken to the parameter sensitivity study. The coarse mesh is sufficient to capture the studied effects and the computations are less demanding that the high-resolution ones.

The results presented in Table 7.11 show that reusing the preconditioner is advantageous in this case. The average number of BiCGStab iterations at a single time step varies between 5 and 7 for the considered mesh resolutions and time steps. Only a few preconditioner computations are required in comparison with the overall number of time steps in each computation. The longer computational times for the high contrast experiment can be attributed to the difference in material properties and different gas evolution dynamics.

7.5 Chapter summary

The results presented in this chapter show that the mathematical model is able to adequately capture important processes of the multiphase CO_2 evolution observed in the experiments. The



(a) Port D4.

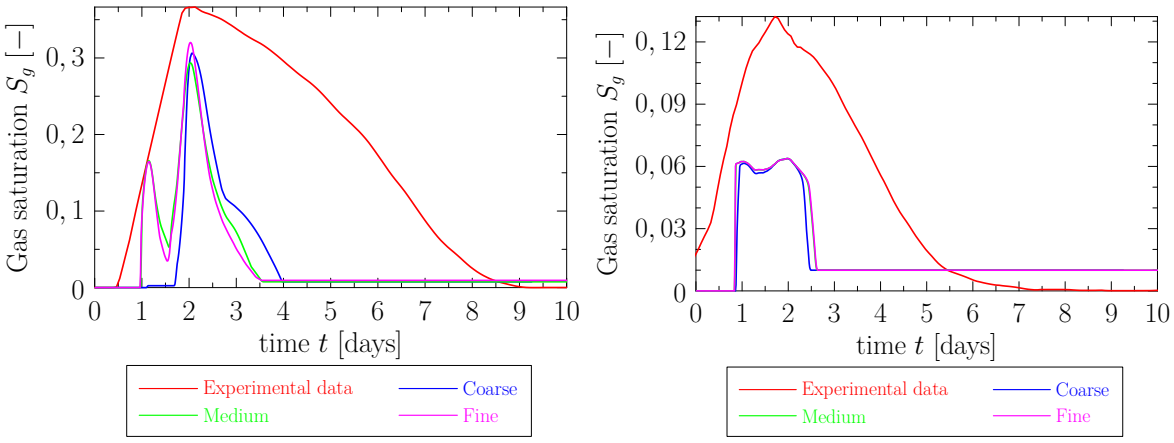
(b) Port D6.

(c) Port E4.

(d) Port F3.

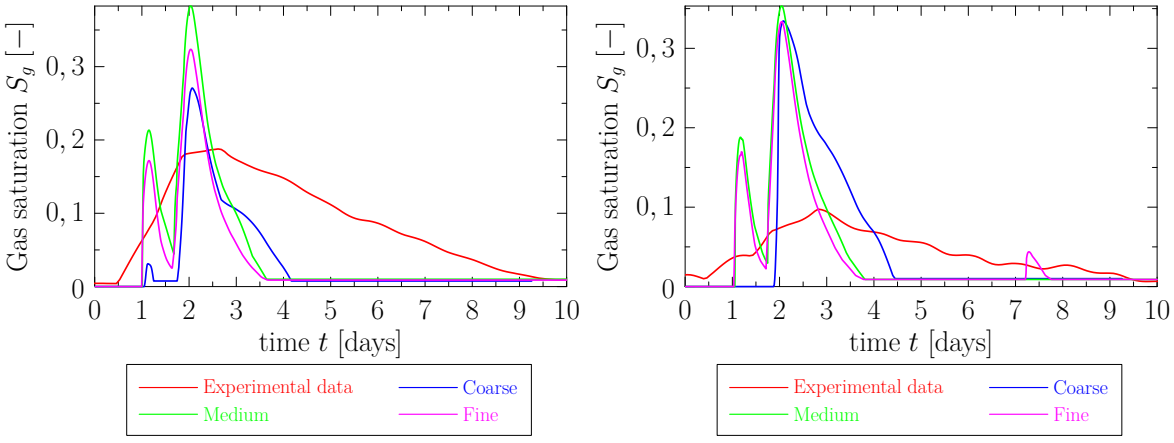
(e) Port F5.

Figure 7.24: Low contrast Experiment C described in Section 7.4, numerical results for gas saturation S_g in selected ports for various mesh resolutions and experimental data.



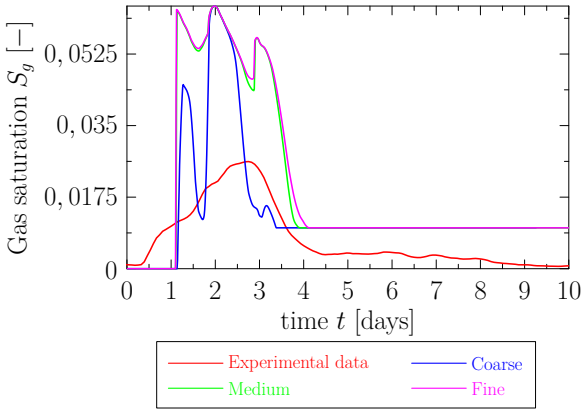
(a) Port D4.

(b) Port D6.



(c) Port E4.

(d) Port F3.



(e) Port F5.

Figure 7.25: High contrast experiment C described in Section 7.4, numerical results for gas saturation S_g in selected ports for various mesh resolutions and experimental data.

kinetic mass transfer model was able to reproduce the near-equilibrium behavior observed in small-scale Experiment A and quasi-1D Experiment B as well as the rate-limited exsolution and dissolution processes observed in Experiment C. In addition, the numerical results presented here indicate that in the larger scale Experiment C the dynamics of the exsolution and dissolution processes differ, and the computations using the kinetic model can be used to quantify this difference. Although the study of the impact of temperature changes was not the original goal of the experiments used in this chapter, the model was also able to demonstrate the importance of temperature in the multiphase CO_2 evolution when compared with experimental data and it indicates that larger temperature fluctuations cannot be neglected.

In all experiments, the model captured the accumulation of the exsolved gas below the layers of fine sand and confirmed that heterogeneity plays an important role in the multiphase CO_2 evolution. The impact of even this simple one-layer heterogeneity in the experiment suggests the need for future study of these effects on more complex heterogeneous systems that can be observed in the shallow subsurface.

The difference between the quasi-1D and 2D cases where the equilibrium and rate-limited approaches, respectively, resulted in the best match with experimental data can be possibly attributed to the different flow rates of water through the partially gas-occupied regions. However, the velocity profiles shown in Figures 7.5, 7.15, and 7.23 indicate that the velocity magnitude would not be the key factor responsible for the fundamentally different results among the experiments. The flow velocity magnitude in the vicinity of the injection ports and the other zones of the tank are comparable.

In the quasi-1D column experiment, water is forced through the gas accumulation, and the dynamic inter-phase contact is therefore likely fast enough that equilibrium mass transfer is applicable. In the 2D case, however, the water flow paths can more easily avoid the gas phase, meaning slower contact with the gas phase, and thus slower dissolution that was observed in Experiment C. The disparity between the numerical results and experimental data when the equilibrium approach was employed for Experiment C was more apparent in the ports further from the injection port. Although the two-dimensional flow field was created in Experiment A, the size of the tank, and short distances between the injection port and gas saturation sensors did not allow for observation of such effects for this experiment. Further investigation and experimental data for more configurations are needed to rigorously test these findings.

Moreover, in this chapter, we focused on quasi-1D and 2D cases only. The results showing fundamental differences between these two cases indicate that an analogous study in 3D seems to be necessary for understanding the gas evolution in the 3D laboratory and also field-scale scenarios.

The computations on finer meshes than the one used for the parameter sensitivity study justify the selected mesh resolution: the results using finer meshes do not differ significantly from those obtained using rather coarse mesh and indicate the numerical convergence of the solution when the mesh is refined. The computations on these finer meshes demonstrated the capabilities of the parallel implementation of the numerical scheme and showed the advantage of reusing the preconditioner in the solution of unsteady problems suggested and tested for simpler benchmark problem in Chapter 6.

The results presented in this chapter were published in [46, 103].

Tracer transport in heterogeneous porous media 8

One of the possible locations for the CO_2 sequestrations are deep aquifers originally occupied by brine. Once the sequestration starts, the supercritical CO_2 is injected through one or more wells which results in a significant increase in the pressure in the aquifer and overlying layers that reach much further from the injection region than the head of the injected supercritical CO_2. In addition, this pressure buildup is almost immediate in comparison with the flow and transport of CO_2 in porous materials forming the aquifer [13, 14]

This increased pressure can result in mechanical changes in the confining layer of the aquifer. Existing faults can open and form permeable pathways between the aquifer and the overlying layers allowing for leakage of the residing brine. It is also possible for new faults and fractures to appear. Due to the increased pressure in the aquifer, the brine originally residing in the aquifer tends to migrate to the surface [95, 96] where due to its high salinity can pose a risk of contaminating freshwater sources in the shallower layers [52, 94].

In the field, due to the scale of the considered problem, there are uncertainties about the subsurface material properties and the exact geological settings of the aquifer and overlying layers. In the laboratory experiments, the original scenario of brine leakage is considered on a much smaller scale. The main advantage of this approach is that the material properties are known and the high-resolution network of sensors can be installed to obtain more information than would be possible in the field.

In this chapter, we present the mathematical model describing the experiment and its simplifications, that are possible due to low tracer concentrations, to speed up the computations. We demonstrate that the simplifications are justifiable and present the results of the mathematical model for the scenarios considered in the laboratory experiments. In addition, we investigate the impact of uncertainties in the parameters of the site on the plume propagation prediction.

8.1 Experiments and goals of the study

A detailed description of the experiment is given in [4], here we summarize the features important for the numerical simulations presented in this chapter. As illustrated in Figure 8.1, the tank is packed in a highly heterogeneous configuration mimicking the geology of the field site and is divided into three zones representing the following regions:

- The first zone represents the deep aquifer selected for the CO_2 sequestration. The flow in this region is considered predominantly horizontal as the aquifer is sealed from the top with an originally impermeable confining layer.

- The second zone represents the geological layers overlying the aquifer. The flow field in this region is considered predominantly vertical as a result of increased pressure in the deep aquifer caused by the injected supercritical CO_2.
- These two zones are separated by an inclined impermeable wall with drilled holes filled with gravel representing the confining layer and fractures through which the residing brine leaks from the aquifer.
- The third zone represents the shallow subsurface with a complex flow field for example affected by wells, precipitation, and atmospheric effects.

The experiment allows for creating various flow patterns in the tank by modifying the inflow and outflow boundary conditions. In addition, the six holes in the wall separating the first and second zones allow for studying the brine leakage through different fractures under the selected flow regime controlled by the boundary conditions.

The sodium bromide (NaBr) is selected as a tracer in these experiments. The bromide ions are initially not present in the water used in the experiments and even very low concentrations can be detected by ion chromatography allowing for accurate tracking of the leaked plume. To help visualize the plume location, the fluorescein dye (ACROS-Organics 17,324-5000) was added to the water with dissolved NaBr. The limits on the range of feasible tracer concentration arise from the design of the experiment. Instead of mimicking the vertical orientation of the field site, the experiment is carried out in the horizontal orientation as shown in Figure 8.1. Therefore, quite low tracer concentrations are needed to eliminate the gravitational effects such as, in particular, the sinking of the tracer plume that is caused by the higher density of water with dissolved sodium bromide. For details, see [4].

The position and shape of the sand regions in the tank and the notation of the parts of the boundary where the various inflow and outflow conditions can be set up are shown in Figure 8.1. The tracer injection ports numbered one to six from top to bottom are located 5 cm in front of the corresponding fracture. The diameter of the injection port is 5 mm.

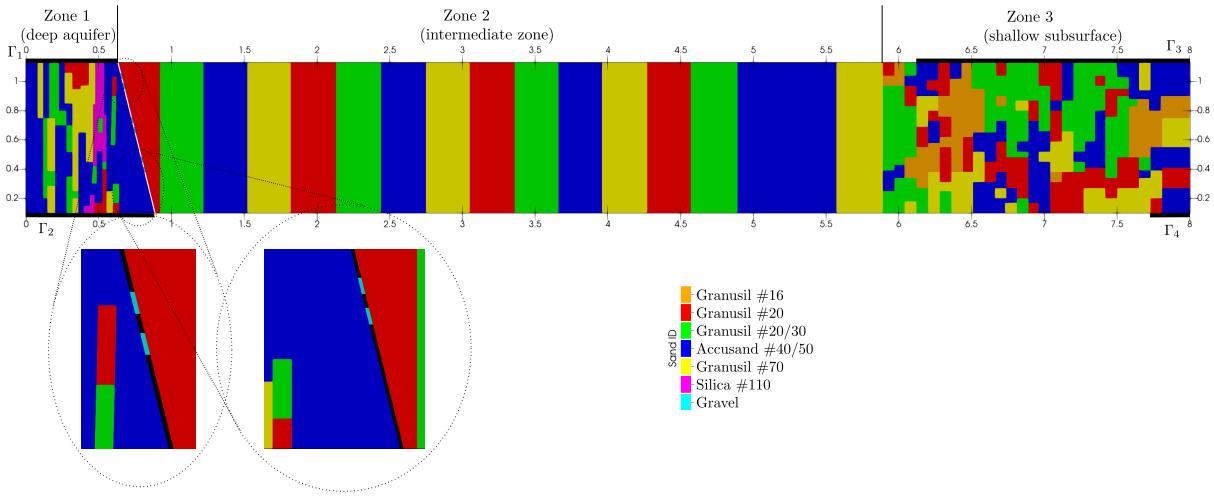


Figure 8.1: Setup of the experiment described in Section 8.1.

The hydraulic properties of the sands used in this experiment are given in Table 8.1.

Similarly to [4], three configurations are considered to represent the three experiments carried out using this tank. The experiments differ in the flow field within the tank controlled by the constant head devices with selected hydraulic heads ψ [cm] on the inflow and outflow sides of the tank or by a pump connected to one of the tank outflows with a selected pumping rate

Sand	ϕ [–]	K [m^2]
Granusil #16	0.397	$5.01 \cdot 10^{-10}$
Granusil #20	0.410	$1.91 \cdot 10^{-10}$
Granusil #20/30	0.406	$1.07 \cdot 10^{-10}$
Accusand #40/50	0.334	$4.72 \cdot 10^{-11}$
Granusil #70	0.413	$1.27 \cdot 10^{-11}$
Silica #110	0.340	$5.79 \cdot 10^{-12}$
Gravel	0.330	$1.00 \cdot 10^{-8}$

Table 8.1: Sand properties for experiments in Chapter 8. Adapted form [4].

Fluid	ρ $[\mathrm{kg\,m^{-3}}]$	μ $[\mathrm{Pa\,s^{-1}}]$	M $\left[\mathrm{kg\,mol^{-1}}\right]$
Water (H_2O)	997.78	$9.72 \cdot 10^{-4}$	$1.802 \cdot 10^{4}$

Table 8.2: Fluid properties for the experiments in Chapters 7 and 8.

Tracer	ρ $[\mathrm{kg\,m^{-3}}]$	M $\left[\mathrm{kg\,mol^{-1}}\right]$
Sodium bromide (NaBr)	2180	$1.029 \cdot 10^{5}$

Table 8.3: Tracer properties for the experiment Chapter 8.

q $[\mathrm{ml\,min^{-1}}]$. In addition to the flow field, the experiments differ in the position of the injection port, tracer injection rate, and its concentration. The settings of these three experiments are summarized in Table 8.4.

The main goal of this study is to show the capabilities of the numerical model and its implementation to capture the tracer transport in heterogeneous porous media and to obtain results comparable to those presented in [4] where the numerical solution was obtained by `FEFLOW` [30].

8.2 Problem formulation

The primary unknown for the single-phase flow equation is pressure: $Z_1 = p$ and the primary unknown for the component transport is the mass fraction of the dissolved tracer: $Z_2 = X$. Similarly to Section 7.1, we for simplicity omit the subscript and superscript of X as only one phase and one component are tracked here. For this choice of primary variables, the problem is

	Experiment		
Boundary condition	1	2	3
Γ_1	$\psi = 165.44\,\mathrm{cm}$	$\psi = 165.44\,\mathrm{cm}$	$\psi = 165.10\,\mathrm{cm}$
Γ_2	$\psi = 164.47\,\mathrm{cm}$	$\psi = 164.47\,\mathrm{cm}$	$q = 0\,\mathrm{ml\,min^{-1}}$
Γ_3	$\psi = 161.15\,\mathrm{cm}$	$\psi = 161.15\,\mathrm{cm}$	$q = 8.3\,\mathrm{ml\,min^{-1}}$
Γ_4	$\psi = 150.50\,\mathrm{cm}$	$\psi = 150.50\,\mathrm{cm}$	$\psi = 157.90\,\mathrm{cm}$
Settings			
Active fracture	3	1	6
Injection rate [$\mathrm{ml\ min^{-1}}$]	7.75	7.90	7.90
Tracer concentration [$\mathrm{mg\,l^{-1}}$]	42.40	54.45	85.00
Experiment duration [days]	6	4.5	7

Table 8.4: Configuration of the three experiments described in Section 8.1, the following notation was used: ψ is the hydraulic head, q is the volumetric flow rate.

represented by the following non-zero coefficients in (3.1)

$$\mathcal{N} = \begin{pmatrix} 0 & \Phi\frac{d\rho}{dX} \\ 0 & \Phi\rho \end{pmatrix}, \mathcal{U} = \begin{pmatrix} 0 & 0 \\ 0 & \rho\boldsymbol{u} \end{pmatrix}, \mathcal{M} = \begin{pmatrix} \frac{\rho}{\mu} \\ \rho \end{pmatrix}, \mathcal{D} = \begin{pmatrix} \mathbb{K} & 0 \\ 0 & \tau\Phi D \end{pmatrix},$$
$$\mathcal{W} = \begin{pmatrix} \rho\mathbb{K}\boldsymbol{g} \\ \mathbf{0} \end{pmatrix}, \mathcal{F} = \begin{pmatrix} f \\ f_X - Xf \end{pmatrix}. \quad (8.1)$$

In general, the water density is a function of dissolved tracer mass fraction X given by

$$\rho(X) = \left(\frac{1-X}{\rho_w} + \frac{X}{\rho_X}\right)^{-1}, \quad (8.2)$$

where ρ_w is a density of water and ρ_X is density of tracer (NaBr), values are provided in Tables 8.2 and 8.3, respectively. However the amount of dissolved tracer considered in the experiments investigated in this chapter is very low, therefore, we assume that the term $\frac{d\rho}{dX}$ in coefficient $\mathcal{N}$ in the formulation of the problem (8.1) can be neglected and a simplified problem with $\frac{d\rho}{dX} = 0$ can be solved instead of the original one similarly to [4].

Under the assumption of $\frac{d\rho}{dX} = 0$, the equation for Z_1 represented by (8.1) is a steady one and the original problem can be split into two simpler ones. First, the equation for Z_1 is solved. As a result, we obtain pressure and velocity fields that, under the assumption of constant density, do not change over time. This problem is further referred to as the pressure equation. Then only the unsteady equation for Z_2 is solved at each time step using the obtained velocity field. This problem is further referred to as the transport equation. To confirm that this simplification does not affect the numerical results and to demonstrate the reduction in computational time, we compare both approaches on a selected artificial scenario adapted from [3] in Section 8.4.

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt [s]
1	$1.38 \cdot 10^{-2}$	34 956	52 784	105 568	60

Table 8.5: Mesh properties used to investigate the density effects in Section 8.4. Note that the number of degrees of freedom is given for the most general formulation of the problem with variable density.

8.3 Parallel computation setup

All the computations presented in this chapter were carried out using the parallel implementation of the numerical scheme presented in Chapter 4. As the problems in highly heterogeneous porous media are solved, the ML variant of the numerical scheme is used. For the most general formulation of the problem described in Section 8.2 (with variable density), matrix $\mathbb{A}$ of the linear problem is updated at each time step and, therefore, the strategy proposed to solve unsteady problems in Section 6.4.3 was employed: the solution from the previous time step was used as the initial vector for the BiCGStab iterations at the current time step and the preconditioner was reused until the trigger value of 10 iterations was exceeded.

The solution strategy is further adapted for the simplified case of constant density. For the solution of the pressure equation, the approach described in Section 6.3 is employed. For the solution of the unsteady transport equation, the approach described in Section 6.4 is further tailored. In this case, matrix $\mathbb{A}$ of the linear problem is constant. Therefore, the whole setup of the BDDC method including the LU factorizations of the local matrices $\mathbb{A}_i$ needed to compute the action of the Schur complement matrix $\mathbb{S}$ is done at the first time step only. Overall, only the right-hand side of the linear problem is updated at each time step.

The computations were run on the `Helios` cluster at the Czech Technical University in Prague, Czech Republic. The computational nodes of `Helios` are equipped with two 16-core AMD EPYC 7281, 2.1GHz processors, and 128 GB RAM. All the computational meshes used in this chapter were generated by `Gmsh` [49].

8.4 Variable density effects

To verify the negligible effect of the tracer concentrations used in the experiments on the water density, we investigate the artificial problem adapted from [3]. The configuration of the problem is illustrated in Figure 8.2. We consider a three-layer horizontal system with the same porous materials that were used for the laboratory experiment. The tracer is injected through the port with diameter 0.5 mm on the left-hand side of the first region with a constant injection rate $2\ \mathrm{ml\,min^{-1}}$ and the tracer concentration values C of 50, 100, 500, and 1000 $\mathrm{mg\,l^{-1}}$ are tested. The injection into the domain is in the model represented by a source term in the area of the port.

The top and bottom boundaries of the domain are impermeable walls with zero Neumann boundary conditions for both primary unknowns. On the left-hand side vertical boundary, $X = 0$ and the hydrostatic profile for p corresponding to hydraulic head of $\psi = 120$ cm are prescribed. On the right-hand side vertical boundary, $\nabla X \cdot \boldsymbol{n} = 0$ and the hydrostatic profile for p corresponding to hydraulic head of $\psi = 115$ cm are prescribed.

The computations are run on a triangular mesh with parameters listed in Table 8.5 divided into four subdomains. The fixed time step $\Delta t = 60$ s is selected and the results are shown at time $t_{fin} = 30$ days.

To study the impact of the tracer concentration on the water density and the resulting plume

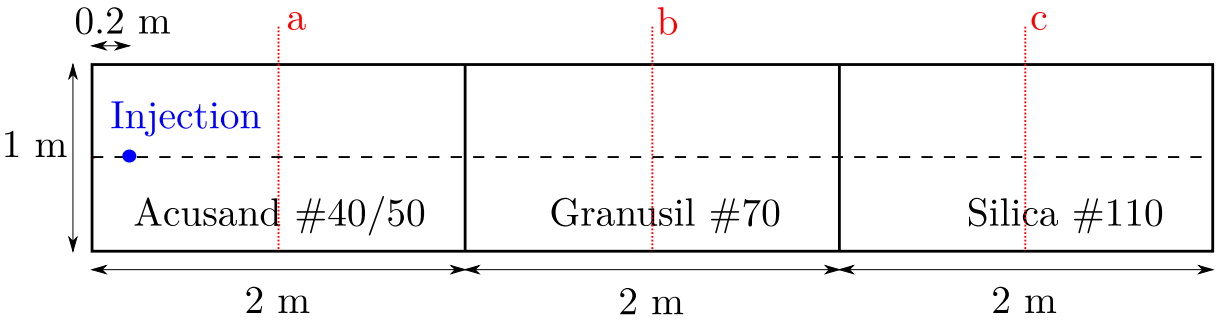


Figure 8.2: Setup of the artificial scenario to investigate the density effects in Section 8.4.

shape and location, the numerical results in the three vertical cross-sections through the centers of the sand regions: a,b,c in Figure 8.2 denoted by the red lines are compared. The results for both constant and variable water density are shown in Figure 8.3 for the four considered tracer concentrations. Under the assumption of constant density, the concentration profiles are symmetric along the horizontal axis of the domain as is expected for this problem configuration. For the variable density and the tracer concentrations of $50\ \mathrm{mg\,l^{-1}}$ and $100\ \mathrm{mg\,l^{-1}}$, the vertical profiles are almost the same as for the constant density. This justifies the assumption of constant density for similar settings in the experiment. When the tracer concentration increases, the impact on the location of the plume is more significant. The density of water containing tracer is higher than the density of the clean water and the plume is sinking towards the bottom boundary of the domain.

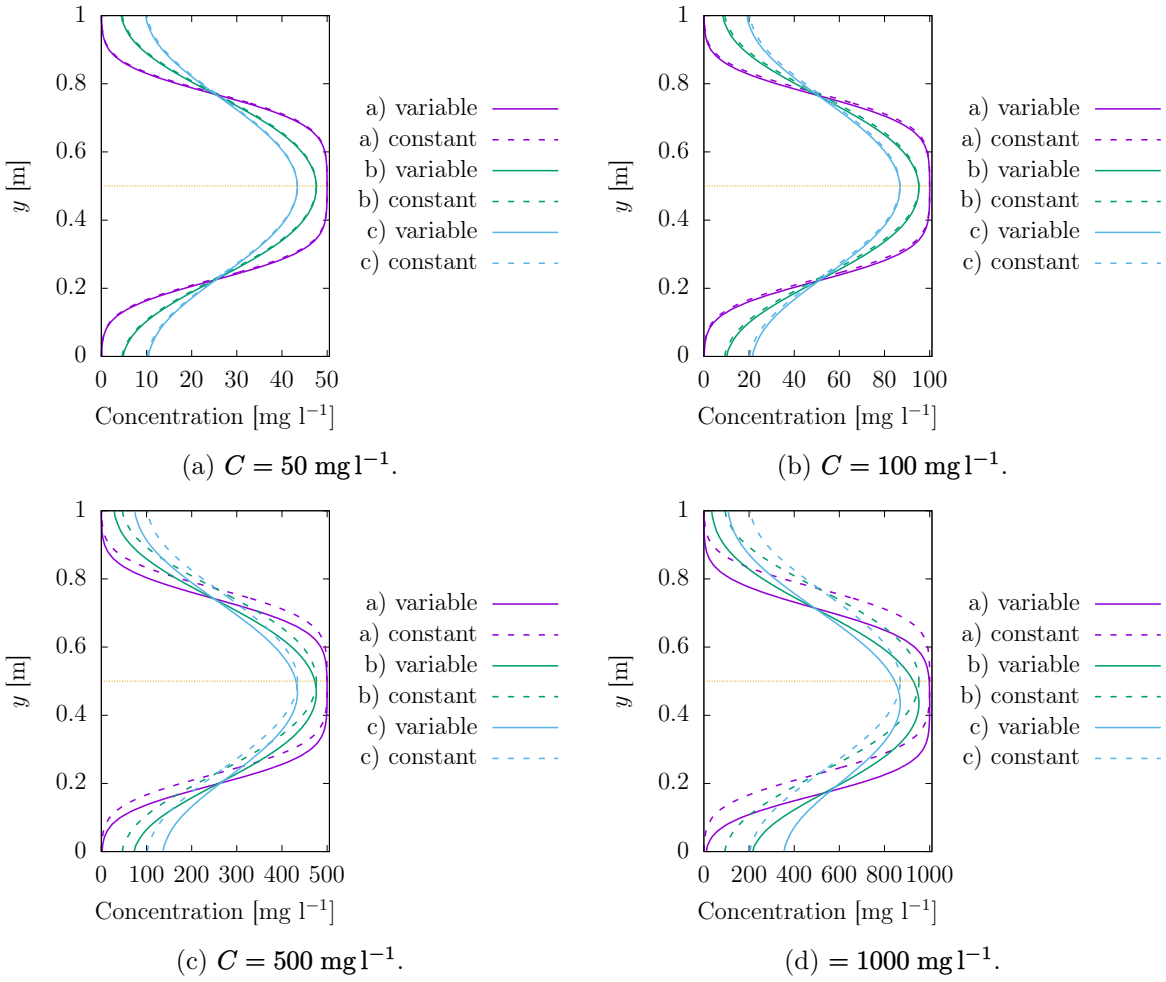
To quantify the error caused by neglecting the effect of the tracer concentration on water density, we compute the y-coordinate of the center of mass of the plume along the cross-sections. In the case of constant density, its value is $0.5\ \mathrm{m}$ for all tracer concentrations. The error of constant density approach E_C for tracer concentration C is then given by

$$E_C = \frac{|0.5 - y_c|}{0.5} \cdot 100\%, \tag{8.3}$$

where y_C is the coordinate of the center of mass for tracer concentration C. For the variable density, the coordinates of the center of mass and the errors of the constant density approach are shown in Table 8.6. The results show that for a given tracer concentration, the errors are very similar for all the cross-sections. For the concentration of $100\ \mathrm{mg\,l^{-1}}$, the errors are lower than 1.5 % which is acceptable with respect to the accuracy of the instrumentation used in the experiments.

The computation times vary only slightly between the various tracer concentrations for both approaches. The computation with the varying density is almost seven times more computationally demanding, than the computation with constant density with the computational times of 2 hours 41 minutes and of 24 minutes, respectively.

The total number of iterations and the number of preconditioner computations are similar for both constant and varying densities. In the case of varying density, the preconditioner is computed at the first time step only and the linear solver converges in 4 BiCGStab iterations. In the following time steps, the preconditioner from the first time step is used and the linear solver converges in 1 iteration. In the case of constant density, the linear solver for the pressure equation converges in 4 BiCGStab iterations. At the first time step of the transport problem, the preconditioner is computed and the linear solver converges in 4 BiCGStab iterations. At the following time steps, this preconditioner is reused and the linear solver converges in 1 iteration.



(a) $C = 50\ \mathrm{mg\,l^{-1}}$. (b) $C = 100\ \mathrm{mg\,l^{-1}}$.

(c) $C = 500\ \mathrm{mg\,l^{-1}}$. (d) $= 1000\ \mathrm{mg\,l^{-1}}$.

Figure 8.3: Concentration profiles for variable (solid lines) and constant (dashed lines) densities along vertical cross-sections for selected tracer concentrations in the artificial scenario described in Section 8.4.

Tracer concentration [mg l^{-1}]	a		b		c	
	y_C [m]	Error	y_C [m]	Error	y_C [m]	Error
50	0.497	0.62 %	0.497	0.63 %	0.497	0.60 %
100	0.494	1.24 %	0.494	1.27 %	0.494	1.21%
500	0.469	6.20 %	0.469	6.61 %	0.470	6.38 %
1000	0.441	11.87 %	0.440	12.09 %	0.442	11.58 %

Table 8.6: Center of mass along the cross-sections and the error of the constant density approach for the artificial problem described in Section 8.4.

Mesh ID	h [m]	# Elements	# Sides	# DOF	Δt [s]
1	$2.89 \cdot 10^{-3}$	1 253 468	1 882 852	1 882 852	2.5

Table 8.7: Mesh properties used to compute the scenarios described in Section 8.1. Note that the number of degrees of freedom is given for the split pressure and transport problems.

8.5 Experiments

The computation for all three experiment settings described in Section 8.1 are run in parallel on the mesh with properties given in Table 8.7. The injection into the domain is in the model represented by the source term in the area of the port.

The tracer concentrations are low enough for the constant density approach to be valid and the problem is decomposed into the pressure and transport problems as described in Section 8.4. The mesh is divided into 32 subdomains resulting in approximately 59 thousand degrees of freedom per subdomain for both pressure and transport equations. Computations of the transport problem for all three experiments are run with a fixed time step $\Delta t = 2.5$ s.

The shape and location of the tracer plume at a selected times obtained by using the material properties listed in Table 8.1 are shown in Figures 8.5, 8.6, and 8.7 for Experiments 1,2, and 3, respectively. However, these results differ significantly from the tracer plumes observed in the experiments reported in [4]. These differences can be attributed to the uncertainties in the material properties and the settings of the experiments. Moreover, the properties of certain features of the experiment can change between the time of packing the tank with sand and running the experiment [4].

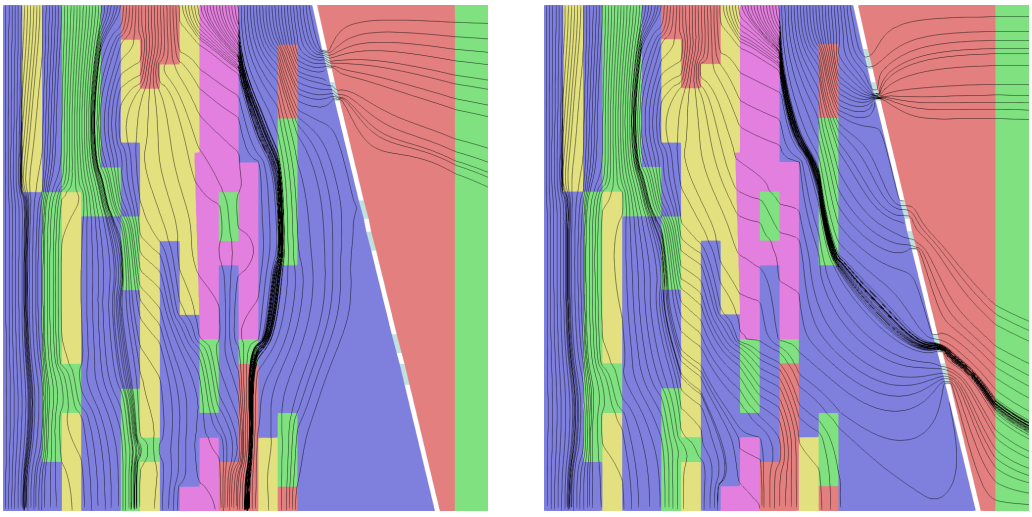
Originally, the holes in the impermeable layer representing the fractures in the confining layer of the aquifer were filled with gravel resulting in high permeability. However, sand before the fracture can be pushed into the fracture by the flowing water partially clogging it and reducing its permeability. This effect is more significant in the upper fractures where higher water flow velocities were observed. The changes in the fracture permeability have a significant effect on the flow field in the vicinity of the fractures and the confining layer. The impact on the flow field is more significant in the experiments with a hydraulic head set up also on the boundary Γ_2 which allows water outflow. The visualization of the flow field from the top boundary Γ_1 through the set of fractures for the high permeable and partially clogged fractures are for Experiment 1 shown in Figure 8.4.

In addition to the fracture clogging, there is a discrepancy between the permeability measured using one-dimensional column laboratory testing and the permeability of the sand packed in the larger two-dimensional tank [7]. Moreover, small heterogeneity affecting the flow field in the tank can arise from the packing of the neighboring regions of different sands and partial mixing of these two materials at the interface. With the available equipment, we cannot quantify this mixing of porous materials and, therefore, these heterogeneities at the interfaces between the sand regions are not represented in the model.

To match the experiments, the material properties were calibrated using the measured experimental data. A detailed description of the calibration procedure is provided in [4].

The results for these modified settings are shown in Figures 8.5, 8.6, and 8.7 for Experiments 1,2, and 3, respectively, where significant differences in the shape and location of the plume can be observed. The results in these figures are in agreement with the shapes and location of the plumes observed in [4].

The total number of iterations for all three experiments is shown in Table 8.8. Because of the selected strategy for the numerical solution, we separately show the number of iterations for the



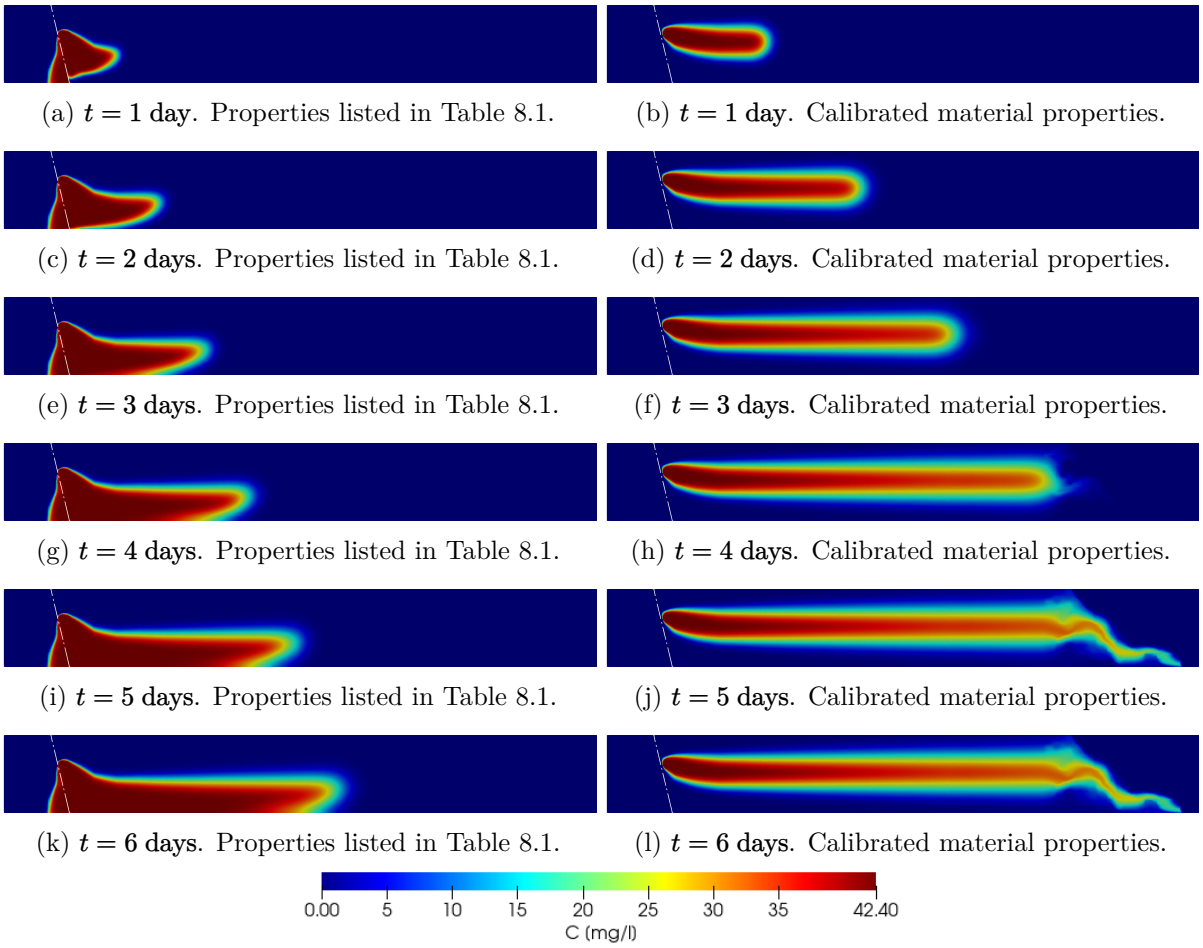
(a) High-permeable fractures. (b) Clogged fractures.

Figure 8.4: Streamlines from the top boundary Γ_1. Impact of the fractures clogging on the flow field for Experiment 1 described in Section 8.1. Color scale corresponds to the sand type, see Figure 8.1.

	Experiment					
	1		2		3	
Material properties	Measured	Calibrated	Measured	Calibrated	Measured	Calibrated
Steady pressure equation						
# Iterations	55	48	53	46	57	51
Time [s]	3.6	3.5	3.6	3.5	3.6	3.5
Unsteady transport equation						
Avg. # iterations	1.75	1.74	1.95	1.97	1.96	1.86
Min. # iterations	1	1	1	1	1	1
Max. # iterations	3	3	3	3	3	3
Time [h]	14.5	14.3	11.3	11.0	17.4	17.0

Table 8.8: Number of iterations and the computational times for experiments described in Section 8.1.

pressure and transport problems. The results in Table 8.8 show that the number of iterations for the pressure problem is significantly higher than those for the transport problems, where the BiCGStab solver converges on average in less than two iterations. The higher number of iterations for the steady pressure problem in comparison with benchmark problems in Chapter 6 can be attributed to highly varying material properties within the computational domain, in particular sand permeability. Overall, the changes in the material properties (before and after calibration), boundary conditions, tracer concentration, and injection rate have a small effect on the number of iterations. The difference in the overall computational time is caused mainly by the different duration of the experiments.



(a) $t = 1$ day. Properties listed in Table 8.1. (b) $t = 1$ day. Calibrated material properties.

(c) $t = 2$ days. Properties listed in Table 8.1. (d) $t = 2$ days. Calibrated material properties.

(e) $t = 3$ days. Properties listed in Table 8.1. (f) $t = 3$ days. Calibrated material properties.

(g) $t = 4$ days. Properties listed in Table 8.1. (h) $t = 4$ days. Calibrated material properties.

(i) $t = 5$ days. Properties listed in Table 8.1. (j) $t = 5$ days. Calibrated material properties.

(k) $t = 6$ days. Properties listed in Table 8.1. (l) $t = 6$ days. Calibrated material properties.

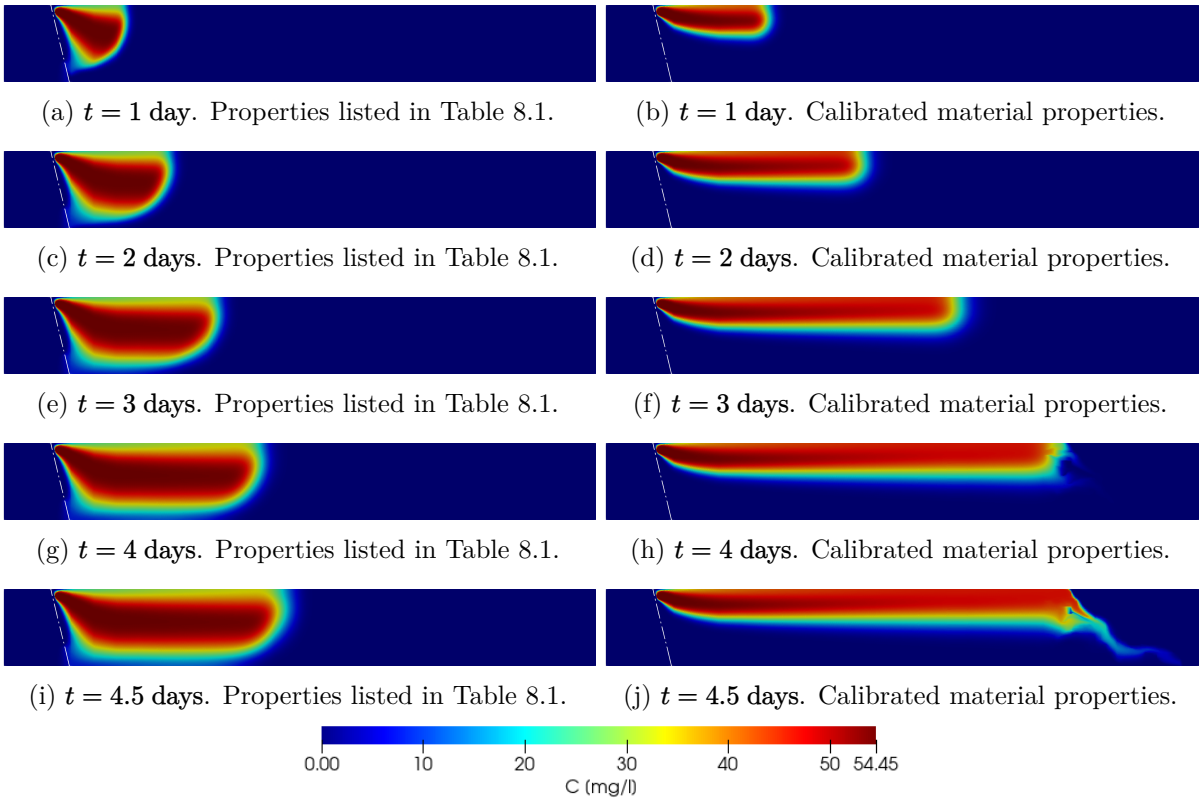
Figure 8.5: Numerical results of tracer concentration for material properties given in Table 8.1 and material properties calibrated to experimental data in Experiment 1 described in Section 8.1.

8.6 Chapter summary

In this chapter, the application of the numerical scheme to the transport in highly heterogeneous porous media was tested on a laboratory experiment mimicking the brine leakage from a deep aquifer.

Firstly, in an artificial scenario, we verified that the dependency of the water density on the tracer concentration can be neglected for the considered range of concentrations. This reduces the complexity of the problem as the problem can be decomposed into steady pressure and unsteady transport effectively halving the size of the linear systems. In addition, the matrix arising from the discretization of the transport problem is constant over time.

On the settings representing the laboratory experiments, we demonstrated the impact of uncertainties in the material properties and settings of the experiments on the results. The clogging of the holes in the impermeable layers representing the fractures in the confining layer of the aquifer resulting in varying permeability of the fractures significantly affects the flow field in the vicinity of the confining layers and the fractures. In addition, the differences in the hydraulic conductivity of the sands measured in the one-dimensional column and packed in the 2D tank affect the location and shape of the tracer plume. The results presented in this chapter obtained using the MHFEM numerical scheme described in Chapter 3 are in agreement with [4] where `FEFLOW` [30] was used.



(a) $t = 1$ day. Properties listed in Table 8.1.

(b) $t = 1$ day. Calibrated material properties.

(c) $t = 2$ days. Properties listed in Table 8.1.

(d) $t = 2$ days. Calibrated material properties.

(e) $t = 3$ days. Properties listed in Table 8.1.

(f) $t = 3$ days. Calibrated material properties.

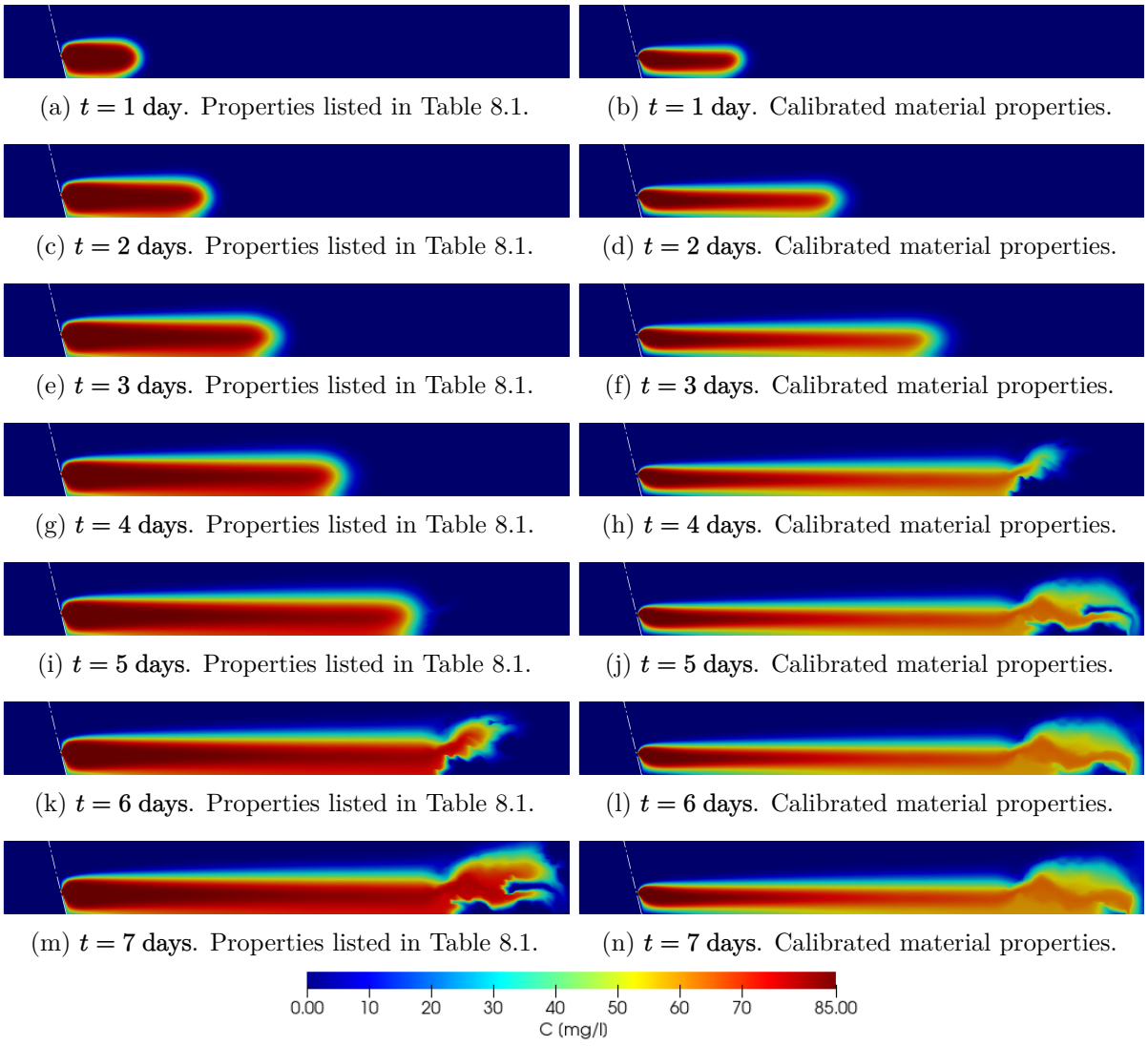
(g) $t = 4$ days. Properties listed in Table 8.1.

(h) $t = 4$ days. Calibrated material properties.

(i) $t = 4.5$ days. Properties listed in Table 8.1.

(j) $t = 4.5$ days. Calibrated material properties.

Figure 8.6: Numerical results of tracer concentration for material properties given in Table 8.1 and material properties calibrated to experimental data in Experiment 2 described in Section 8.1.



(a) $t = 1$ day. Properties listed in Table 8.1.

(b) $t = 1$ day. Calibrated material properties.

(c) $t = 2$ days. Properties listed in Table 8.1.

(d) $t = 2$ days. Calibrated material properties.

(e) $t = 3$ days. Properties listed in Table 8.1.

(f) $t = 3$ days. Calibrated material properties.

(g) $t = 4$ days. Properties listed in Table 8.1.

(h) $t = 4$ days. Calibrated material properties.

(i) $t = 5$ days. Properties listed in Table 8.1.

(j) $t = 5$ days. Calibrated material properties.

(k) $t = 6$ days. Properties listed in Table 8.1.

(l) $t = 6$ days. Calibrated material properties.

(m) $t = 7$ days. Properties listed in Table 8.1.

(n) $t = 7$ days. Calibrated material properties.

Figure 8.7: Numerical results of tracer concentration for material properties given in Table 8.1 and material properties calibrated to experimental data in Experiment 3 described in Section 8.1.

Conclusion 9

We formulated the mathematical model describing the two-phase compositional flow in porous media with kinetic interphase mass transfer. We proposed and implemented a suitable numerical method for solving these problems and used it to investigate the processes in porous media. A detailed description of the results is provided in the following sections.

9.1 Numerical solution and implementation

The problems solved in this work range from a steady Laplace problem to a two-phase compositional flow with mass transfer. Therefore, the numerical scheme based on the MHFEM spatial discretization designed for solving the general system of coupled non-linear advection-diffusion-reaction equations was proposed in Chapter 3. In this form, all the problems considered in this work can be represented and this approach allows for separating the formulation of the problem, such as governing equations and boundary conditions, from the implementation of the numerical solver. Two variants of the scheme are considered based on the method used to compute the local coefficients on each element: exact integration and the mass lumping technique.

In the case of a homogeneous porous medium, the convergence of the method is verified on a benchmark problem with a known semi-analytical solution showing that both variants of the method are of the first order of accuracy with only slight differences in the errors. In the case of a heterogeneous porous medium, the exact integration leads to non-physical oscillations at the material interfaces and the mass lumping variant of the method is used to eliminate this effect.

The method is implemented both in serial and in parallel. The parallel approach is based on the BDDC method. The scaling tests of the implementation show a good parallel efficiency up to a thousand CPU cores. The limitation on the number of CPU cores that can be used efficiently arises from the selected two-level BDDC approach and can be avoided by employing the multi-level BDDC method which is beyond the scope of this work.

To speed up the solution of the transient problems using the BDDC method, we propose to use the solution from the previous time step as the initial vector for the linear solver in the current time step and reuse the BDDC preconditioner for several time steps until the number of linear solver iterations exceeds a given trigger value. For the benchmark problems, this leads up to a 40 % reduction in the computational time.

9.2 Carbon dioxide evolution in shallow subsurface

The developed numerical method was employed in the investigation of the mass transfer models in the two-phase systems consisting of water and CO_2 in the shallow subsurface. We used the

data from three laboratory experiments that focused on multiphase CO_2 evolution under various conditions. In the case of quasi-1D column experiments, the simpler local equilibrium model gives results that match the experimental data which is in agreement with the previous work of [90] for similar settings. However, for the more complex two-dimensional flow field, the equilibrium model leads to over-prediction of gas exsolution and the kinetic model is necessary to match the experimental data.

In addition, the results tend to confirm our hypothesis that the difference in the dynamics of the exsolution and dissolution processes plays a significant role and needs to be represented in the model by selecting different mass transfer coefficients for the corresponding processes to get an agreement with the gas evolution observed in the experiment. Although the experiments were designed as isothermal, the problems with air conditioning in the laboratory allowed us to study the impact of temperature on the multiphase CO_2 evolution. The results indicate that temperature plays an important role and even the temperature fluctuations in the range of 10°C cannot be neglected.

We thoroughly investigated the mass transfer processes in the setting of the three experiments considered in this work. However, many processes that were not the main driving mechanisms in the studied scenarios were simplified, for example, the dependence of the physical quantities on temperature, or neglected, for example, the mechanical dispersion.

9.3 Tracer transport

The numerical scheme was used for the computations of the single-phase tracer transport problems in a highly heterogeneous porous medium. These scenarios were inspired by the problem of brine leakage from a deep aquifer mimicked on the laboratory scale. Firstly, in the artificial scenario, we investigated the impact of the tracer concentration on the water density and the possibility to simplify the governing equations of the problem by assuming the water density independent of the tracer concentration. The numerical results justify this assumption for the concentrations used in the experiments allowing for a significant speedup of the computations by decoupling the pressure and transport equations. The results for the settings of the laboratory experiments demonstrate the capabilities of the numerical scheme presented in this work to solve problems in highly heterogeneous porous media. The shape and the location of the tracer plume are in agreement with the experiments and the numerical results obtained by `FEFLOW` presented in [4]. Moreover, we demonstrated the impact of the uncertainties in both material properties and settings of the experiments on the numerical results.

9.4 Future work

The limitations of the presented numerical method are the first order of accuracy only and the performance of the parallel implementation is limited by the selected two-level BDDC approach. The future goal is to design a method of a higher order of accuracy and improve the parallel performance for a higher number of CPU cores by implementing the multi-level BDDC method.

In the current mathematical model, some of the processes are considered in the simplified form only or neglected completely. We plan to expand the mathematical model to comprise these processes and use the data from additional experiments to study their impact on the evolution of multi-phase compositional systems in the shallow subsurface. In this work, we demonstrated the fundamental difference in the mass transfer and transport processes between the quasi-1D and 2D cases. Hence, a question arises about the nature of these processes in 3D and their relation

to the simpler 2D and 1D cases. The results presented here indicate that such a study seems to be necessary for understanding the gas evolution in 3D on both laboratory and field scales.

Bibliography

A

[1] Abramowitz, M. and Stegun, I. A. (1972). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. National Bureau of Standards Applied Mathematics Series 55. Tenth Printing.* ERIC.

[2] van Antwerp, D. J., Falta, R. W., and Gierke, J. S. (2008). Numerical Simulation of Field-Scale Contaminant Mass Transfer during Air Sparging. *Vadose Zone Journal*, 7(1):294–304.

[3] Askar, A. H. (2021). *Modeling, Monitoring, and Controlling Tools for Risk Management of Shallow Aquifer Contamination by Far-Field Brine Leakage from CO2 Storage Formations: Intermediate-scale Laboratory Testing.* PhD thesis, Colorado School of Mines.

[4] Askar, A. H., Illangasekare, T. H., Trautz, A., Solovský, J., Zhang, Y., and Fučík, R. (2021). Exploring the Impacts of Source Condition Uncertainties on Far-Field Brine Leakage Plume Predictions in Geologic Storage of CO2: Integrating Intermediate-Scale Laboratory Testing With Numerical Modeling. *Water Resources Research*, 57(9):e2021WR029679.

B

[5] Badia, S., Martín, A. F., and Principe, J. (2016). Multilevel Balancing Domain Decomposition at Extreme Scales. *SIAM Journal on Scientific Computing*, 38(1):C22–C52.

[6] Barenblatt, G., Zheltov, I., and Kochina, I. (1960). Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks. *Journal of Applied Mathematics and Mechanics*, 24(5):1286–1303.

[7] Barth, G. R., Hill, M. C., Illangasekare, T. H., and Rajaram, H. (2001). Predictive modeling of flow and transport in a two-dimensional intermediate-scale, heterogeneous porous medium. *Water Resources Research*, 37(10):2503–2512.

[8] Bastian, P. (2000). *Numerical Computation of Multiphase Flows in Porous Media.* Habilitation thesis, Kiel university.

[9] Bastian, P., Blatt, M., Dedner, A., Dreier, N.-A., Engwer, C., Fritze, R., Gräser, C., Grüninger, C., Kempf, D., Klöfkorn, R., Ohlberger, M., and Sander, O. (2021). The Dune framework: Basic concepts and recent developments. *Computers and Mathematics with Applications*, 81:75–112. Development and Application of Open-source Software for Problems with Numerical PDEs.

[10] Bear, J. (2018). *Modeling Phenomena of Flow and Transport in Porous Media*. Springer.

[11] Bear, J. and Cheng, A. (2010). *Modeling Groundwater Flow and Contaminant Transport*, volume 23. Springer.

[12] Bear, J. and Verruijt, A. (1987). *Modeling groundwater flow and pollution*, volume 2. Springer Science & Business Media.

[13] Birkholzer, J. T., Nicot, J. P., Oldenburg, C. M., Zhou, Q., Kraemer, S., and Bandilla, K. (2011). Brine flow up a well caused by pressure perturbation from geologic carbon sequestration: Static and dynamic evaluations. *International Journal of Greenhouse Gas Control*, 5(4):850–861.

[14] Birkholzer, J. T., Zhou, Q., and Tsang, C.-F. (2009). Large-scale impact of CO2 storage in deep saline aquifers: A sensitivity study on pressure response in stratified systems. *International Journal of Greenhouse Gas Control*, 3(2):181–194.

[15] Boffi, D., Brezzi, F., and Fortin, M. (2013). *Mixed Finite Element Methods and Applications*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[16] Bramble, J. H., Pasciak, J. E., and Schatz, A. H. (1986). The construction of preconditioners for elliptic problems by substructuring. I. *Mathematics of Computation*, 47(175):103–134.

[17] Bramble, J. H., Pasciak, J. E., and Schatz, A. H. (1987). The construction of preconditioners for elliptic problems by substructuring. II. *Mathematics of Computation*, 49(179):1–16.

[18] Bramble, J. H., Pasciak, J. E., and Schatz, A. H. (1988). The construction of preconditioners for elliptic problems by substructuring. III. *Mathematics of Computation*, 51(184):415–430.

[19] Bramble, J. H., Pasciak, J. E., and Schatz, A. H. (1989). The construction of preconditioners for elliptic problems by substructuring. IV. *Mathematics of Computation*, 53(187):1–24.

[20] Brooks, R. and Corey, A. (1964). Hydraulic Properties of Porous Media. *Colorado State University, Hydrology Paper*, 3:27.

[21] Buckley, S. E. and Leverett, M. (1942). Mechanism of fluid displacement in sands. *Transactions of the AIME*, 146(01):107–116.

[22] Burdine, N. (1953). Relative Permeability Calculations from Pore Size Distribution Data. *Journal of Petroleum Technology*, 5.

C

[23] Chavent, G. and Roberts, J. (1989). A unified physical presentation of mixed, mixed-hybrid finite elements and usual finite differences for the determination of velocities in waterflow problems. *[Research Report] RR-1107, INRIA*.

[24] Chen, Y., Davis, T. A., Hager, W. W., and Rajamanickam, S. (2008). Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate. *ACM Transactions on Mathematical Software*, 35(3):22:1–22:14.

[25] Class, H., Helmig, R., and Bastian, P. (2002). Numerical simulation of non-isothermal multiphase multicomponent processes in porous media.: 1. An efficient solution technique. *Advances in Water Resources*, 25(5):533–550.

[26] Coats, K. H. (1980). An Equation of State Compositional Model. *SPE Journal*, 20(05):363–376.

[27] Concus, P., Golub, G. H., and O'Leary, D. P. (1976). A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations. In Bunch, J. R. and Rose, D. J., editors, *Sparse Matrix Computations*, pages 309–332. Academic Press.

[28] Corey, A. T. (1994). *Mechanics of immiscible fluids in porous media.* Water Resources Publication.

D

[29] Davis, T. A. (2004). Algorithm 832: UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method. *ACM Transactions on Mathematical Software*, 30(2):196–199.

[30] Diersch, H.-J. G. (2013). *FEFLOW: finite element modeling of flow, mass and heat transport in porous and fractured media.* Springer Science & Business Media.

[31] Dohrmann, C. R. (2003). A Preconditioner for Substructuring Based on Constrained Energy Minimization. *SIAM Journal of Scientific Computing*, 25:246–258.

[32] Douglas, Jr., J. (1983). Finite Difference Methods for Two-Phase Incompressible Flow in Porous Media. *SIAM Journal on Numerical Analysis*, 20(4):681–696.

[33] Dryja, M. (1989). An additive Schwarz algorithm for two-and three-dimensional finite element elliptic problems. *Domain decomposition methods*, pages 168–172.

[34] Dryja, M. and Widlund, O. B. (1990a). Chapter 16 - Some Domain Decomposition Algorithms for Elliptic Problems. In Kincaid, D. R. and Hayes, L. J., editors, *Iterative Methods for Large Linear Systems*, pages 273–291. Academic Press.

[35] Dryja, M. and Widlund, O. B. (1990b). Towards a unified theory of domain decomposition algorithms for elliptic problems. In *Third international symposium on domain decomposition methods for partial differential equations*, pages 3–21.

[36] Durlofsky, L., Efendiev, Y., and Ginting, V. (2007). An adaptive local–global multiscale finite volume element method for two-phase flow simulations. *Advances in Water Resources*, 30(3):576–588.

[37] Dziuk, G. (1994). Convergence of a semi-discrete scheme for the curve shortening flow. *Mathematical Models and Methods in Applied Sciences*, 04(04):589–606.

E

[38] Efendiev, Y., Ginting, V., Hou, T., and Ewing, R. (2006). Accurate multiscale finite element methods for two-phase flow simulations. *Journal of Computational Physics*, 220(1):155–174.

F

[39] Falta, R. W. (2000). Numerical modeling of kinetic interphase mass transfer during air sparging using a dual-media approach. *Water Resources Research*, 36(12):3391–3400.

[40] Falta, R. W. (2003). Modeling sub-grid-block-scale dense nonaqueous phase liquid (DNAPL) pool dissolution using a dual-domain approach. *Water Resources Research*, 39(12).

[41] Fučík, R. (2010). *Advanced Numerical Methods for Modelling Two-Phase Flow in Heterogeneous Porous Media*. PhD thesis, FNSPE of Czech Technical University Prague.

[42] Fučík, R., Illangasekare, T. H., and Beneš, M. (2016). Multidimensional self-similar analytical solutions of two-phase flow in porous media. *Advances in Water Resources*, 90:51–56.

[43] Fučík, R., Klinkovský, J., Solovský, J., Oberhuber, T., and Mikyška, J. (2019). Multidimensional mixed–hybrid finite element method for compositional two-phase flow in heterogeneous porous media and its parallel implementation on GPU. *Computer Physics Communications*, 238:165–180.

[44] Fučík, R. and Mikyška, J. (2011). Mixed-hybrid finite element method for modelling two-phase flow in porous media. *Journal of Math-for-Industry*, 3(2):9–19.

[45] Fučík, R., Mikyška, J., Beneš, M., and Illangasekare, T. H. (2007). An Improved Semi-Analytical Solution for Verification of Numerical Models of Two-Phase Flow in Porous Media. *Vadose Zone Journal*, 6(1):93–104.

[46] Fučík, R., Solovský, J., Plampin, M. R., Wu, H., Mikyška, J., and Illangasekare, T. H. (2020). Computational Methodology to Analyze the Effect of Mass Transfer Rate on Attenuation of Leaked Carbon Dioxide in Shallow Aquifers. *Acta Polytechnica*.

G

[47] van Genuchten, M. (1980). A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils. *Soil Science Society of America Journal*, 40:892–898.

[48] Gerke, H. H. and van Genuchten, M. T. (1993). A dual-porosity model for simulating the preferential movement of water and solutes in structured porous media. *Water Resources Research*, 29(2):305–319.

[49] Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331.

[50] Gorucu, S. E., Johns, R. T., et al. (2015). Comparison of reduced and conventional two-phase flash calculations. *SPE Journal*, 20(02):294–305.

[51] Griffioen, J. W., Barry, D. A., and Parlange, J.-Y. (1998). Interpretation of two-region model parameters. *Water Resources Research*, 34(3):373–384.

[52] Gupta, P. K. and Yadav, B. (2020). Leakage of CO2 from geological storage and its impacts on fresh soil–water systems: a review. *Environmental Science and Pollution Research*, 27.

[53] Guéguen, Y., Dormieux, L., and Boutéca, M. (2004). Fundamentals of Poromechanics. In Guéguen, Y. and Boutéca, M., editors, *Mechanics of Fluid-Saturated Rocks*, volume 89 of *International Geophysics*, pages 1–54. Academic Press.

H

[54] Hanek, M., Šístek, J., and Burda, P. (2015). An application of the BDDC method to the Navier-Stokes equations in 3-D cavity. In Chleboun, J., Přikryl, P., Segeth, K., Šístek, J., and Vejchodský, T., editors, *Proceedings of Programs and Algorithms of Numerical Mathematics 17, Dolní Maxov, Czech Republic, June 8–13, 2014*, pages 77–85. Institute of Mathematics AS CR.

[55] Hanek, M., Šístek, J., and Burda, P. (2020). Multilevel BDDC for Incompressible Navier-Stokes Equations. *SIAM Journal on Scientific Computing*, 42(6):C359–C383.

[56] Hassanizadeh, M. and Gray, W. G. (1979). General conservation equations for multi-phase systems: 1. Averaging procedure. *Advances in Water Resources*, 2:131–144.

[57] Hassanizadeh, M. and Gray, W. G. (1980). General conservation equations for multi-phase systems: 3. Constitutive theory for porous media flow. *Advances in Water Resources*, 3(1):25–40.

[58] Hassanizadeh, S. M. and Gray, W. G. (1990). Mechanics and thermodynamics of multiphase flow in porous media including interphase boundaries. *Advances in Water Resources*, 13(4):169–186.

[59] Hassanizadeh, S. M. and Gray, W. G. (1993a). Thermodynamic basis of capillary pressure in porous media. *Water Resources Research*, 29(10):3389–3405.

[60] Hassanizadeh, S. M. and Gray, W. G. (1993b). Toward an improved description of the physics of two-phase flow. *Advances in Water Resources*, 16(1):53–67. Research Perspectives in Hydrology.

[61] Haugen, K. B. and Beckner, B. L. (2013). A critical comparison of reduced and conventional EOS algorithms. *SPE Journal*, 18(02):378–388.

[62] Helmig, R. (1997). *Multiphase Flow and Transport Processes in the Subsurface, A contribution to the Modelling of Hydrosystems*. Springer.

[63] Hesse, M. and Woods, A. (2010). Buoyant dispersal of CO2 during geological storage. *Geophysical Research Letters*, 37(1).

[64] Hornung, U. (1996). *Homogenization and porous media*, volume 6. Springer Science & Business Media.

[65] Hoteit, H. and Firoozabadi, A. (2006). Compositional Modeling by the Combined Discontinuous Galerkin and Mixed Methods. *SPE Journal*, 11(01):19–34.

[66] Hoteit, H. and Firoozabadi, A. (2008). Numerical Modeling of Two-Phase Flow in Heterogeneous Permeable Media with Different Capillarity Pressures. *Advances in Water Resources*, 31(1):56–73.

I

[67] Illangasekare, T. H., Frippiat, C. C., and Fučík, R. (2010). *Handbook of Estimation Methods: Environmental Mass Transport Coefficients, Editors: L. J. Thibodeaux and D. Mackay*, chapter Dispersion and Mass Transfer Coefficients in Groundwater of Near-surface Geologic Formations. CRC Press, Taylor and Francis Group, UK.

[68] Imhoff, P. T., Jaffe, P. R., and Pinder, G. F. (1994). An experimental study of complete dissolution of a nonaqueous phase liquid in saturated porous media. *Water Resources Research*, 30(2):307–320.

J

[69] Jung, Y., Pau, G. S. H., Finsterle, S., and Pollyea, R. M. (2017). TOUGH3: A new efficient version of the TOUGH suite of multiphase flow and transport simulators. *Computers & Geosciences*, 108:2–7.

K

[70] Karypis, G. and Kumar, V. (1999). A Fast and Highly Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM Journal on Scientific Computing*, 20:359–392.

[71] Klawonn, A. and Rheinbach, O. (2007). Robust FETI-DP methods for heterogeneous three dimensional elasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1400–1414. Domain Decomposition Methods: recent advances and new challenges in engineering.

[72] Klinkovský, J. (2017). Matematické modelování dvoufázového kompozičního proudění v porézním prostředí. Master's thesis, FNSPE of Czech Technical University Prague.

M

[73] Mandel, J., Sousedík, B., and Dohrmann, C. R. (2008). Multispace and multilevel BDDC. *Computing*, 83:55–85.

[74] Mandel, J., Sousedík, B., and Šístek, J. (2012). Adaptive BDDC in three dimensions. *Mathematics and Computers in Simulation*, 82(10):1812–1831. "The Fourth IMACS Conference: Mathematical Modelling and Computational Methods in Applied Sciences and Engineering" Devoted to Owe Axelsson in ocassion of his 75th birthday.

[75] Mayer, A. and Oostrom, M. (2005). *Site Characterization and Monitoring*, chapter 4, pages 97–139. American Geophysical Union (AGU).

[76] McWhorter, D. B. and Sunada, D. K. (1990). Exact integral solutions for two-phase flow. *Watter Resources Research*, 26:399–413.

[77] Metz, B., Davidson, O., De Coninck, H., Loos, M., and Meyer, L. (2005). *IPCC special report on carbon dioxide capture and storage*. Cambridge: Cambridge University Press.

[78] Miller, C. T., Poirier-McNeil, M. M., and Mayer, A. S. (1990). Dissolution of trapped nonaqueous phase liquids: Mass transfer characteristics. *Water Resources Research*, 26(11):2783–2796.

[79] Millington, R. J. and Quirk, J. P. (1961). Permeability of Porous Solids. *Transactions of the Faraday Society*, 57:1200–1207.

[80] Mualem, Y. (1976). A new model for predicting the hydraulic conductivity of unsaturated porous media. *Water Resources Research*, 12:513–522.

N

[81] de Neef, M. and Molenaar, J. (1997). Analysis of DNAPL infiltration in a medium with a low-permeable lens. *Computational Geosciences*, 1:191–214.

[82] Niessner, J. and Hassanizadeh, S. M. (2008). A model for two-phase flow in porous media including fluid-fluid interfacial area. *Water Resources Research*, 44(8).

[83] Niessner, J. and Hassanizadeh, S. M. (2009). Modeling Kinetic Interphase Mass Transfer for Two-Phase Flow in Porous Media Including Fluid-Fluid Interfacial Area. *Transport in Porous Media*, 80:329–344.

[84] Nordbotten, J. M., Celia, M. A., Dahle, H. K., and Hassanizadeh, S. M. (2007). Interpretation of macroscale variables in Darcy's law. *Water Resources Research*, 43(8).

P

[85] Plampin, M., Illangasekare, T., Sakaki, T., and Pawar, R. (2014a). Experimental study of gas evolution in heterogeneous shallow subsurface formations during leakage of stored CO2. *International Journal of Greenhouse Gas Control*, 22:47–62.

[86] Plampin, M. R., Lassen, R. N., Sakaki, T., Porter, M. L., Pawar, R. J., Jensen, K. H., and Illangasekare, T. H. (2014b). Heterogeneity-enhanced gas phase formation in shallow aquifers during leakage of CO2-saturated water from geologic sequestration sites. *Water Resources Research*, 50(12):9251–9266.

[87] Plampin, M. R., Porter, M. L., Pawar, R. J., and Illangasekare, T. H. (2014c). Multi-scale experimentation and numerical modeling for process understanding of CO2 attenuation in the shallow subsurface. *Energy Procedia*, 63:4824–4833.

[88] Plampin, M. R., Porter, M. L., Pawar, R. J., and Illangasekare, T. H. (2017). Intermediate-Scale Experimental Study to Improve Fundamental Understanding of Attenuation Capacity for Leaking CO2 in Heterogeneous Shallow Aquifers. *Water Resources Research*, 53(12):10121–10138.

[89] Polívka, O. and Mikyška, J. (2015). Compositional modeling of two-phase flow in porous media using semi-implicit scheme. *IAENG Journal of Applied Mathematics*, 45(3):218–226.

[90] Porter, M. L., Plampin, M., Pawar, R., and Illangasekare, T. (2015). CO2 leakage in shallow aquifers: A benchmark modeling study of CO2 gas evolution in heterogeneous porous media. *International Journal of Greenhouse Gas Control*, 39:51–61.

[91] Powers, S. E., Abriola, L. M., and Weber Jr, W. J. (1992). An experimental investigation of nonaqueous phase liquid dissolution in saturated subsurface systems: Steady state mass transfer rates. *Water Resources Research*, 28(10):2691–2705.

[92] Powers, S. E., Abriola, L. M., and Weber Jr, W. J. (1994). An experimental investigation of nonaqueous phase liquid dissolution in saturated subsurface systems: Transient mass transfer rates. *Water Resources Research*, 30(2):321–332.

[93] Przemieniecki, J. S. (1963). Matrix structural analysis of substructures. *AIAA Journal*, 1(1):138–147.

Q

[94] Qafoku, N. P., Lawter, A. R., Bacon, D. H., Zheng, L., Kyle, J., and Brown, C. F. (2017). Review of the impacts of leaking CO2 gas and brine on groundwater quality. *Earth-Science Reviews*, 169:69–84.

R

[95] Rutqvist, J., Birkholzer, J., Cappa, F., and Tsang, C.-F. (2007). Estimating maximum sustainable injection pressure during geological sequestration of CO2 using coupled fluid flow and geomechanical fault-slip analysis. *Energy Conversion and Management*, 48(6):1798–1807. Geologic Carbon Sequestration and Methane Hydrates Research from the TOUGH Symposium 2006.

[96] Rutqvist, J., Birkholzer, J., and Tsang, C.-F. (2008). Coupled reservoir–geomechanical analysis of the potential for tensile and shear failure associated with CO2 injection in multilayered reservoir–caprock systems. *International Journal of Rock Mechanics and Mining Sciences*, 45(2):132–143.

S

[97] Saad, Y. (2003). *Iterative methods for sparse linear systems*. SIAM.

[98] Sahimi, M., Heiba, A. A., Ted Davis, H., and Scriven, L. (1986). Dispersion in flow through porous media - II. Two-phase flow. *Chemical Engineering Science*, 41(8):2123–2136.

[99] Sander, R. (1999). Compilation of Henry's Law Constants for Inorganic and Organic Species of Potential Importance in Environmental Chemistry. *Max-Planck Institute of Chemistry, Mainz, Germany*.

[100] Snir, M., Otto, S., Huss-Lederman, S., Walker, D., and Dongarra, J. (1996). *MPI: The Complete Reference*. The MIT Press.

[101] Solovský, J. and Fučík, R. (2017). A Parallel Mixed-Hybrid Finite Element Method for Two Phase Flow Problems in Porous Media Using MPI. *Computer Methods in Materials Science*, 17:84–93.

[102] Solovský, J. and Fučík, R. (2019). Mass Lumping for MHFEM in Two Phase Flow Problems in Porous Media. In Radu, F. A., Kumar, K., Berre, I., Nordbotten, J. M., and Pop, I. S., editors, *Numerical Mathematics and Advanced Applications ENUMATH 2017*, pages 635–643. Springer International Publishing.

[103] Solovský, J., Fučík, R., Plampin, M. R., Illangasekare, T. H., and Mikyška, J. (2020). Dimensional effects of inter-phase mass transfer on attenuation of structurally trapped gaseous carbon dioxide in shallow aquifers. *Journal of Computational Physics*, 405:109178.

[104] Solovský, J., Fučík, R., and Šístek, J. (2022). BDDC for MHFEM discretization of unsteady two-phase flow in porous media. *Computer Physics Communications*, 271:108199.

[105] Sousedík, B. (2013). Nested BDDC for a saddle-point problem. *Numerische Mathematik*, 125(4):761–783.

[106] Sousedík, B. (2019). On adaptive BDDC for the flow in heterogeneous porous media. *Applications of Mathematics*, 64(3):309–334.

[107] Sousedík, B., Šístek, J., and Mandel, J. (2013). Adaptive-Multilevel BDDC and its parallel implementation. *Computing*, 95:1087–1119.

[108] Šístek, J., Březina, J., and Sousedík, B. (2015). BDDC for mixed-hybrid formulation of flow in porous media with combined mesh dimensions. *Numererical Linear Algebra with Applications*, 22:903–929.

[109] Šístek, J., Čertíková, M., Burda, P., and Novotný, J. (2012). Face-based selection of corners in 3D substructuring. *Mathematics and Computers in Simulation*, 82:1799–1811.

T

[110] Toselli, A. and Widlund, O. (2005). *Domain Decomposition Methods – Algorithms and Theory*. Springer-Verlag.

[111] Tsimpanogiannis, I. and Yortsos, Y. C. (2014). The critical gas saturation in a porous medium in the presence of gravity. *Journal of Colloid and Interface Science*, 270:388–395.

[112] Tu, X. (2007a). A BDDC Algorithm for Flow in Porous Media with a Hybrid Finite Element Discretization. *Electronic Transactions on Numerical Analysis*, 26:146–160.

[113] Tu, X. (2007b). Three-level BDDC in Three Dimensions. *SIAM Journal on Scientific Computing*, 29(4):1759–1780.

[114] Tu, X. and Li, J. (2008). A balancing domain decomposition method by constraints for advection-diffusion problems. *Communications in Applied Mathematics and Computational Science*, 3(1):25–60.

V

[115] van der Vorst, H. A. (1992). Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific & Statistical Computing*, 13(2):631–644.

W

[116] Whitaker, S. (1986). Flow in porous media I: A theoretical derivation of Darcy's law. *Transport in Porous Media*, 1(1):3–25.

[117] Widlund, O. and Dryja, M. (1987). *An additive variant of the Schwarz alternating method for the case of many subregions.* Department of Computer Science, Courant Institute.

Y

[118] Yano, M. (2009). Massively Parallel Solver for the High-Order Galerkin Least-Squares Method. Master's thesis, Massachusests Institute of Technology.

Z

[119] Zampini, S. and Tu, X. (2017). Multilevel Balancing Domain Decomposition by Constraints Deluxe Algorithms with Adaptive Coarse Spaces for Flow in Porous Media. *SIAM Journal on Scientific Computing*, 39(4):A1389–A1415.