

**Integrated Adaptive Numerical Methods for Transient Two-phase Flow
in Heterogeneous Porous Media**

by

CHIH-CHE CHUEH

B.Sc., National Cheng Kung University, 2002

M.Sc., National Taiwan University, 2004

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Mechanical Engineering

© CHIH-CHE CHUEH, 2010

University of Victoria

All rights reserved. This dissertation may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

**Integrated Adaptive Numerical Methods for Transient Two-phase Flow
in Heterogeneous Porous Media**

by

CHIH-CHE CHUEH

B.Sc., National Cheng Kung University, 2002

M.Sc., National Taiwan University, 2004

Supervisory Committee

Dr. Ned Djilali, Supervisor

(Department of Mechanical Engineering, University of Victoria)

Dr. Boualem Khouider, Outside Member

(Department of Mathematics and Statistics, University of Victoria)

Dr. David Sinton, Departmental Member

(Department of Mechanical Engineering, University of Victoria)

Dr. Afzal Suleman, Departmental Member

(Department of Mechanical Engineering, University of Victoria)

Dr. Wolfgang Bangerth, Additional Member

(Department of Mathematics, Texas A&M University)

Supervisory Committee

Dr. Ned Djilali, Supervisor
(Department of Mechanical Engineering, University of Victoria)

Dr. Boualem Khouider, Outside Member
(Department of Mathematics and Statistics, University of Victoria)

Dr. David Sinton, Departmental Member
(Department of Mechanical Engineering, University of Victoria)

Dr. Afzal Suleman, Departmental Member
(Department of Mechanical Engineering, University of Victoria)

Dr. Wolfgang Bangerth, Additional Member
(Department of Mathematics, Texas A&M University)

ABSTRACT

Transient multi-phase flow problems in porous media are ubiquitous in engineering and environmental systems and processes; examples include heat exchangers, reservoir simulation, environmental remediation, magma flow in the earth crust and water management in porous electrodes of PEM fuel cells. This thesis focuses on the development of accurate and computationally efficient numerical models to simulate such flows. The research challenges addressed in this work fall in two areas. For a numerical standpoint, conventional numerical methods including Newton-Raphson linearization and a simple upwind scheme do not always provide the required computational efficiency or sufficiently accurate resolution of the flow field. From a modelling perspective, closure schemes required in volume-averaged formulations, such as the

generalized Leverett J function for capillary pressure, are specific to certain media (e.g. lithologic media) and are not valid for fibrous porous media, which are of central interest in fuel cells.

This thesis presents a set of algorithms that are integrated efficiently to achieve computations that are more than two orders of magnitude faster compared to traditional techniques. The method uses an adaptive operator splitting method based on an a posteriori criterion to separate the flow from the transport equations which eliminates unnecessary and costly solution of the implicit pressure-velocity term at every time step; adaptive meshing to reduce the size of the discretized problem; efficient block preconditioned solver techniques for fast solution of the discrete equations; and a recently developed artificial diffusion strategy to stabilize the numerical solution of the transport equation. The significant improvements in accuracy and efficiency of the approach is demonstrated using numerical experiments in 2D and 3D. The method is also extended to advection-dominated problems to specifically investigate two-phase flow in heterogeneous porous media involving capillary transport. Both hydrophilic and hydrophobic media are considered, and insights relevant to fuel cell electrodes are discussed.

Contents

Supervisory Committee	ii
Abstract	iii
Table of Contents	v
List of Figures	vii
Acknowledgements	viii
Dedication	ix
1 Introduction	1
1.1 Background and motivation	1
1.2 Contributions	3
1.3 Goal and structure	4
2 Physical and mathematical background	7
2.1 Basic definitions and concepts	7
2.2 Constitutive relationships	9
2.3 Two-phase flow governing equations	12
2.4 Buckley-Leverett problem	13
2.5 Finite element method	18
2.6 Artificial diffusive stabilization	26
2.7 Adaptive mesh refinement	27
2.8 Operator splitting techniques	31
2.9 Iterative solution of linear systems	34
3 Conclusions and future work	38
3.1 Conclusions	38

3.2 Future work	40
A Multi-level Adaptive Simulation of Transient Two-phase Flow in Heterogeneous Porous Media	42
B An H-adaptive Operator Splitting Method for Two-phase Flow in 3D Heterogeneous Porous Media	55
C Saturation and Permeability Gradient Effects in Two-phase Flows in Heterogeneous Porous Media	98
D Numerical Analysis of Transient Two-phase Flow in Hydrophobic Heterogeneous Porous Media	129
Bibliography	156

List of Figures

Figure 1.1	Examples of porous materials. (a) carbon fibers in diffusion media of PEM fuel cells. (b) a piece of slate in porous media of reservoir rocks.	2
Figure 2.1	Typical relative permeability curves	11
Figure 2.2	Multiple-valued saturation profile adapted from Buckley and Leverett [1] and Wooding & Morel-Seytoux [2]	15
Figure 2.3	Variation of the wetting-phase fraction of the volumetric flow with respect to saturation. Top: the fractional flow of wetting phase F_w adapted from Wooding & Morel-Seytoux [2]; bottom: the first derivative of F_w with respect to S_w	17
Figure 2.4	One dimensional Lagrange linear element and associated basis functions	23
Figure 2.5	One dimensional Lagrange quadratic element and associated basis functions	23
Figure 2.6	Illustration of two dimensional Lagrange bi-linear mapping . . .	25
Figure 2.7	Comparison of finite element method between discontinuous Galerkin space and continuous Lagrange space. (see Appendix A or Chueh et al. [3] for further details)	26
Figure 2.8	Refinement and coarsening bounds for the absolute-value method	32
Figure 2.9	Illustration of time operator splitting of the general form of a convection-diffusion equation in a reactive system (C : a scalar variable; ν : a diffusion coefficient; \mathbf{u} : a velocity vector; S_r : a reactive source term)	34

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support of four people. I wish to express my gratitude to my main supervisor, Dr. Ned Djilali, who was abundantly helpful and offered invaluable assistance, support and guidance, and offered enough funding for me to perform my research during my PhD program. In addition, I want to give a special thank to Dr. Wolfgang Bangerth (Texas A&M University) for providing timely and instructive comments, important inputs and constructive suggestions. Next, I wish to thank Dr. Chien-Cheng Chang (National Taiwan University) and Dr. Maw-Ji Chao (Chung-Shan Institute of Science and Technology) for strictly training me to have a solid grounding in the expertise of applied mathematics during my Master program.

Next, thanks are due to Dr. Tao Tang (Hong Kong Baptist University) for kindly inviting me to regularly submitting my future papers to the Journal Communications in Computational Physics, to Dr. Zhangxin Chen (University of Calgary) for his invaluable insights on finite element theory, and to Dr. Sergey Mashchenko (McMaster University) for installing the relevant libraries.

Finally, deepest gratitudes are also due to one member of the supervisory committee, Dr. Boualem Khouider; his course in applied mathematics motivated me to work in this challenging area, and helped to set a solid foundation for further understanding of the subject.

Special thanks also to all my friends, especially group members; Jay Sui, Te-Chun Wu, Kyle Lange, Chin-Hsien Cheng, Jingwei Hu, Viatcheslav Berejnov, Boris Chernyavsky, Marc Secanell, Carlos Escobedo, Peyman Taheri, Naser Yasrebi, Erik Kjeang and Aimy Bazylak for sharing with me your enthusiasm for research and all your expertise.

Last but importantly, I wish to express my thanks to my dear wife, Mrs. Hsiao-Yun Janette Cheng, for helping me with graphics and constantly giving me strength to finish my thesis, and to my parents, Mr. Ching-Mu Chueh and Mrs. Yueh-Chin Lin, for giving me opportunities and encouragement to learn. Without their support, the thesis would never have been completed.

DEDICATION

To my family

Chapter 1

Introduction

1.1 Background and motivation

Flow processes in porous media play a decisive role in human civilization and industrial applications. A theme of topical interest, for example, is the issue of water management in diffusion media (porous media) of proton exchange membrane (PEM) fuel cells. As PEM fuel cells operate, excess water, produced by electrochemical reactions, accumulates inside their diffusion media (See Figure 1.1a) and, as a result, prevents gas reactants from reaching reaction sites called catalyst layers, resulting in a significant decrease in cell performance. Therefore, the design of diffusion media to facilitate water transport is very important. Before the design, a clear understanding of the relevant flow processes of the water and gas mixture (e.g. air) in porous media is necessary. Numerical simulations of two-phase flow processes which consider porous material properties can help engineers provide a correct prognosis of the reliability or applicability of the different materials of diffusion media.

Another important issue is reservoir simulations. The primary objective of a reservoir study is to predict the future performance of a reservoir and find ways and means of increasing the ultimate oil recovery or production in complex geological environments (see Figure 1.1b). Reservoir simulation technology is constantly improved and enhanced. New models to simulate more complex recovery schemes are being proposed all the time. A thorough understanding of the techniques used for black-oil models is essential in order to develop some appreciation for more complex models.

These examples cover only a small part of the wide field of multi-phase flow processes. There are numerous applications and special cases that must be dealt

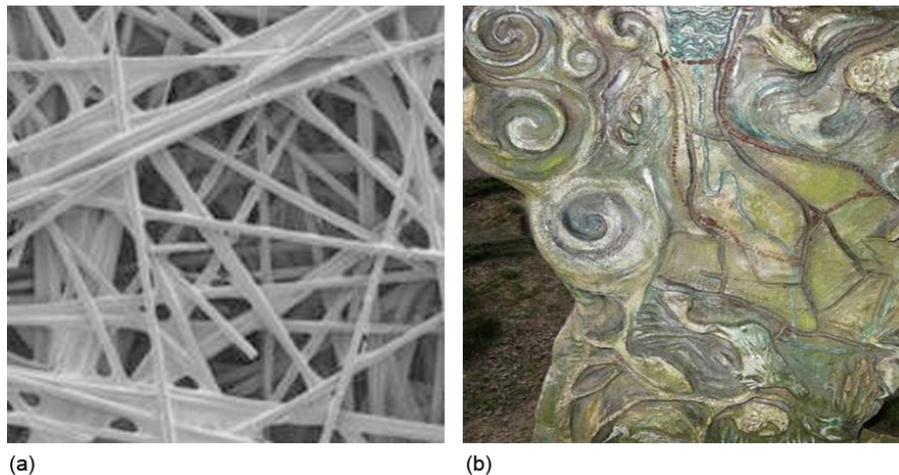


Figure 1.1: Examples of porous materials. (a) carbon fibers in diffusion media of PEM fuel cells. (b) a piece of slate in porous media of reservoir rocks.

with. Certain aspects, however, can be found in a large number of problems. In the following, three of these aspects are explained further.

The first one is the so-called sharp concentration or saturation profiles. These take place when diffusive effects play only a negligible role and there is an abrupt transition between a region completely filled with one fluid (e.g. water) and another region completely saturated with another fluid (e.g. oil). Sharp fronts create a difficult problem for a numerical scheme to solve. The difficulties lie not only in the choice of a sufficiently small element-size resolution, but also in the choice of a suitable discretization technique. This is because the numerical solutions for problems involving sharp fronts tend to show unphysical oscillations or display incorrect profiles if incorrect methods are used.

The second aspect involves heterogeneities, i.e. heterogeneous porous media, and specific physical properties of a particular porous material. They greatly influence the flow processes because they affect the spatiotemporal flow behaviors and therefore the mass distribution of phases in a system. This is found not only in PEM fuel cells but also in many civil engineering applications where there is a large number of problems involving heterogeneity.

The third aspect is how to increase the computational efficiency for the pressure-velocity part in multi-phase flow processes in porous media. In the most simulations, the bulk of the computational time is spent on the implicit calculation of the pressure. In fact, according to actual test simulations without an appropriate preconditioner

for solving linear systems, the computation of the pressure can be more than 99 percent of the total computational load. Therefore, in order to improve the computational efficiency of numerical methods, the cost of the pressure calculation must be substantially reduced.

The aspects mentioned above increase the complexity of two-phase flow problems. Especially for real world problems (i.e. not just academic problems), it is not only important to get correct numerical results as a basis for predictions, but also to get results as fast as possible. Based on what has been mentioned, the objective of this thesis is to integrate state-of-the-art numerical methods and establish an efficient, correct framework from which more complicated models in the field of multi-phase flow in porous media can develop. Specifically, this work aims to integrate four numerical methods: a powerful diffusive stabilization for damping unphysical oscillations, adaptive mesh refinement (AMR) for identifying regions of higher importance while areas of lower importance are treated with lower computational resolution, adaptive operator splitting to avoid solving the computationally expensive pressure term at every saturation time step, as well as an efficient preconditioner for solving linear systems. This combination of approaches makes it more practical to perform 2D/3D simulations that would enhance the applicability of the integrated methods described herein to a broader range of multi-phase transport problems of practical interest.

1.2 Contributions

The main contributions of this work are located in the areas of numerical methods and modelling of transient two-phase transport problems in porous media. In the area of numerical analysis, this work contributes to the literature by the following:

1. introducing Guermond and Pasquetti's diffusive stabilization (see Appendix A for details) to the field of multi-phase flow in porous media and successfully validating for the first time that it can work together correctly with the implicit pressure and explicit saturation (IMPES) finite element method and AMR and with no upwinding.
2. developing an adaptive operator splitting with an a posteriori criterion instead of a fixed operator splitting (see Appendix B for details) and at the same time providing not only theoretical numerical analysis but also, most importantly, proofs.

3. introducing an existing, fast algorithm for block matrix preconditioning (see Appendix B for details) to solve linear systems.
4. integrating all the previously-mentioned state-of-the-art techniques in order to increase the computational efficiency and enhance the numerical accuracy.
5. making the implementation based on deal.II (one Open Source finite element software written in C++) available as part of this library's tutorial so that it becomes an effective framework from which more complicated models for multi-phase problems in porous media will develop in the future.

In the area of physical modelling, this work corrects a misunderstanding: some people in the community of multi-phase fuel cell modelling underestimate the ability of the multi-phase mixture approach (MMA) to accurately capture spatiotemporal transport phenomena since they thought that MMA could not account properly for mass and momentum transfer across phase boundaries. But, in reality, this problem is caused by a numerical issue (see Appendix C and D for details), rather than the MMA itself.

Second, in this work, with Guermond and Pasquetti's stabilization, the issue of sharp fronts stemming from an advection term in the MMA formulation is able to be resolved without causing any unphysical oscillations. Therefore, in this work, the numerical issue is no longer a problem in MMA.

Third, to the best of our knowledge, there has been no paper dealing with the permeability gradient effect which affects capillary transport. However, this effect would be one of the necessary conditions to pursue the direction in trying to simulate realistic physical transport phenomena in porous media (see Appendix C and D for details).

1.3 Goal and structure

This thesis is organized into three chapters providing the background and an overview of the methodology and contributions, followed by four appendices consisting of journal paper manuscripts that have been published (Appendix A), submitted (Appendix B), and prepared for submission (Appendices C and D)

CHAPTER 2: This chapter gives an overview of the basic mathematical and physical background for two-phase flow in porous media. Additionally, the rationale be-

hind forming jump conditions caused by the fluid advection mechanism is described theoretically. Finally, the representative comparison between discontinuous Galerkin (DG) and continuous finite elements is shown for readers to easily recognize a subtle difference.

CHAPTER 3: Finally, Chapter 3 presents some final conclusions and possible avenues for future research.

APPENDIX A: The goal of the first piece of this work is to ensure that Guermond and Pasquetti's diffusive stabilization can work correctly together with AMR and IMPES, increasing the accuracy of representative simulations. In fact, the present numerical results are essential in agreement with those that use a DG space for pressure and saturation, and a Raviart-Thomas space for a vector variable (i.e. velocity).

APPENDIX B: The goal of the second piece of this work is to focus more on improving the computational efficiency by incorporating an adaptive operator splitting method for splitting the computationally expensive pressure from saturation and an existing, fast algorithm for block matrix preconditioning to efficiently solve linear systems. Most importantly, theoretical numerical analysis as well as proofs are given with a newly developed indicator function to adaptively determine whether to solve the pressure-velocity part in each saturation time step. Finally, 2D/3D results presented in this appendix are validated numerically.

APPENDIX C: The previous basic advection-dominated model is extended into one which considers immobile or irreducible saturation and capillary transport in hydrophilic porous media. The detailed physical interpretations behind two-phase mixture formulae are described. In addition, this is the first time that the permeability gradient effect on capillary transport is included in the modelling of two-phase flow in porous media.

APPENDIX D: Formulation of a two-phase transport problem in hydrophobic media is considered in order to simulate two-phase flow transport phenomena in diffusion media of PEM fuel cells. The model which uses a standard Leverett J function is compared to that which employs another newly validated Leverett J function for PEMFC diffusion media. More importantly, the main goal of this piece of the work is to correct a faulty statement in the field of multi-phase fuel cell modelling that the MMA does not have the ability to account properly for mass and momentum transfer across phase boundaries compared to the multi-fluid fuel cell model.

The manuscripts in the above Appendices stem from research done entirely by the author, who was also responsible for the first draft of each of the manuscripts. The co-

authors contributed to setting research directions (ND), guidance in methodology and implementation (ND, WB), initial guidance in use of deal.II library (MS), guidance in interpretation and analysis of results, and technical and editorial input for the draft manuscripts (ND, WB).

Chapter 2

Physical and mathematical background

2.1 Basic definitions and concepts

Averaging process A wide range of materials can be considered as porous media. Among them are, for example, synthetic forms (used as impact absorbers in cars) and bone materials. The porous material that will be considered in this thesis, however, is the gas diffusion layer (GDL) in PEM fuel cells [4, 5]. The GDL, which is about 200 micrometers thick, consists of carbon fibers and PTFE, a hydrophobic material which coats the carbon fibers. The function of the PTFE coating is to prevent the accumulation of liquid water in the GDLs. The gas diffusion layer (porous media) provides pathways for gaseous reactants to reach the catalysts. However, when PEM fuel cells are in operation, water is produced by the electrochemical reactions. This water can accumulate inside the gas diffusion layer, hindering the effective reactant transport and leading to a significant drop in cell performance.

All porous media (both compressible and incompressible) are composed of a solid material and void spaces. These spaces are also called pores. Due to heterogeneity of the porous media, it is impossible or difficult to completely describe the geometry of the pores. Therefore, porous media flow models often consist in a continuum approach. Here, the properties on the microscale are averaged over a representative elementary volume (REV) [6], which represents the macroscale ($\geq 10^{-6}m$) in this thesis. The discontinuities that are present on the microscale are now smeared and no longer recognizable. This averaging process creates a new set of physical quantities,

which are only available at the macroscopic level, such as the saturation and porosity. These physical quantities will be introduced in the following.

The difficulties arising from the continuum approach lie in the choice of the size of a REV: if the selected REV is too large, process-related discontinuities may also be smeared and the simulation results may therefore give a faulty picture of the actual circumstances.

Fluid wettability: When a two-phase flow in porous media is considered, the fluids cannot only be distinguished by their specific fluid properties such as density and viscosity. They also differ in their contact angle. The contact angle θ_c is the angle at which a liquid/vapor interface meets a solid surface and that varies between 0° and 180° . The contact angle represents the degree of the wettability of the porous media ($\theta_c < 90^\circ$ for hydrophilic media and $\theta_c > 90^\circ$ for hydrophobic media). In the case of the fluids considered in hydrophobic porous media, liquid water represents the non-wetting phase and is marked by nw . The wetting phase is comprised by a gas mixture (i.e. air) and is marked by w .

Porosity: The porosity ϵ of a porous medium is defined as the fraction of the total volume of the medium that is occupied by void space:

$$\epsilon = \frac{\text{volume of the pore space}}{\text{volume of the REV}} \quad (2.1)$$

And therefore $1 - \epsilon$ is the fraction that is occupied by the solid.

Saturation: For two (immiscible) fluids, each one shares the pore space with the other. Although the exact location of the fluid particles is no longer known due to the averaging process, how much of the pore space in a REV is filled with a certain phase is still of interest. This is described by the saturation of phase j (S_j):

$$S_j = \frac{\text{volume of fluid } j \text{ within REV}}{\text{volume of the pore space within the REV}} \quad (2.2)$$

It is assumed that the fluid phases fill the pore space completely so the sum of the different saturations must be unity:

$$\sum_j S_j = S_w + S_{nw} = 1 \quad (2.3)$$

Mass-averaged velocities: For a single-phase flow, the mass-averaged velocity

of a fluid in a porous medium is usually governed by Darcy's Law:

$$\mathbf{u} = -\frac{1}{\mu} \mathbf{K} (\nabla p - \rho \mathbf{g}) \quad (2.4)$$

where μ is the viscosity, \mathbf{K} is the tensor of absolute permeability, p is the pressure, and \mathbf{g} is the vector of gravitational acceleration $((0, 0, -g)^T$ in three dimensions). This Darcy's Law is derived from an experimental point of view: it describes the amount of fluid which flows over the whole cross-section of the domain. So, based on this mass-averaged velocity, it is assumed that the fluid flow is not restricted to the pores of a porous medium alone but uses the entire area. Of course, in reality, the flow only takes place in the pores, so the same amount of fluid has to flow through a much smaller area with a much higher velocity.

For an extension to multi-phase flow in porous media (see Scheidegger [7] and Helmig [8]), we have

$$\mathbf{u}_j = -\frac{1}{\mu_j} \mathbf{K}_j (\nabla p_j - \rho_j \mathbf{g}) \quad (2.5)$$

$$= -\frac{k_{rj}}{\mu_j} \mathbf{K} (\nabla p_j - \rho_j \mathbf{g}) \quad (2.6)$$

$$= -\lambda_j \mathbf{K} (\nabla p_j - \rho_j \mathbf{g}) \quad (2.7)$$

where \mathbf{u}_j is called the phase j velocity, j represents the subscript for the different phases w and nw , k_{rj} is the relative permeability for each phase and will be described in the following, \mathbf{K} is the absolute permeability, while p_j and ρ_j are the pressure and density for each phase. The ratio k_{rj}/μ_j is also called the mobility λ_j .

2.2 Constitutive relationships

Capillary pressure: The interface between a wetting and a non-wetting phase is always curved. Due to equilibrium constraints, the pressure of the non-wetting phase has to be larger at the interface than that of the wetting one. The difference between those two pressures is called the capillary pressure p_c :

$$p_c = p_{nw} - p_w \quad (2.8)$$

In fact, from a microscopic point of view, the capillary pressure depends on the interfacial tension and the pore radius (see, for example, Lister and Djilali [9] and Helmig [8]). The smaller the pore radius, the larger the capillary pressure. This indicates that, in the case of a wetting-phase drainage process, the wetting fluid in larger pores drains more easily than the wetting fluid contained in smaller pores. The amount of entrapped wetting phase fluid is called the residual or irreducible saturation of wetting phase (S_{wirr}).

In analogy to the residual saturation of the wetting phase, a residual saturation of the non-wetting phase can also exist. This means that, if the porous medium is filled mostly with a wetting fluid, small entrapped non-wetting phase bubbles or drops still exist that will not vanish.

In order to describe the behavior of different residual saturations, the effective saturation S_e is introduced:

$$S_e = \frac{S_w - S_{wirr}}{1 - S_{wirr} - S_{nwirr}} \quad (2.9)$$

In this thesis, S_{nwirr} is set to zero. The effective saturation can be correlated with the capillary pressure, based on the fact that as the saturation of the wetting-phase decreases, the capillary pressure increases. By convention, this function is mostly expressed in S_w rather than S_e .

In the literature, there are various empirical approaches of the $p_c(S_w)$ function. The simplest one is to assume a linear dependence [10]:

$$p_c(S_w) = (p_c)_{max} \left(1 - \frac{S_w - S_{wirr}}{1 - S_{wirr} - S_{nwirr}} \right) \quad (2.10)$$

where $(p_c)_{max}$ represents the maximum capillary pressure when $S_{wirr} = 0$. A more sophisticated approach is the one after Leverett [11, 12]:

$$p_c = \frac{\sigma_c \cos(\theta_c)}{\left(\frac{k}{\epsilon}\right)^{1/2}} J(S_e) \quad (2.11)$$

where σ_c is the surface tension and k is the absolute permeability. This approach has been used widely for numerical modelling in the community of PEM fuel cells, even though it is not of general applicability since the J-function originates from specific lithologic-type porous media [13].

Relative permeability Underlying the extension of the motion equation of a

single fluid to the simultaneous flow of two or more fluids is the concept of relative permeability. The relative permeability k_r represents another important physical quantity for multi-phase flow in porous media. At first sight, it is reasonable to assume that when the flow of one of the fluids at an area is being considered, since part of the pore space in the vicinity of that area is occupied by another fluid, the permeability of the porous medium would be reduced with respect to the fluid considered. This implies that the relative permeability depends only on the saturation. In fact, it describes to what extent the presence of one fluid affects the flow behavior of another fluid. This effect is due to interaction forces as well as a change of possible flow pathways.

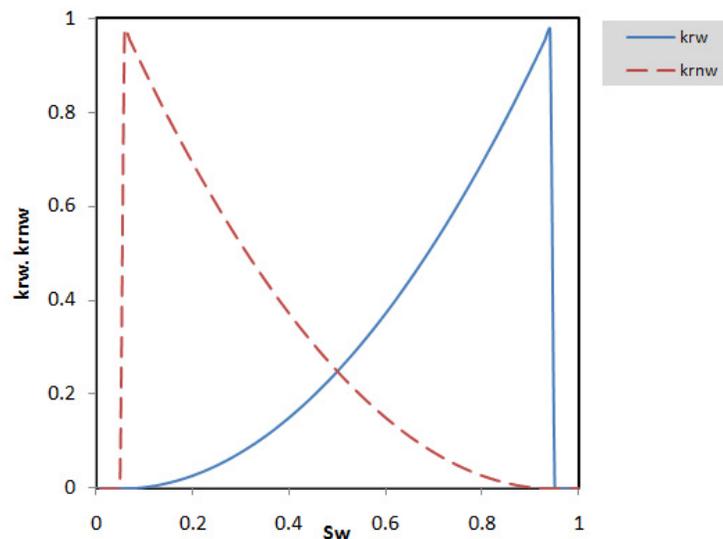


Figure 2.1: Typical relative permeability curves

Figure 2.1 shows typical relative permeability curves for a pair of fluids. The rapid decline in k_{rw} indicates that the larger pores are occupied first by the non-wetting phase. As S_{nw} increases, the average pore size saturated by the wetting fluid becomes progressively smaller. This is demonstrated by a rapid rise in k_{rnw} . In other words, when the non-wetting saturation increases beyond its residual saturation, the non-wetting fluid occupies larger pores than does the wetting one. So, if one phase fills out the available pore size completely (irrespective of the residual saturation of the other phase), the relative permeability for this phase is 1. On the other hand, if a phase is only present in residual saturation and therefore remains immobile, the relative permeability is zero.

As it is impossible to describe the complex pore geometry precisely, the relationship between the relative permeability and saturation can only be described quantitatively.

In this thesis, the relative permeability is determined using the commonly used relationships [14]:

$$k_{rw} = S_e^n \quad (2.12)$$

$$k_{rnw} = (1 - S_e)^n \quad (2.13)$$

where $n = 2$ is used in Appendices A and B whereas $n = 3$ in Appendices C and D.

2.3 Two-phase flow governing equations

The governing equations for the flow of two immiscible fluid phases in porous media are given by the conservation of mass

$$\epsilon \frac{\partial (\rho_w S_w)}{\partial t} + \nabla \cdot (\rho_w \mathbf{u}_w) = \rho_w q_w \quad (2.14)$$

$$\epsilon \frac{\partial (\rho_{nw} S_{nw})}{\partial t} + \nabla \cdot (\rho_{nw} \mathbf{u}_{nw}) = \rho_{nw} q_{nw} \quad (2.15)$$

and the conservation of momentum in terms of Darcy's Law

$$\mathbf{u}_w = -\frac{k_{rw}}{\mu_w} \mathbf{K} \nabla p_w = -\lambda_w \mathbf{K} \nabla p_w \quad (2.16)$$

$$\mathbf{u}_{nw} = -\frac{k_{rnw}}{\mu_{nw}} \mathbf{K} \nabla p_{nw} = -\lambda_{nw} \mathbf{K} \nabla p_{nw} \quad (2.17)$$

In the community of PEM fuel cells, the set of equations is called a multi-fluid model (e.g. Lister and Djilali [9]).

The four equations above, however, cannot evaluate the six variables (i.e. S_w , S_{nw} , \mathbf{u}_w , \mathbf{u}_{nw} , p_w , p_{nw}) that must be solved. The situation is resolved completely by two supplementary equations:

$$S_w + S_{nw} = 1 \quad (2.18)$$

and

$$p_c = p_{nw} - p_w \quad (2.19)$$

With the use of these relations, two of the four unknowns (i.e. S_w , S_{nw} , p_w , p_{nw}) can

be eliminated.

After a sophisticated mathematical derivation (See Appendix C for details), the equations (2.14), (2.15), (2.16), (2.17), (2.18), and (2.19) are rearranged into another set of equations:

$$\mathbf{u}_t = -\lambda_t \mathbf{K} \nabla p_w - \lambda_{nw} \mathbf{K} \nabla p_c \quad (2.20)$$

$$\nabla \cdot \mathbf{u}_t = q_w + q_{nw} \quad (2.21)$$

$$\epsilon \frac{\partial S_w}{\partial t} + \nabla \cdot (F_w \mathbf{u}_t) + \nabla \cdot (\lambda_{nw} F_w \mathbf{K} \nabla p_c) = q_w \quad (2.22)$$

where the total mobility λ_t , fractional flow of the wetting phase F_w and total velocity \mathbf{u}_t ¹ are respectively defined by

$$\lambda_t = \lambda_w + \lambda_{nw} = \frac{k_{rw}}{\mu_w} + \frac{k_{rnw}}{\mu_{nw}} \quad (2.23)$$

$$F_w = \frac{\lambda_w}{\lambda_t} = \frac{\lambda_w}{\lambda_w + \lambda_{nw}} = \frac{k_{rw}/\mu_w}{k_{rw}/\mu_w + k_{rnw}/\mu_{nw}} \quad (2.24)$$

$$\mathbf{u}_t = \mathbf{u}_w + \mathbf{u}_{nw} \quad (2.25)$$

In the community of PEM fuel cells, this set of equations is called a mixture model (e.g. Lister and Djilali [9]).

2.4 Buckley-Leverett problem

When the advective (hyperbolic) term in equation (2.22) is dominant, its approximation is a main problem when discretizing the two-phase flow equations. In order to explain the problem arising here, we have to take a look at a special case of two-phase flow: the Buckley-Leverett problem. Here, capillary effects are neglected, which indicates that there is no pressure difference existing on the interface of two fluids (i.e., $p_w = p_{nw}$). Also, sources and sinks are omitted. For a quasi-linear system, this leads to the following equation (Buckley and Leverett [1]):

$$\epsilon \frac{\partial S_w}{\partial t} + \mathbf{u}_t \frac{dF_w}{dS_w} \frac{\partial S_w}{\partial x} = 0 \quad (2.26)$$

¹in this thesis \mathbf{u}_t denotes the total velocity; the notation of the first derivative of velocity with respect to time is simply represented by $\frac{\partial \mathbf{u}}{\partial t}$.

This is a quasi-linear first order partial differential equation, which can be treated by a numerical method or by the method of characteristics.

The total derivative of $S_w(x, t)$ with respect to time is:

$$\frac{DS_w}{Dt} = \frac{\partial S_w}{\partial x} \frac{dx}{dt} + \frac{\partial S_w}{\partial t} \quad (2.27)$$

If $x = x(t)$ is chosen to coincide with an advancing surface of fixed S_w , then on such advancing surface we have:

$$\frac{DS_w}{Dt} = 0 \quad (2.28)$$

After a combination of equations (2.27) and (2.28), one equation is obtained as follows:

$$\left. \frac{dx}{dt} \right|_{S_w} = - \frac{\frac{\partial S_w}{\partial t}}{\frac{\partial S_w}{\partial x}} \quad (2.29)$$

Physically, the above equation is equivalent to a velocity (i.e. $\mathbf{u}_F = dx/dt$) at which the front of a given S_w is advancing. It is noted that \mathbf{u}_F represents the velocity of saturation fronts along a fixed S_w propagation rather than the total velocity of fluid particles moving \mathbf{u}_t .

By combining (2.26) and (2.29), we have:

$$\mathbf{u}_F = \left. \frac{dx}{dt} \right|_{S_w} = \frac{\mathbf{u}_t}{\epsilon} \frac{dF_w}{dS_w} \quad (2.30)$$

Therefore, the partial differential equation (2.26) has been replaced by two ordinary differential equations (2.28) and (2.30) which are called the Buckley-Leverett equations. By integrating (2.30) with respect to time, we obtain:

$$x|_{S_w}(t) - x|_{S_w}(0) = \frac{dF_w}{dS_w} \frac{U(t) - U(0)}{\epsilon A} \quad (2.31)$$

where $x|_{S_w}(t)$ and $x|_{S_w}(0)$ are the coordinates x of a plane at which a specified saturation S_w exists at times t and 0. A is a cross-section area and $U(t)$ is the cumulative total volume passed through the system at a time t and is defined as

$$U(t) = \int_0^t \mathbf{u}_t(t) A dt \quad (2.32)$$

The complication with the Buckley-Leverett approach lies in the fact that equation

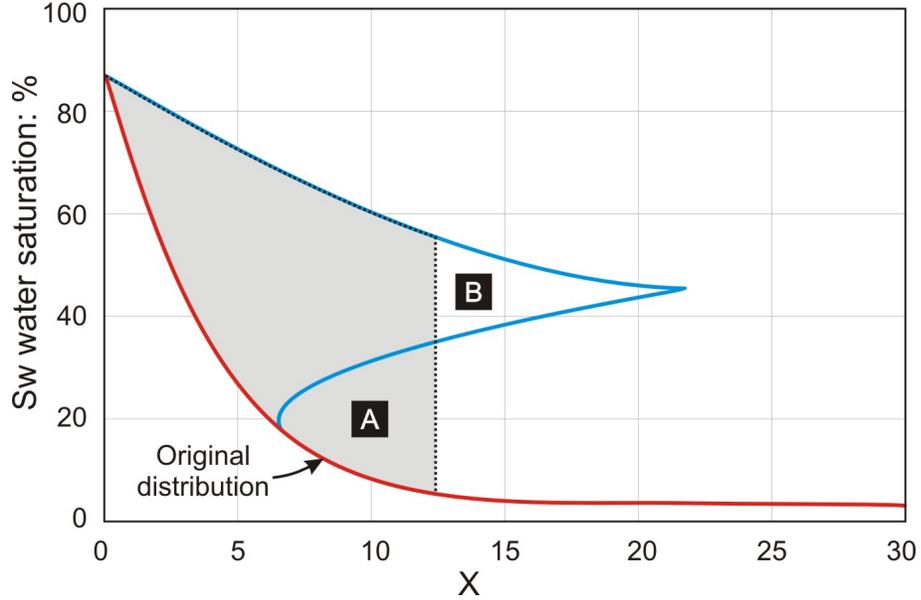


Figure 2.2: Multiple-valued saturation profile adapted from Buckley and Leverett [1] and Wooding & Morel-Seytoux [2]

(2.31) leads to a profile that is multiple-valued in S_w as a function of x (see the blue dotted line in Figure 2.2), requiring the introduction of a discontinuity or front (see the black dotted line in Figure 2.2). Buckley and Leverett [1] explained the reason as to why their model failed to express the correct profile and the need for correction by a front: the new computed curve is S-shaped and is triple-valued between $S_w \approx 7$ and $S_w \approx 22$. So, as a front shape with such multiple saturation values is physically meaningless, the correct saturation profile can be ensured by fulfilling two conditions (Lake [15] and Helmig [8]): a jump condition (Rankine Hugoniot) and an entropy condition.

Jump condition: As mentioned previously, the front velocity \mathbf{u}_F in equation (2.30) is proportional to dF_w/ds_w . In order to satisfy the continuity equation at a saturation jump, the front velocity must satisfy the following condition (Helmig [8]):

$$(\mathbf{u}_F)_{\Delta S_w} = \frac{F_w(S_w^u) - F_w(S_w^d)}{S_w^u - S_w^d} = \frac{\Delta F_w}{\Delta S_w} \quad (2.33)$$

where S_w^u represents the upstream while S_w^d stands for the downstream wetting phase saturation of the shock. This means that the velocity at which the front travels is proportional to the slope of F_w with respect to S_w .

Equation (2.33) tells us that the specific front velocity has to correspond to the physical process in order to fulfill the continuity condition at the jump. This condition, however, does not guarantee that one physical position corresponds to only one physical saturation (i.e. multiple values). For example, in Figure 2.2, the blue solid line produces more than one saturation value over a portion of its length, although it satisfies the continuity equation (the areas of A and B are the same: mass is distributed equally). This can be fixed by the next condition.

Entropy condition: The saturation jump must fulfill the entropy condition (LeVeque [16]):

$$\frac{F_w(S_w) - F_w(S_w^d)}{S_w - S_w^d} \leq \mathbf{u}^{shock} \leq \frac{F_w(S_w) - F_w(S_w^u)}{S_w - S_w^u} \quad (2.34)$$

This is valid for all S_w between S_w^u and S_w^d . Interpreted physically, the upstream velocity \mathbf{u}_F^u of the shock always has to be larger than the downstream velocity \mathbf{u}_F^d . For example, in Figure 2.3, in the case of $\mu_w/\mu_{nw} = 0.1$, the front will only be correct for saturations which are larger than an inflection value (i.e., $d^2F_w/dS_w^2 = 0$) roughly at $S_w = 0.2$. More precisely, in a physically correct flow, S_w decreases monotonically with x ($x = 0$ is the location of injection of the wetting phase). Then, the front velocity \mathbf{u}_F is expected to decrease as S_w increases such that the higher saturations are always behind the lower saturations. For this to happen, we need $d^2F_w/dS_w^2 < 0$ so that \mathbf{u}_F and dF_w/dS_w ($\mathbf{u}_F \propto dF_w/dS_w$) decrease with respect to S_w . Therefore, in the case of $\mu_w/\mu_{nw} = 0.1$ (see Figure 2.3), for approximately $0.0 \leq S_w \leq 0.2$ where $d^2F_w/dS_w^2 > 0$, there is no solution with a physical shock that satisfies the entropy condition. In addition to what has been mentioned above, there is one more interesting thing in Figure 2.3. As the viscosity ratio (μ_w/μ_{nw}) increases, the inflection point (i.e. $d^2F_w/dS_w^2 = 0$) gradually shifts to the right. This means that the area of $d^2F_w/dS_w^2 > 0$ becomes larger and larger as the viscosity ratio rises. For example, in the case of $\mu_w/\mu_{nw} = 10.0$, the stable region ranges from approximately 0.8 to 0.1 while, in the case of $\mu_w/\mu_{nw} = 0.1$, the stable region lies between about 0.2 and 1.0. The physical representation of this is that when the viscosity ratio is much larger or smaller than one, the particles of one fluid are advancing at velocities much higher than those of the other fluid, resulting in a special phenomenon called interface instability or fingering [6].

If capillary effects dominate over convection, the entropy condition is fulfilled (Helmig [17]). However, if the capillary (diffusion) term approaches to zero, the

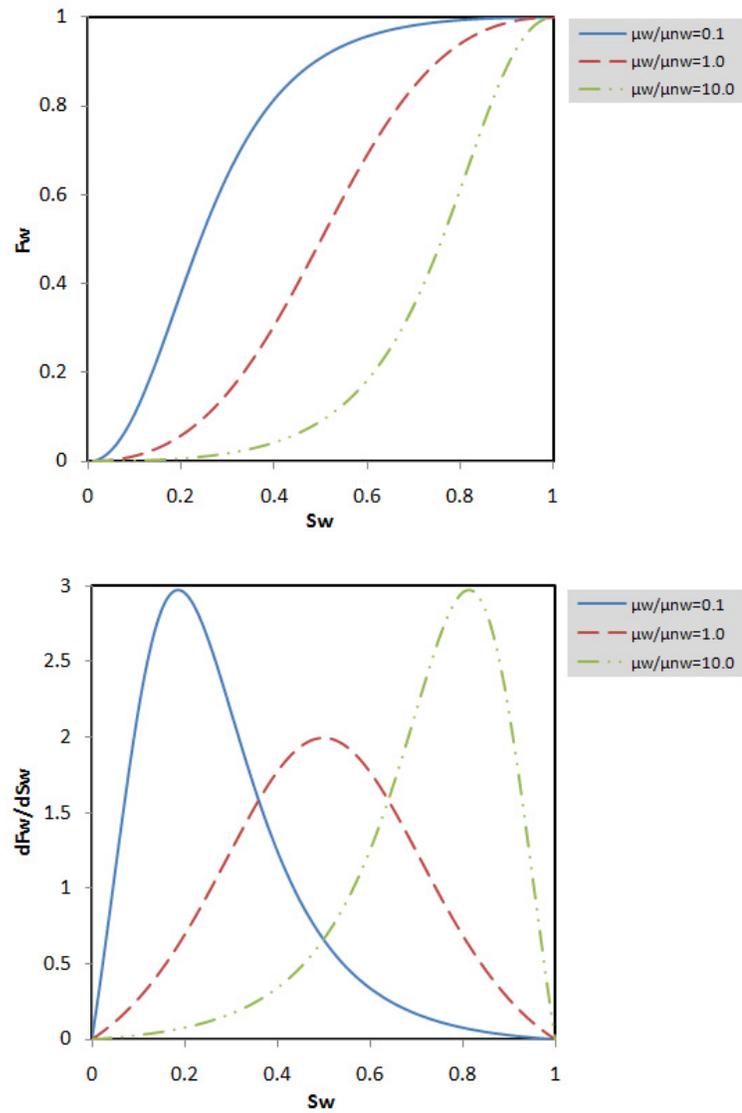


Figure 2.3: Variation of the wetting-phase fraction of the volumetric flow with respect to saturation. Top: the fractional flow of wetting phase F_w adapted from Wooding & Morel-Seytoux [2]; bottom: the first derivative of F_w with respect to S_w

discretization method has to introduce enough numerical diffusion in order to make sure that the entropy condition is satisfied.

2.5 Finite element method

The purpose of this section is to briefly illustrate the basic ideas about the finite element method (FEM). Therefore, the section is neither exhaustive nor complete. Readers are referred to the existing literature for a comprehensive presentation (e.g. Hirsch [18], Zienkiewicz et al. [19], Ern and Guermond [20], and Chen [21]).

It is assumed that the reader is familiar with a few mathematical terms: function orthogonality, Galerkin's method, weak form formulation, basis function, method of variation, spectral method, collocation method, spline interpolation method, Legendre polynomials, Lagrange polynomials, Chebyshev polynomials, and so on. More information on these concepts can be found in the literature (e.g. Golub and Ortega [22]).

Galerkin's method: Before going into the finite element method, we introduce Galerkin's method, which forms the basis of the modern finite element method. In mathematics, in the area of numerical analysis, Galerkin methods are a class of methods for converting a continuous operator problem (such as a differential equation) to a discrete problem. In principle, it is the equivalent of applying the method of variation [23] to a function space, by converting the equation to a weak formulation. Typically one then applies some constraints on the function space to characterize the space with a finite set of basis functions. Often when using a Galerkin method, one also gives the name along with typical approximation methods used, such as the Petrov-Galerkin method or the Ritz-Galerkin method [20].

We will illustrate the general idea about Galerkin's method with the linear two-point boundary-value problem

$$v''(x) + q(x)v = f(x), \quad 0 \leq x \leq 1, \quad (2.35)$$

with its two boundary conditions

$$v(0) = 0, \quad v(1) = 0, \quad (2.36)$$

where, for simplicity, the interval has been taken to be $[0, 1]$.

Suppose that one looks for an approximate solution of (2.35), (2.36) of the form

$$u(x) = \sum_{j=1}^n c_j \phi_j(x), \quad (2.37)$$

where the basis functions ϕ_j , which properties will be described later, satisfy the boundary conditions:

$$\phi_j(0) = \phi_j(1) = 0, \quad j = 1, \dots, n. \quad (2.38)$$

Then, to obtain the approximate solution, we need to determine the coefficients c_1, \dots, c_n in (2.37). One of the approaches to do so is Galerkin's method and consists in the concept of the orthogonality of functions. Two vectors \mathbf{f} and \mathbf{g} are said to be orthogonal if their inner product satisfies

$$(\mathbf{f}, \mathbf{g}) \equiv \mathbf{f}^T \mathbf{g} = \sum_{j=1}^n f_j g_j = 0 \quad (2.39)$$

Now suppose that the components of the vectors \mathbf{f} and \mathbf{g} are the values of two functions f and g at n equally spaced grid points in the interval $[0, 1]$; that is,

$$\mathbf{f} = (f(h), f(2h), \dots, f(nh)), \quad (2.40)$$

$$\mathbf{g} = (g(h), g(2h), \dots, g(nh)), \quad (2.41)$$

where $h = (n + 1)^{-1}$ is the grid-point spacing. The orthogonality relation (2.39) becomes

$$\sum_{j=1}^n f(jh)g(jh) = 0, \quad (2.42)$$

and this relation remains unchanged if we multiply by h :

$$h \sum_{j=1}^n f(jh)g(jh) = 0. \quad (2.43)$$

Now, let $n \rightarrow \infty$ (or, equivalently, let $h \rightarrow 0$). Then, assuming that the functions f and g are integrable, the sum in (2.43) will tend to the integral:

$$\int_0^1 f(x)g(x)dx = 0. \quad (2.44)$$

With this motivation, we define functions f and g to be orthogonal on the interval $[0, 1]$ if the relation (2.44) holds.

The rationale for the Galerkin's approach is as follows. Let the residual function for $u(x)$ be defined by

$$r(x) = u''(x) + q(x)u(x) - f(x), \quad 0 \leq x \leq 1. \quad (2.45)$$

If $u(x)$ were the exact solution of (2.35), then the residual function would be identically zero. Obviously, the residual would then be orthogonal to every function, and, in particular, it would be orthogonal to the set of basis functions. However, we cannot expect $u(x)$ to be the exact solution because we restrict $u(x)$ to be a linear combination of the basis functions. The Galerkin criterion is to choose $u(x)$ so that its residual is orthogonal to all of the basis functions ϕ_1, \dots, ϕ_n :

$$\int_0^1 [u''(x) + q(x)u(x) - f(x)] \phi_i(x) dx = 0, \quad i = 1, \dots, n. \quad (2.46)$$

If we put (2.37) into (2.46) and interchange the summation and integration, we have

$$\sum_{j=1}^n c_j \int_0^1 [\phi_j''(x) + q(x)\phi_j(x)] \phi_i(x) dx = \int_0^1 f(x)\phi_i(x) dx, \quad i = 1, \dots, n. \quad (2.47)$$

This is a system of n linear equations in the n unknowns c_1, \dots, c_n . The computational problem is to first evaluate the coefficients

$$a_{ij} = \int_0^1 [\phi_j''(x) + q(x)\phi_j(x)] \phi_i(x) dx \quad (2.48)$$

and

$$f_i = \int_0^1 f(x)\phi_i(x) dx \quad (2.49)$$

If we integrate the first term in this integral by parts,

$$\int_0^1 \phi_j''(x)\phi_i(x) dx = \phi_j'(x)\phi_i(x)|_0^1 - \int_0^1 \phi_j'(x)\phi_i'(x) dx \quad (2.50)$$

and note that the first term vanishes because ϕ_i is zero at the end points, we can

re-write a_{ij} as

$$a_{ij} = - \int_0^1 \phi_j'(x) \phi_i'(x) dx + \int_0^1 q(x) \phi_j(x) \phi_i(x) dx \quad (2.51)$$

Thus, the system of equations to solve for coefficients c_1, \dots, c_n in Galerkin's method is $\mathbf{A}\mathbf{c} = \mathbf{f}$, with the elements of \mathbf{A} given by (2.51) and those of \mathbf{f} by (2.49).

Finite element definition of basis functions: As mentioned before, the field variables are approximated by a linear combination of known basis functions:

$$u(\mathbf{x}) = \sum_{j=1}^n c_j \phi_j(\mathbf{x}), \quad (2.52)$$

where the summation extends over all nodes j . Hence one basis function is attached to each nodal value or degree of freedom. These functions $\phi_j(\mathbf{x})$ can be quite general, with varying degrees of continuity at the inter-element boundaries.

On the one hand, methods based on defining the basis functions on the whole domain, such as trigonometric functions leading to Fourier series, are used in collocation and spectral methods, where the functions $\phi_j(\mathbf{x})$ can be defined as orthogonal polynomials of Legendre, Chebyshev or similar types. Other possible choices are spline functions for $\phi_j(\mathbf{x})$, leading to spline interpolation methods. In these cases, the coefficients c_j are obtained from the expansions in series of the basis functions.

On the other hand, in standard finite element methods, the basis functions are chosen to be locally defined polynomials within each element, being zero outside the considered element. In addition, the coefficients c_j are the unknown nodal values of the variables u . As a result, the local basis functions satisfy the following conditions on each element (e) , with j being a node of (e) :

- $\phi_j^{(e)}(\mathbf{x}) = 0$ if \mathbf{x} doesn't lie in element (e) .
- $u(\mathbf{x}_j) = c_j$ since c_j are the values of the unknowns at node number j .
- $\phi_j^{(e)}(\mathbf{x}_i) = \delta_{ij}$, where δ_{ij} is the Kronecker delta, for any point \mathbf{x}_i .
- $\sum_j \phi_j^{(e)}(\mathbf{x}) = 1$ for all $\mathbf{x} \in (e)$ to represent exactly a constant function $u(\mathbf{x}) = \text{constant}$.

The global basis function ϕ_j is obtained by assembling the contributions $\phi_j^{(e)}$ of all the elements to which node j belongs. The above condition connects the various basis

functions within an element and the allowed polynomials will be highly dependent on the number of nodes within each element.

Lagrange basis polynomials: In this work, the most commonly used Lagrange basis polynomials are adopted for basis functions. The Lagrange basis polynomials $\phi_j(\xi)$ are defined as follows:

$$\phi_j(\xi) = \prod_{f=1, f \neq j}^k \left(\frac{\xi - \xi_f}{\xi_j - \xi_f} \right) = \frac{\xi - \xi_1}{\xi_j - \xi_1} \cdots \frac{\xi - \xi_{j-1}}{\xi_j - \xi_{j-1}} \frac{\xi - \xi_{j+1}}{\xi_j - \xi_{j+1}} \cdots \frac{\xi - \xi_k}{\xi_j - \xi_k}, \quad (2.53)$$

where ξ is a scalar variable. When we expand the product (2.53), there are two properties fulfilling the conditions for finite element basis functions mentioned before. The first one is that all the terms $(\xi_i - \xi_f)/(\xi_j - \xi_f)$ are equal to one if $i = j$ because the product skips $f = j$. The second one is that, if $i \neq j$, since $f \neq j$ doesn't preclude it, one term in the product will be for $f = i$ zero, making the entire product be zero. Therefore, based on the above two observations, we have

$$\phi_j(\xi_i) = \begin{cases} 1, & \text{if } j = i \\ 0, & \text{if } j \neq i \end{cases} \quad (2.54)$$

1D linear Lagrange element: The simplest element has a piecewise linear basis function and contains two nodes in Figure 2.4. Since the basis functions on each element are locally defined Lagrange polynomials, the universal form through a mapping from (x_{i-1}, x_i) to the ξ -space $(0, 1)$ is taken in terms of (2.53).

$$\phi_1(\xi) = \frac{\xi - \xi_2}{\xi_1 - \xi_2} = \frac{\xi - 1}{0 - 1} = 1 - \xi \quad (2.55)$$

$$\phi_2(\xi) = \frac{\xi - \xi_1}{\xi_2 - \xi_1} = \frac{\xi - 0}{1 - 0} = \xi \quad (2.56)$$

and the condition $\sum_j \phi_j^{(e)}(\xi) = 1$ is easily verified.

Since any linear function is represented exactly on the element, the linear mapping $\xi(x)$ or $x(\xi)$ is expressed as a function of the linear basis functions (2.55), (2.56). Thus, we obtain

$$x(\xi) = \sum_{j=1}^2 x_j \phi_j(\xi) \quad (2.57)$$

1D quadratic Lagrange element: On an element with three nodes $(i-1, i, i+1)$ (Figure 2.5), the basis functions will be second-order polynomials, enabling the

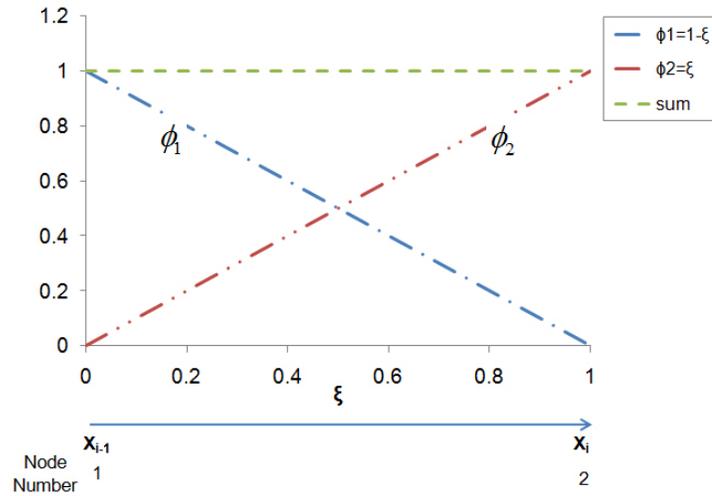


Figure 2.4: One dimensional Lagrange linear element and associated basis functions

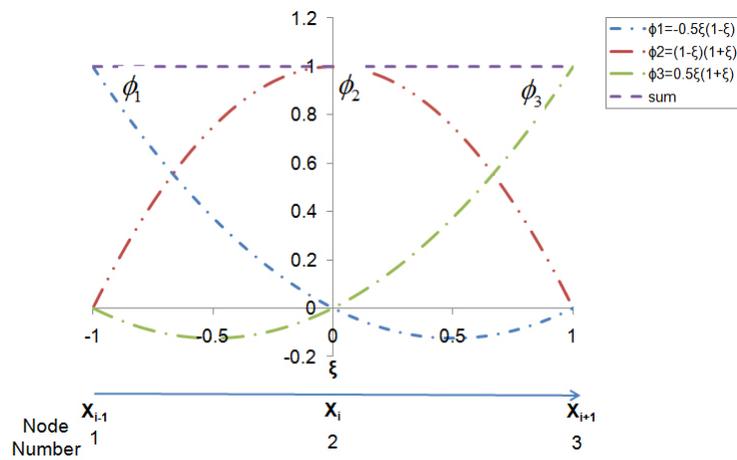


Figure 2.5: One dimensional Lagrange quadratic element and associated basis functions

exact representation of quadratic functions on the element. Since a mapping from (x_{i-1}, x_i, x_{i+1}) to the ξ -space $(-1, 0, 1)$ is defined through a quadratic function $\xi = \xi(x)$, the three basis functions are derived in terms of (2.53)

$$\phi_1(\xi) = \frac{\xi - \xi_2}{\xi_1 - \xi_2} \frac{\xi - \xi_3}{\xi_1 - \xi_3} = \frac{\xi - 0}{-1 - 0} \cdot \frac{\xi - 1}{-1 - 1} = -0.5\xi(1 - \xi) \quad (2.58)$$

$$\phi_2(\xi) = \frac{\xi - \xi_1}{\xi_2 - \xi_1} \frac{\xi - \xi_3}{\xi_2 - \xi_3} = \frac{\xi - (-1)}{0 - (-1)} \cdot \frac{\xi - 1}{0 - 1} = (1 - \xi^2) \quad (2.59)$$

$$\phi_3(\xi) = \frac{\xi - \xi_1}{\xi_3 - \xi_1} \frac{\xi - \xi_2}{\xi_3 - \xi_2} = \frac{\xi - (-1)}{1 - (-1)} \cdot \frac{\xi - 0}{1 - 0} = 0.5\xi(1 + \xi) \quad (2.60)$$

and again the condition $\sum_j \phi_j^{(e)}(\xi) = 1$ is easily verified.

The general mapping between x and ξ is quadratic, and, since any quadratic function $\xi(x)$ on an element can be written as a linear combination of the basis function $\phi_j (j = 1, 2, 3)$, the following mapping can be defined:

$$x(\xi) = \sum_{j=1}^3 x_j \phi_j(\xi) \quad (2.61)$$

where the summation 1, 2, 3 corresponds to nodes $(i-1, i, i+1)$ and the basis functions ϕ_j are given by equations (2.58), (2.59), (2.60).

2D linear Lagrange element: Since two dimensions are considered, we have one linear Lagrange basis function in each direction (i.e. x and y correspond to ξ and η , respectively) to form a bi-linear quadrilateral element (see Figure 2.6). Therefore, since a mapping from the 2D physical coordinates to the ξ - η -space $(1, 1)$, $(1, -1)$, $(-1, -1)$, $(-1, 1)$ is defined through two functions $\xi = \xi(x, y)$ and $\eta = \eta(x, y)$, the four basis functions are defined through the tensor product Lagrange polynomials:

$$\mathbf{L}_\xi \otimes \mathbf{L}_\eta = \mathbf{L}_\xi^T \mathbf{L}_\eta = \begin{bmatrix} \frac{1-\xi}{2} \\ \frac{1+\xi}{2} \end{bmatrix} \begin{bmatrix} \frac{1-\eta}{2} & \frac{1+\eta}{2} \end{bmatrix} = \begin{bmatrix} \frac{1-\xi}{2} \frac{1-\eta}{2} & \frac{1-\xi}{2} \frac{1+\eta}{2} \\ \frac{1+\xi}{2} \frac{1-\eta}{2} & \frac{1+\xi}{2} \frac{1+\eta}{2} \end{bmatrix} \quad (2.62)$$

where $\mathbf{L}_\xi = (l_{\xi 1}, l_{\xi 2})$ is a row Lagrange polynomial vector in the ξ -direction and $\mathbf{L}_\eta = (l_{\eta 1}, l_{\eta 2})$ in the η -direction, and the superscript T represents the vector transpose. Then, we can have

$$\begin{bmatrix} \phi_1(\xi, \eta) & \phi_2(\xi, \eta) \\ \phi_4(\xi, \eta) & \phi_3(\xi, \eta) \end{bmatrix} = \begin{bmatrix} \frac{1-\xi}{2} \frac{1-\eta}{2} & \frac{1-\xi}{2} \frac{1+\eta}{2} \\ \frac{1+\xi}{2} \frac{1-\eta}{2} & \frac{1+\xi}{2} \frac{1+\eta}{2} \end{bmatrix} \quad (2.63)$$

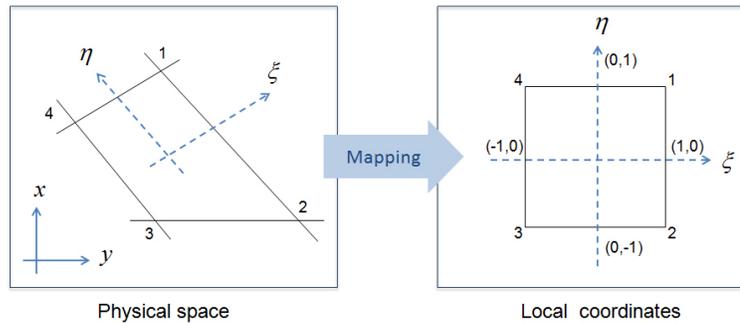


Figure 2.6: Illustration of two dimensional Lagrange bi-linear mapping

Once again, the condition $\sum_j \phi_j^{(e)}(\xi, \eta) = 1$ is easily verified and the following mapping can be defined:

$$\mathbf{x}(\xi, \eta) = \sum_{j=1}^4 \mathbf{x}_j \phi_j(\xi, \eta) \quad (2.64)$$

where \mathbf{x} is a vector with two components (x and y).

Comparison between discontinuous Galerkin and continuous spaces:

The discontinuous Galerkin finite element space (see mathematical specifications in Chen [21]) is very useful for the convection term because its interpolation (basis) functions don't have the C^0 -continuity across inter-element boundaries. More precisely speaking, discontinuous finite element space can easily satisfy the upwind scheme because it can help discretize hyperbolic PDEs by using differencing biased in the direction determined by the sign (+ or -) of characteristic speeds but continuous space cannot. For instance, in Figure 2.7 on the left side, it is clearly seen that the saturation values on the interface between any two cells are discontinuous. Note that, in the same Figure on the right side, results are obtained using continuous Lagrange finite element space with a stabilizing term (see, for example, Appendix A).

On the other hand, discontinuous space is not useful for the diffusion term because the diffusion term represents particles that go in all directions rather than in a particular direction (convection term). This implies that continuous finite element space is a good choice for discretizing the diffusion term. Therefore, if one only considers diffusive transport for all species and one thereby doesn't encounter any stability problems when he uses the general continuous (Lagrange polynomial) finite element space.

In this work, the continuous space is used for saturation but it requires a stabi-

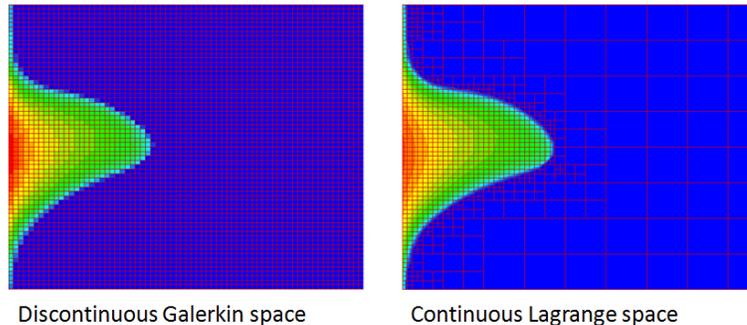


Figure 2.7: Comparison of finite element method between discontinuous Galerkin space and continuous Lagrange space. (see Appendix A or Chueh et al. [3] for further details)

lizing term (artificial viscous term) because the general finite element method with continuous space does nothing for upwinding.

2.6 Artificial diffusive stabilization

When general continuous finite element discretization is adopted for the saturation transport (advection) equation in two-phase flow problems, spurious and unphysical oscillations may appear in the solution, requiring the introduction of a stabilizing (diffusive) term [21]. However, this type of stabilization results in smearing of sharp fronts and can also cause grid-orientation difficulties [21]. Finding the right balance between preserving accuracy and providing stability is essential to the reliability of the numerical solution of the conservation laws. In this work, we implement the artificial diffusion terms proposed by Guermond and Pasquetti [24]. This entropy-based nonlinear viscosity provides a powerful approach yielding both accuracy and stability. First, the artificial viscosity term acts only in the vicinity of strong gradients in the saturation and other discontinuities [24]; secondly, the term does not affect the solution in smooth regions; and finally the scheme offers higher order accuracy and stability than simple upwind schemes [24]. In this thesis, this approach is combined with an IMPES algorithm and we present an extension of shock-type adaptive refinement to saturation gradients to investigate transient transport phenomena in heterogeneous porous media. The use of this shock-type adaptive refinement technique allows us to provide fine-scale resolution locally and to concentrate numerical efforts near the areas where the two-phase interfaces evolve.

Guermond and Pasquetti [24] demonstrate excellent performance and computational results for this scheme and provide details on the derivation. The stabilization term is critical in order to obtain a saturation field that is oscillation free. Spurious oscillations that occur without the stabilization term are illustrated in Chueh et al. [3]. Results discussed in this thesis using the above method are free of such unphysical oscillations.

2.7 Adaptive mesh refinement

Numerical solutions are an approximation of the exact solution. In the course of a computation, we may encounter many possible sources of error. There are three kinds of systematic errors involved (Ferziger and Perić [25]):

1. **Modelling errors:** The exact solution of a mathematical model does not represent the actual flow exactly, as generalizations and assumptions are made to arrive at that model.
2. **Iteration errors:** They describe the difference between the exact solution of the discretized algebraic equation system and the iterative solution.
3. **Discretization errors:** They comprise the difference between the exact (analytical) solution of the model equations and the exact solution of the algebraic system of the discretized equations.

The effect of modelling errors is that the model equations may be fulfilled exactly while the solution is qualitatively wrong. Modelling errors can only be discovered when solutions, where the iteration and discretization errors are very small, are compared with experimental data. Round-off errors are easier to control and usually not problematic, but nonetheless unavoidable. Last, discretization errors are always present due to the nature of this scheme. They decrease, however, as grids get finer and finer in a convergent scheme. This implies that discretization errors only give a rate at which error decreases as smaller element sizes decline.

Adaptive mesh refinement (AMR) techniques can help to minimize the discretization errors. A variety of AMR methods have been proposed depending on the type of physical problem and associated partial differential equations (PDE), and a large body of literature exists for these methods (see, for example, [26, 27, 28, 29] for a

general overview, and [30, 31] for overviews in multiphase flows). In general, the following adaptive methods can be categorized (Ellsiepen [30], Hinkelmann [31]):

1. **h-adaptive methods:** Single elements are refined or coarsened (change of the element size h). This results in a change of the node and element densities at certain locations. During refinement, the element is divided at the midpoint of a side or at the center of mass; during coarsening, the process is the opposite of refinement. The initial elements serve as a reference for the geometry as well as for the associated material properties and therefore must not be coarsened. In the h -adaptive method, the quality of the starting mesh is of great importance, as the angles and aspect ratios of the side lengths of the elements are kept throughout the computation. In some cases, this may be regarded as a disadvantage of this method. Another requirement is as follows: the h -adaptive method needs a dynamic data structure, as the number of elements is constantly changed.
2. **p-adaptive methods:** The order p of the polynomial test functions is changed for selected elements. As a consequence, the formulation for an element gets more elaborate, which may in essence result in a state where solving cost dramatically increases. Furthermore, this method requires a very high regularity of the solution, which may not be suitable for all kinds of problems (e.g. mixed hyperbolic/parabolic problems).
3. **h-p-adaptive methods:** As a combination of the two above-mentioned methods, the element size h as well as the order of the polynomial p are adapted to the local behavior of the solution.
4. **r-adaptive methods:** This method is also called the moving mesh method (see, e.g., Tang and Tang [32], Tan *et al.* [33, 34], Di *et al.* [35, 36, 37], Li and Tang [38] and Zhang *et al.* [39]). The aim is to achieve the best orientation of the mesh with a pre-defined number of degrees of freedom for a user-defined criterion. The nodes of the mesh are moved to the location of the maximum error or in alignment to streamlines, which results automatically in a reduction of nodes in other parts of the domain. An advantage of this method is the simple data storage since the stored structure of the original grids is kept unchanged throughout the computation. On the other hand, however, the r -adaptive method is less flexible than the other methods mentioned above, as

the movement of the nodes can only take place to a certain extent. A further disadvantage lies in the fact that the element matrices have to be computed again at every iteration step, which means that the r-adaptive method is not useful for complex boundaries and should only be used for simple problems.

5. **subgrid methods:** Arbitrarily orientated quadrilateral grids (subgrids) are superimposed over the original grids [40]. These subgrids can have a smaller element size than the underlying basic grid. As the time step is coupled with the element size, the subgrids are computed with a smaller time step. A great limitation of this method is its inability to handle complex computational domains.

Generally, the h - and p -adaptive methods are commonly used in engineering applications. For the two-phase flow problems considered in this work, the h -adaptive method is used. The dynamic data structure provided by deal.II (Bangerth, Hartmann and Kanschat [41]) written in C++ supports this method very well.

In order to refine or coarsen the mesh, error criteria are applied. The error evaluation can be divided into two general types:

- **a priori** error evaluation: these estimations are evaluated before the calculation is started. They use exclusively known data, (e.g. initial and boundary conditions), as well as material properties. A priori estimations are rarely used in practice.
- **a posteriori** error evaluation: here, the error is given in terms of the starting data and computed solution after each time step.

In the thesis, only a posteriori error evaluation is considered. The a posteriori error evaluation can be implemented in two ways: error estimators and error indicators. The former is based on mathematics and the latter is usually physically motivated and obtained by heuristic (try-and-error) methods. Furthermore, a posteriori error estimators and indicators can be divided into the following classes, which will be explained briefly. A detailed description can be found in Verfürth [27] and Ellsiepen [30].

1. **Residual-based error estimator:** With the residual of the strong form of the partial differential equation, the error of the numerical simulation is measured.

This can be regarded as an elementwise controlling of whether or not the partial differential equations are fulfilled by substituting computed solutions into them. The basis of this approach, which was first presented by Babuška and Rheinboldt [42, 43], is the estimation of the error in the energy or L^2 norm via the residual of the finite element solution in the domain and at the boundary. Other scientists (Johnson and Hansbo [44]) extended this approach.

2. **Error estimation by solving local problems:** For small element patches, higher-order solutions are used. In these areas, instead of the given problem, a similar, simple one is solved. The solution is then taken as a reference solution for the error evaluation.
3. **Hierarchical error estimators:** A numerical solution is compared to a solution that was calculated in a relatively higher finite element space (higher order of test functions or finer meshes). The rationale behind introducing this is that the solution in the higher finite element space is expected to be more accurate than in the lower finite element space. For more information, see Ellsiepen [30].
4. **Gradient-based error indicators:** This kind of indicator, which was originally developed for structural mechanics, is often referred to as the Z^2 -error indicator, after the initials of the authors that first introduced this idea in 1987 (Zienkiewicz and Zhu [45]).

The gradients of FE solutions with linear shape functions are usually continuous over the element edges. For certain gradients (for example the tension σ in the case of the Z^2 indicator or the velocity \mathbf{v} in the case of fluid flow), a locally improved solution is calculated by computing a smoothed course of the gradients. With the help of a certain norm of the difference between the gradients of the original and the gradients of the improved solution, the value of the error estimator is calculated.

Error estimators which are based on the energy or L^2 norm are widespread for parabolic and elliptic problems (e.g. Johnson [46] and Papastavrou [47]). They cannot, however, generally be applied for mixed hyperbolic-parabolic equations.

5. **Empirical error indicators:** Here, the error criterion is empirically or heuristically derived. For example, steep gradients or large curvature of the solution can serve as criteria. The empirical error indicator is applied in this work. These are shown in further detail in Appendix A-D.

After each element has been assigned an error value η by an estimator or indicator, the elements are marked for refinement or coarsening. Generally, the error of the element is computed for each element with tolerances to determine whether or not the elements are refined or coarsened:

$$\eta > \textit{tolerance} \rightarrow \textit{refine} \quad (2.65)$$

$$\eta < \textit{tolerance} \rightarrow \textit{coarsen} \quad (2.66)$$

For the choice of tolerances, different possibilities can be categorized (Barlag [48]). For example:

- **Absolute value:** The tolerances are set to fixed values which do not change throughout the simulation. As the magnitude of the errors in the course of simulation is not known and the method is not advisable.
- **Mean value:** The tolerance is given by an arithmetic mean of the errors of all elements. With this method, not only the size of the error, but also the amount of erroneous elements is considered.
- **Maximum value:** The refinement and coarsening tolerances are associated with taking a maximum value of all the indicator values for elements.

Figure 2.8 illustrates the principle of refinement and coarsening for the absolute-value method used in this work. It can be observed that the appropriate choice of refinement and coarsening bounds is very important: if a large peak of the indicator values exists in the domain, other areas with a relatively large indicator value may not be identified. In general, the choice of the values for the refinement and coarsening criteria always depend on the error estimator or indicator as well as on the problem itself.

2.8 Operator splitting techniques

The multi-phase fluid flow models usually represent strongly coupled systems of nonlinear partial differential equations. A number of algorithms (e.g. Godlewski and Raviart [49], Holden and Risebro [50], Kröner [51], LeVeque [16], Morton [52] and Toro [53]) for the solution of the nonlinear partial differential equations have been developed over the years. But still there is an acute need for better solution algorithms

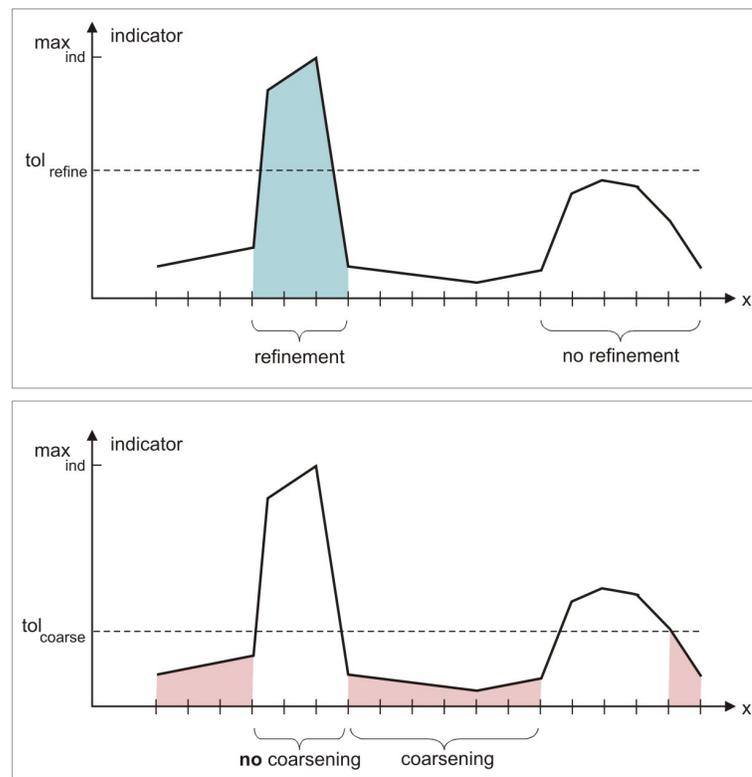


Figure 2.8: Refinement and coarsening bounds for the absolute-value method

as well as mathematical theory which supports them and the models. The different scales of variation appearing in a multi-phase flow model demand an adaptive adjustment of the solution algorithm to the problems. For multi-phase flow in porous media, numerical algorithms which provide local conservation properties are needed. Moreover, the transport part of these models requires accurate pressure and velocity calculations.

There is a close relationship between the challenges in mathematical and numerical modelling and the development of mathematical and numerical tools. An example is provided by operator splitting algorithms which calculate an approximate transport in some parts. These algorithms have become an important part of many solution methods for fluid dynamics (e.g. Beale et al. [54], Glowinski and Pironneau [55], Karniadakis and Henderson [56], Pironneau [57] and Tai and Neittaanmäki [58]).

In general, the following operator splitting methods can be distinguished:

1. **time splitting technique:** Because these numerical methods effectively split the time derivative between the advective and dispersive parts, for example, they are collectively referred to as time-splitting techniques. For non-reactive systems, the splitting results in two partial differential equations which are then solved sequentially at each time step (e.g. Espedal and Ewing [59], Dahle [60], Espedal and Karlsen [61] and Karlsen et al. [62]). For reactive transport, an additional splitting can be added. Split operator methods for reactive transport have been used for a number of years (e.g. Cederberg et al. [63], Walter et al. [64], Zysset et al. [65], and Bell and Binning [66], and James and Jawitz [67]). Roughly speaking, one can say that the splitting algorithms simplify the original problem into a hyperbolic problem, a parabolic problem, and an ordinary differential problem (see Figure 2.9), each of which is solved separately by suitable numerical methods. The main feature of the operator splitting algorithms is the ability to use long time steps and, at the same time, keep the implicit numerical diffusion at a minimum.
2. **non-time splitting technique:** A characteristic of this kind of operator splitting is to separate a non-time-dependent equation from a time-dependent species transport equation. For instance, in two-phase flow in porous media (e.g. Chen et al. [68, 69] and Abreu et al. [70]), there is a non-time-dependent pressure-velocity system appearing. Since the variation of pressure is much slower than that of saturation, a better way is to solve the pressure-velocity part once only

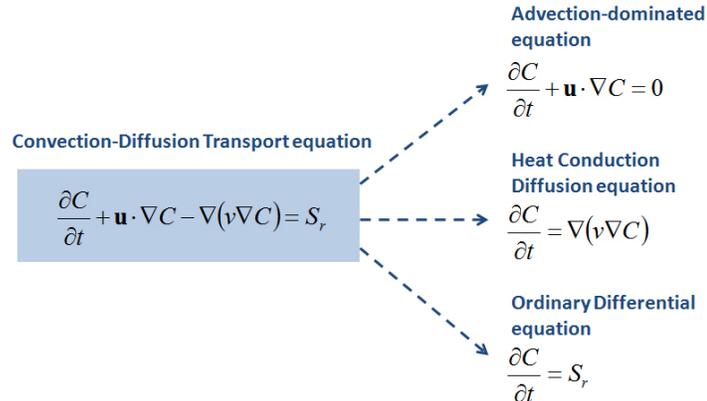


Figure 2.9: Illustration of time operator splitting of the general form of a convection-diffusion equation in a reactive system (C : a scalar variable; ν : a diffusion coefficient; \mathbf{u} : a velocity vector; S_r : a reactive source term)

every few time steps, resulting in a rise in computational efficiency.

At any rate, the two kinds of operator splitting permit independent discretizations which can be applied to the components, allow a great deal of flexibility in selecting appropriate solution methods, and keep numerically implicit diffusion (a computationally expensive part) at a minimum.

2.9 Iterative solution of linear systems

The purpose of this section is to briefly present the basic background of iterative solution of linear systems. The section is neither exhaustive nor complete. Readers are referred to the existing literature for a comprehensive presentation (e.g. Golub and van Loan [71]).

The general form of a linear system is shown as follows:

$$\mathbf{A}\mathbf{X} = \mathbf{B} \tag{2.67}$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is the linear system, $\mathbf{X} \in \mathbb{R}^{n \times 1}$ the solution column vector, and $\mathbf{B} \in \mathbb{R}^{n \times 1}$ the right hand side column vector.

One can distinguish between two different aspects of the iterative solution of a linear system. The first one lies in the particular acceleration technique for a sequence of iteration vectors, which is a technique used to construct a new approximations for

the solution with information from previous approximations. This leads to specific iteration methods that are, in general, of Krylov type such as conjugate gradient or GMRES (Saad and Schultz [72]). The second aspect is the transformation of the given system to one that is solved more efficiently by a particular iteration method. This is called preconditioning. A good preconditioner improves the convergence of the iteration method, sufficiently to overcome the extra cost of its construction and application. Indeed, without a preconditioner, the iterative method might even fail to converge in practice.

The convergence of iterative methods depends on the spectral properties of the linear system matrix and the algorithm itself. The basic idea is to replace the original system (2.67) by the left preconditioned system:

$$\mathbf{P}^{-1}\mathbf{A}\mathbf{X} = \mathbf{P}^{-1}\mathbf{B} \quad (2.68)$$

or the right preconditioned system

$$\mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{X} = \mathbf{B} \quad (2.69)$$

where \mathbf{P}^{-1} is the linear transformation, called preconditioner, and the above right preconditioned system is then solved via $(\mathbf{A}\mathbf{P}^{-1})\mathbf{Y} = \mathbf{B}$ and in turn $\mathbf{X} = \mathbf{P}^{-1}\mathbf{Y}$.

The goal of this preconditioned system is to reduce the condition number of the left or right preconditioned system matrix $\mathbf{P}^{-1}\mathbf{A}$ or $\mathbf{A}\mathbf{P}^{-1}$, respectively. The preconditioned matrix $\mathbf{P}^{-1}\mathbf{A}$ or $\mathbf{A}\mathbf{P}^{-1}$ is almost never explicitly formed. Only the action of applying the preconditioner solve operation \mathbf{P}^{-1} to a given vector need be computed in iterative methods.

From a modern perspective, the general problem of finding an efficient preconditioner is to identify a linear operator \mathbf{P} with the following properties:

- $\mathbf{P}^{-1}\mathbf{A}$ is near to the identity matrix or the diagonally unit-valued matrix.
- the cost of applying the preconditioner is relatively low.
- preconditioning is scalable for parallel algorithms.

For example, the following two linear systems are given respectively as:

$$\begin{pmatrix} 4 & -9 & 2 \\ 2 & -4 & 4 \\ -1 & 2 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \quad (2.70)$$

and

$$\begin{pmatrix} 4 & -9 & 2 \\ 0 & 0.5 & 3 \\ 0 & 0 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2.5 \end{pmatrix} \quad (2.71)$$

Apparently, both the systems have the same solution and the only difference is that equation (2.71) is scaled numerically. However, through using iterative methods, equation (2.71) is much easier to solve than equation (2.70) since the matrix \mathbf{A} in the linear system (2.71) is closer to the identity matrix (see the zero elements in the lower triangular). Based on this observation, one can say that an efficient preconditioner transforms the original matrix \mathbf{A} into one which behaves like the matrix in the linear system (2.71), resulting in a significant rise in computational efficiency.

The role of \mathbf{P} in the iterative method is simple. At each iteration, it is necessary to solve an auxiliary linear system $\mathbf{P}\mathbf{Z}_m = \mathbf{R}_m$ for \mathbf{Z}_m while unnecessary to explicitly invert \mathbf{P} . It should be emphasized that computing the inverse of \mathbf{P} is not mandatory; in fact, the role of \mathbf{P} is to precondition the residual at step m through the solution of the additional system $\mathbf{P}\mathbf{Z}_m = \mathbf{R}_m$. This system should be easier to solve than the original system (2.67).

A broad class of effective preconditioners consists greatly in incomplete factorization of the linear system matrix. Such preconditioners are often referred to as incomplete lower/upper (ILU) preconditioners. ILU preconditioning techniques lie between direct and iterative methods and provide a balance between numerical efficiency and reliability. ILU preconditioners are constructed in the factorized form $\mathbf{P} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}$, where $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{U}}$ are the lower triangular and upper triangular matrices, respectively.

ILU preconditioners are based on the observation that, even though most matrices \mathbf{A} admit an LU factorization $\mathbf{A} = \mathbf{L}\mathbf{U}$, where \mathbf{L} is the unit lower triangular (all main diagonal elements is 1) and \mathbf{U} is the upper triangular, the factors \mathbf{L} and \mathbf{U} often contain too many nonzero terms, making the cost of factorization too expensive in time or memory use, or both. Therefore, ILU preconditioners are used more efficiently

than LU preconditioners.

Chapter 3

Conclusions and future work

3.1 Conclusions

An efficient framework has been presented. This framework is composed of a 2D/3D transient advection-dominated two-phase flow model in heterogeneous porous media and a 2D/3D transient two-phase flow model in heterogeneous porous media with consideration of capillary transport. This framework has three unique features:

- it provides an adequate amount of resolution of the saturation field with the use of diffusive stabilization and with no upwinding while keeping the number of degrees of freedom as low as possible;
- it is developed using the state-of-the-art numerical algorithms to greatly reduce computational costs;
- it is capable of resolving the physical flow representation and analyzing time-dependent two-phase transport phenomena in heterogeneous porous media.

All contributions in this thesis incorporated 2D/3D code implementation and numerical justification and validation are described in section 1.2. Some of these components were carried out in collaboration with Wolfgang Bangerth, who provided key suggestions and important inputs.

Key points are summarized as follows:

Diffusive stabilization: The literature on multi-phase flow in porous media does not contain articles that describe stabilization methods that can provide the same amount of accuracy as what has been proposed in this thesis. This is not a surprise:

The stability issues surrounding the transport equation have long led people to believe that finite volume schemes (or their modern incarnation in the form of DG methods) combined with the usual set of flux/slope limiters are the only way to go. While they can be combined with adaptive meshes (and frequently have been), the relatively low accuracy leads to an unnecessary smearing of the solution. Furthermore, the discontinuity of the solution often makes more comprehensive modeling of additional physical effects (e.g. surface tension) difficult.

Continuous finite element spaces are known to allow simple modeling of additional effects. However, they were considered unstable for the transport equation and there were few good stabilization techniques (streamline diffusion, for example, introduces smearing everywhere, even in regions where this is not necessary). The method that is presented here, therefore, shows a significant step forward: it borrows recent results obtained in entirely different contexts and shows that they lead to attractive methods for multiphase porous media flow problems.

Adaptive mesh refinement: To increase computational speed and, at the same time, increase the accuracy of the results, adaptive mesh refinement is used in this thesis. This approach is to refine and coarsen grids at each time step, depending on where saturation discontinuities propagate. The use of adaptive finite elements provides the present framework with the ability to adapt the computational mesh to the physics of two-phase transport phenomena.

General block preconditioning: To speed up the convergence of linear systems considered in this thesis, general block preconditioning is used for the linear system of mixed formulation for diffusion. The purpose of the preconditioning is to control the computational time within a reasonable frame increasing applicability and reliability. This approach resulted in a 99 percent reduction of computational time.

Adaptive operator splitting: To keep the implicit diffusion (expensive part) at a minimum, operator splitting is used. In addition to this, in order to make the operator splitting adaptive, an a posteriori criterion with a theoretical mathematical proof is used to determine whether or not the computationally expensive part must be solved. This approach resulted in a 91 percent reduction of computational time.

In summary, this thesis integrates above-mentioned numerical methods for two-phase flow in heterogeneous porous media, providing an efficient framework from which more complicated multi-phase flow problems can be solved. The results are encouraging and show that, with these numerical methods, simulations achieve physically accurate flow representations. However, much work remains to be done in the

areas of fuel cell analysis and design before numerical works can achieve their full potential.

3.2 Future work

This thesis has focused mainly on numerical improvements to two-phase flow in heterogeneous porous media. However, the two-phase flow problem is just part of PEM fuel cell modelling and reservoir simulation. With this in mind, the future work is vast and full of challenges and opportunities in the areas of PEM fuel cell modeling.

In order to further increase the range of applicability of this framework, more complicated fuel cell models need to be developed. In this work, a set of equations has been introduced to describe physical two-phase transport phenomena in heterogeneous porous media with capillary transport, i.e. the material properties of the porous media (GDL) have been considered. But without the consideration of electrochemical reactions at catalyst sites, realistic complex interactions between two fluids cannot be expressed in porous media of PEM fuel cells. The electrochemical reaction modelling should be further developed in order to better understand physical transport phenomena including electron and proton transport and to show the flooding effects caused by electrochemically produced water inside porous media.

Many physical problems are required to consider more than two phases in porous media. For instance, in reservoir simulations, water and carbon dioxide both have been used to displace petroleum and to take it out of the ground, requiring the introduction of three phases in porous media. However, in this work, only two phases are taken into account, which is not enough to cover a broad range of physical problems in the real world.

In order to increase further the computational efficiency, parallelization of the program needs to be developed to control the amount of computational time within a reasonable time frame. There are several algorithms to achieve this numerical improvement. One potential avenue is the use of a time decomposition method called “Parareal” or parallel algorithms [73, 74, 75]. Such a parallel algorithm has been applied in reservoir simulations [76] and is expected to speed up simulations substantially.

A summary of possible avenues of research in multi-phase flow in porous media and numerical improvements includes

1. Three phase flow in heterogeneous porous media
2. Electro-chemical reactions in the catalyst sites for PEM fuel cell modelling
3. Parallelization of the program

Appendix A

Multi-level Adaptive Simulation of Transient Two-phase Flow in Heterogeneous Porous Media



Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media

C.C. Chueh^a, M. Secanell^a, W. Bangerth^b, N. Djilali^{a,*}

^a Institute for Integrated Energy Systems and Department of Mechanical Engineering, University of Victoria, PO Box 3055 STN CSC, Victoria, BC, Canada V8W 3P6

^b Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA

ARTICLE INFO

Article history:

Received 17 November 2009
Received in revised form 6 May 2010
Accepted 17 May 2010
Available online 2 June 2010

Keywords:

Two-phase flow
Heterogeneous porous media
Adaptive mesh refinement
Stabilized finite element method

ABSTRACT

An implicit pressure and explicit saturation (IMPES) finite element method (FEM) incorporating a multi-level shock-type adaptive refinement technique is presented and applied to investigate transient two-phase flow in porous media. Local adaptive mesh refinement is implemented seamlessly with state-of-the-art artificial diffusion stabilization allowing simulations that achieve both high resolution and high accuracy. Two benchmark problems, modelling a single crack and a random porous medium, are used to demonstrate the robustness of the method and illustrate the capabilities of the adaptive refinement technique in resolving the saturation field and the complex interaction (transport phenomena) between two fluids in heterogeneous media.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

Modelling of two-phase flow in porous media plays a key role in many engineering areas such as environmental remediation [1,2], oil recovery [3–6] and water management in polymer electrolyte fuel cells [7–12]. In polymer electrolyte fuel cells, which motivated in part the developments described herein, water produced at the cathode as a result of the electrochemical reaction can condense [7,8], and is eventually transported through the porous electrode by a combination of mechanisms, including capillary diffusion. At high reaction rates however, an imbalance between liquid water production and transport can result in flooding of the electrode and, consequently, restricted access of the reactant gases to the reactions sites (catalyst layer); this results in a significant performance drop. Understanding of the two-phase transport processes and design of the porous media to mitigate this are therefore crucial and can be facilitated by robust and physically representative simulations. A number of recent publications have addressed some of the modelling challenges associated with two-phase transport in complex porous media. These include the development of improved numerical schemes for simulation of multi-phase, multi-component processes [13]; interface conditions and linearization schemes [14]; advanced numerical procedures based on high-order time integration schemes [15], fractional flow approaches [16], and reduced degrees of freedom [17]. Theoretical investigations based on pore-network models [18], non-oscillation

central scheme [19], and multi-scale finite volume/element methods [20–24] have also been developed. Helmig et al. [25] note that numerical methods have to be able to capture both advection or diffusion/dispersion dominated processes. An excellent review of the recent modelling efforts and current challenges is provided by Gerritsen and Durlofsky [5]. A key challenge remains the robust and accurate resolution of fine-scale localized flow.

In transient two-phase flow simulations related to petroleum engineering, the implicit pressure and explicit saturation (IMPES) algorithm, originally developed by Sheldon et al. [26] and Stone and Gardner [27], is widely used. The basic idea of this classical method when applied to two-phase flow in porous media is to separate the computation of pressure from that of saturation. Namely, the coupled system is split into a pressure equation and a saturation equation, and the pressure and saturation equations are solved using implicit and explicit time approximation approaches, respectively. This method is easy to implement and efficient to solve, and requires less memory than other methods such as the simultaneous solution method [28]. Detailed discussions of this method can be found in [26,27], and recent algorithmic improvements are discussed in Chen et al. [29,30].

The numerical simulation of transient two-phase flow transport in heterogeneous porous media (Fig. 1) is computationally expensive, and adequate resolution of complex flow features is not always possible, thus compromising the reliability of the results. Achieving physically representative simulations that resolve all salient length and time scales and localized flow features efficiently remains a challenge. An alternative to global mesh refinement which demands very large computing resources, is adaptive mesh

* Corresponding author. Tel.: +1 250 721 6034; fax: +1 250 721 6323.
E-mail address: ndjilali@uvic.ca (N. Djilali).

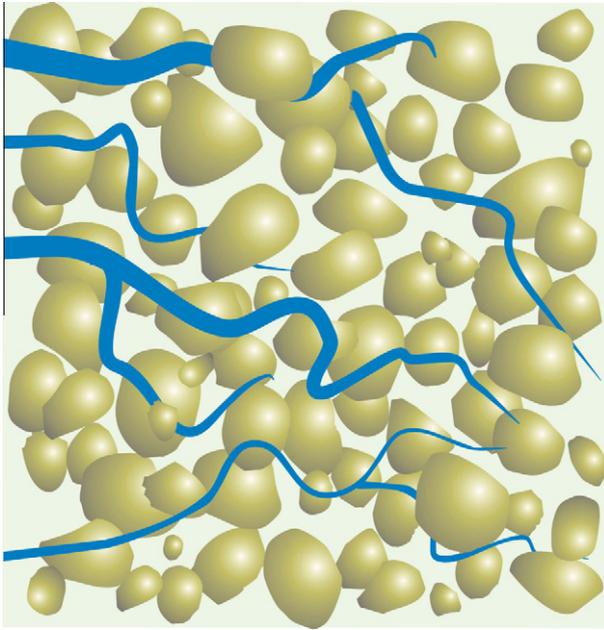


Fig. 1. Schematic of fluid flow in a heterogeneous porous medium.

refinement (AMR). A variety of AMR methods have been proposed depending on the type of physical problem and associated partial differential equations (PDE), and a large body of literature [31–33] exists for these methods. One can use a simple refinement indicator, such as those proposed in [34], to refine and coarsen the mesh at each time step, depending on where the discontinuities (phase boundaries in the present context) propagate. Recent work by Noelle et al. [35] shows that a central scheme with AMR can be implemented on non-conforming 3D Cartesian grids to extend the classical hydrodynamics AMR framework [30]. Smoothness indicators for conservation laws were developed by [36]. Another approach to adaption is the use of a moving-mesh method such as that of Tang and Tang [37] to align the mesh with the important features of the flow. In any case, the major advantages of using grid adaption are high-quality resolution of the physical features as they evolve in space and time while simultaneously reducing computational cost by refining only in areas where necessary and coarsening in areas where unnecessarily fine grids exist. Note that in the context of multi-phase flow the porous medium is frequently strongly heterogeneous within the computational domain. However, as we will show below, there is no need to resolve these heterogeneities everywhere unless they interact with flow fronts. Consequently, adaptive mesh refinement has the potential to significantly reduce the computational cost of multi-phase flow simulations. Despite these obvious advantages, the literature is relatively limited for transient adaptive methods suitable for multi-phase flow in porous media.

When a general continuous finite element discretization is adopted for the saturation transport (advection) equation in two-phase flow problems, spurious and unphysical oscillations appear in the solution, requiring the introduction of a stabilizing (diffusive) term [38]. However, this results in smearing of sharp fronts and can also cause grid-orientation difficulties [38]. Finding the right balance between preserving accuracy and providing stability is therefore of great importance in the numerical solution of conservation laws. In this work, we implement the artificial diffusion terms proposed by Guermond and Pasquetti [39]. This entropy-based nonlinear viscosity provides a powerful approach yielding both accuracy and stability. First, the artificial viscosity term acts only in the vicinity of strong gradients in the saturation and other discontinuities [39]; secondly, the term does not affect the solution in smooth regions; and finally the scheme offers higher order accuracy and sta-

bility than simple upwind schemes [39]. In this paper, this approach is combined with an IMPES algorithm and we present an extension of shock-type adaptive refinement to saturation gradients to investigate transient transport phenomena in heterogeneous porous media. The use of this shock-type adaptive refinement technique allows us to provide fine-scale resolution locally and to concentrate numerical efforts near the area where the two-phase interfaces evolve.

2. Basic numerical model

Let us consider the flow of two incompressible, immiscible fluids in a porous media domain $\Omega \subset \mathbb{R}^2$ in which the movement (displacement) of two fluids is dominated by viscous effects and the effects of gravity and capillary pressure are negligible. The two phases are referred to as wetting and non-wetting, and identified by subscripts w and nw , respectively. Thus in a water–oil system (hydrophilic case), water is the wetting and oil the non-wetting phase; in the air–water system (hydrophobic case), air is the wetting phase and water the non-wetting phase. The mass-averaged velocity with which each of the two phases moves is determined by Darcy's law. It states that the velocity is proportional to the pressure gradient [5]:

$$\mathbf{u}_j = -\frac{k_{rj}(S)}{\mu_j} \mathbf{K} \cdot \nabla p, \quad (1)$$

where \mathbf{u}_j is the velocity of phase $j = w, nw$, \mathbf{K} is the permeability tensor, k_{rj} is the relative permeability of phase j , p is the pressure, and μ_j is the viscosity of phase j . Finally, S is the saturation of the porous media defined as

$$S = \frac{V_w}{V_w + V_{nw}}, \quad (2)$$

where V_w and V_{nw} are the volume fraction of the wetting and non-wetting phases. In this work, the permeability tensor, \mathbf{K} , is a second-order diagonal tensor.

After combining Darcy's law with the mass conservation equation, the following set of equations is obtained [5]:

$$\mathbf{u}_t = -\mathbf{K} \lambda_t(S) \nabla p, \quad (3)$$

$$\nabla \cdot \mathbf{u}_t = q, \quad (4)$$

$$\epsilon \frac{\partial S}{\partial t} + \nabla \cdot (\mathbf{u}_t F(S)) = 0, \quad (5)$$

where λ_t is the total mobility, ϵ is the porosity, F is the fractional flow of the wetting phase, q is a source term, and \mathbf{u}_t is the total velocity. These are given by:

$$\lambda_t(S) = \lambda_w + \lambda_{nw} = \frac{k_{rw}(S)}{\mu_w} + \frac{k_{rnw}(S)}{\mu_{nw}}, \quad (6)$$

$$F(S) = \frac{\lambda_w}{\lambda_t} = \frac{\lambda_w}{\lambda_w + \lambda_{nw}} = \frac{k_{rw}(S)/\mu_w}{k_{rw}(S)/\mu_w + k_{rnw}(S)/\mu_{nw}}, \quad (7)$$

$$\mathbf{u}_t = \mathbf{u}_w + \mathbf{u}_{nw} = -\lambda_t(S) \mathbf{K} \cdot \nabla p. \quad (8)$$

For the sake of simplicity, we consider the case with no source term q . Furthermore the porosity ϵ is set to one as it is essentially a scaling factor that does not affect the qualitative behaviour of Eq. (5). For the purpose of this paper, we will assume the following concrete form for the total mobility λ_t and the fractional flow $F(S)$:

$$\lambda_t(S) = \frac{S^2}{\mu_w} + \frac{(1-S)^2}{\mu_{nw}}, \quad (9)$$

$$F(S) = \frac{S^2}{S^2 + 0.2 \cdot (1-S)^2}, \quad (10)$$

where $\mu_w = 0.2$ and $\mu_{nw} = 1$.

2.1. Permeability of the porous media

In this and the following section, we will discuss the setup of the numerical examples used in Section 5. We consider a heterogeneous, isotropic porous medium with a permeability tensor \mathbf{K} defined as

$$\mathbf{K}(\mathbf{x}) = k(\mathbf{x}) \cdot \mathbf{I} \quad (11)$$

where $k(\mathbf{x})$ is a scalar depending on position \mathbf{x} , and \mathbf{I} is the second-order unit tensor. In order to demonstrate the robustness of the proposed shock-type adaptive refinement technique we consider the two model porous media used previously by Li and Bangerth [40]. The first test case corresponds to a single crack along a sine curve, with a permeability given by

$$k_{sm}(\mathbf{x}) = \max \left\{ \exp \left(- \left(\frac{y - 0.5 - 0.1 \cdot \sin(10x)}{0.1} \right)^2 \right), 0.01 \right\} \quad (12)$$

and $k(\mathbf{x})$ in Eq. (11) is replaced by $k_{sm}(\mathbf{x})$ for the first test case. Taking the maximum in Eq. (12) ensures boundedness of $k(\mathbf{x})$ from below.

The second test case corresponds to a porous medium with random permeability prescribed using

$$k_{rm}(\mathbf{x}) = \min \left\{ \max \left\{ \sum_{l=1}^N \Psi_l(\mathbf{x}), 0.01 \right\}, 4 \right\} \quad (13)$$

and

$$\Psi_l(\mathbf{x}) = \exp \left(- \left(\frac{|\mathbf{x} - \mathbf{x}_l|}{0.05} \right)^2 \right) \quad (14)$$

where again $k_{rm}(\mathbf{x})$ is substituted for $k(\mathbf{x})$ in Eq. (11) for the second test case, and the centers \mathbf{x}_l are N randomly chosen locations inside the domain. This function models a domain in which there are $N=40$ centers of higher permeability, representing regions with cracks embedded in a matrix of intact background rock.

The permeability fields for these two testcases are shown in Fig. 2.

2.2. Initial and boundary conditions

The simulations are performed in a computational domain $\Omega = [0, 1] \times [0, 1]$ for $t \in [0, T]$. The initial condition is $S(\mathbf{x}, 0) = 0$, i.e. the reservoir contains only non-wetting fluid. Initial conditions for pressure or velocity are not necessary because the equations do

not contain time derivatives of these variables. Finally, the following pressure boundary condition is used:

$$p(\mathbf{x}, t) = 1 - x \quad \text{on } \partial\Omega \quad (15)$$

i.e. a linear pressure drop is assumed on the boundaries. Boundary conditions for the saturation need only be specified on the inflow part of the boundary given by

$$\Gamma_{in}(t) = \{\mathbf{x} \in \partial\Omega : \mathbf{n} \cdot \mathbf{u}_t < 0\}. \quad (16)$$

On the inflow boundary, we assume that wetting fluid enters the domain from the left, and consequently prescribe the following saturation values:

$$S(\mathbf{x}, t) = 1 \quad \text{on } \Gamma_{in}(t) \cap \{x = 0\}, \quad (17)$$

$$S(\mathbf{x}, t) = 0 \quad \text{on } \Gamma_{in}(t) \setminus \{x = 0\}. \quad (18)$$

3. Finite element approximations and numerical methods

The system of partial differential equations (PDE) Eqs. (3)–(5) that describes two-phase transport in a hydrophilic medium is discretized using a mixed finite element method [41,38]. The numerical solutions shown below were obtained using a C++ program based on the deal.II library [42]. The implementation discussed here also uses parts of the step-21, step-31 and step-33 tutorial programs of this library [40,43,44] but extends the functionality compared with all of these programs.

3.1. Time discretization

A standard implicit pressure and explicit saturation (IMPES) algorithm [26,27] is used for time discretization in conjunction with an implicit Euler method. In this algorithm, the pressure and velocity equations are first solved implicitly, and then the saturation equation is solved using an explicit time stepping method.

Using this time discretization, we obtain the following set of equations for each time step:

$$\mathbf{u}_t^{(n+1)} + \lambda_t(S^{(n)}) \mathbf{K} \nabla p^{(n+1)} = 0, \quad (19)$$

$$\nabla \cdot \mathbf{u}_t^{(n+1)} = q, \quad (20)$$

$$\epsilon \left(\frac{S^{(n+1)} - S^{(n)}}{\Delta t^{(n+1)}} \right) + \mathbf{u}_t^{(n+1)} \cdot \nabla F(S^{(n)}) + F(S^{(n)}) \nabla \cdot \mathbf{u}_t^{(n+1)} = 0, \quad (21)$$

where superscripts (n) and ($n+1$) represent physical quantities existing at times $t^{(n)}$ and $t^{(n+1)}$, respectively, and $\Delta t^{(n)} = t^{(n+1)} - t^{(n)}$ is the length of the n th time step.

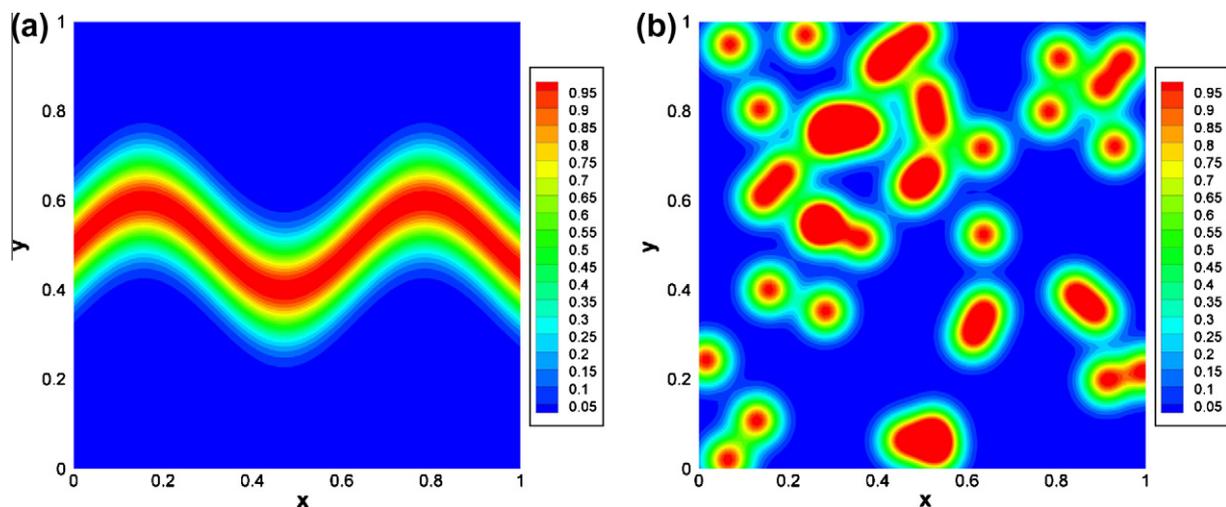


Fig. 2. Permeability fields for (a) single crack medium (b) random medium.

Using the fact that $\nabla \cdot \mathbf{u}_t = q$, Eq. (21) becomes

$$\epsilon \left(\frac{S^{(n+1)} - S^{(n)}}{\Delta t^{(n+1)}} \right) + \mathbf{u}_t^{(n+1)} \cdot \nabla F(S^{(n)}) + F(S^{(n)})q = 0. \quad (22)$$

To ensure stability of this time stepping scheme, the time step has to be chosen in accordance with the spatial discretization and satisfy the Courant–Friedrichs–Lewy (CFL) condition. In our numerical experiments, we choose the time step adaptively as

$$\Delta t^{(n+1)} = \frac{\min_K h_K}{7 \|\mathbf{u}_t^{(n+1)}\|_{L^\infty(\Omega)}}, \quad (23)$$

where h_K denotes the diameter of cell K . This choice is sufficient to satisfy both the CFL condition that results from the advection of the saturation as well as from the artificial diffusion term discussed below, while our numerical experiments indicate that larger values violate these conditions.

Fig. 3 shows the size of the time steps chosen by this criterion for the two testcases considered below, and compares it with the time step chosen in the lower-order method used in [40]. We need to choose our time step smaller since we use a piecewise linear instead of the piecewise constant saturation approximation. The higher spatial order then requires a correspondingly smaller time step for stability.

3.2. Weak form and spatial discretization for the pressure/velocity part

By multiplying Eqs. (19) and (20) with test functions \mathbf{v} and w respectively and then integrating terms by parts as necessary,

the weak form of the problem reads: find \mathbf{u}, p so that for all test functions \mathbf{v}, w there holds

$$\left((\mathbf{K}\lambda_t(S^{(n)}))^{-1} \mathbf{u}_t^{(n+1)}, \mathbf{v} \right)_\Omega - (p^{(n+1)}, \nabla \cdot \mathbf{v})_\Omega = -(p^{(n+1)}, \mathbf{n} \cdot \mathbf{v})_{\partial\Omega}, \quad (24)$$

$$-(\nabla \cdot \mathbf{u}_t^{(n+1)}, w)_\Omega = -(q, w)_\Omega. \quad (25)$$

Here, \mathbf{n} represents the unit outward normal vector to $\partial \Omega$ and the pressure $p^{(n+1)}$ can be prescribed weakly on the boundary $\partial\Omega$.

We use continuous finite elements to discretize the velocity and pressure equations. Specifically, we use mixed finite elements to ensure high order approximation for both vector (e.g. a fluid velocity) and scalar variables (e.g. pressure) simultaneously. For saddle point problems, it is well established that the so-called Babuska–Brezzi or Ladyzhenskaya–Babuska–Brezzi (LBB) conditions [41,38] need to be satisfied to ensure stability of the pressure-velocity system. These stability conditions are satisfied in the present work by using elements for velocity that are one order higher than for the pressure, i.e. $u_h \in Q_{p+1}^d$ and $p_h \in Q_p$, where $p = 1$, d is the space dimension, and Q_s denotes the space of tensor product Lagrange polynomials of degree s in each variable.

3.3. Stabilization, weak form and spatial discretization of the saturation transport equation

Advection problems such as the transport equation for the saturation are frequently discretized by finite volume schemes (or their more recent re-formulation in the form of discontinuous Galerkin (DG) finite element methods [38]). In contrast to this, we use continuous finite elements to discretize the saturation

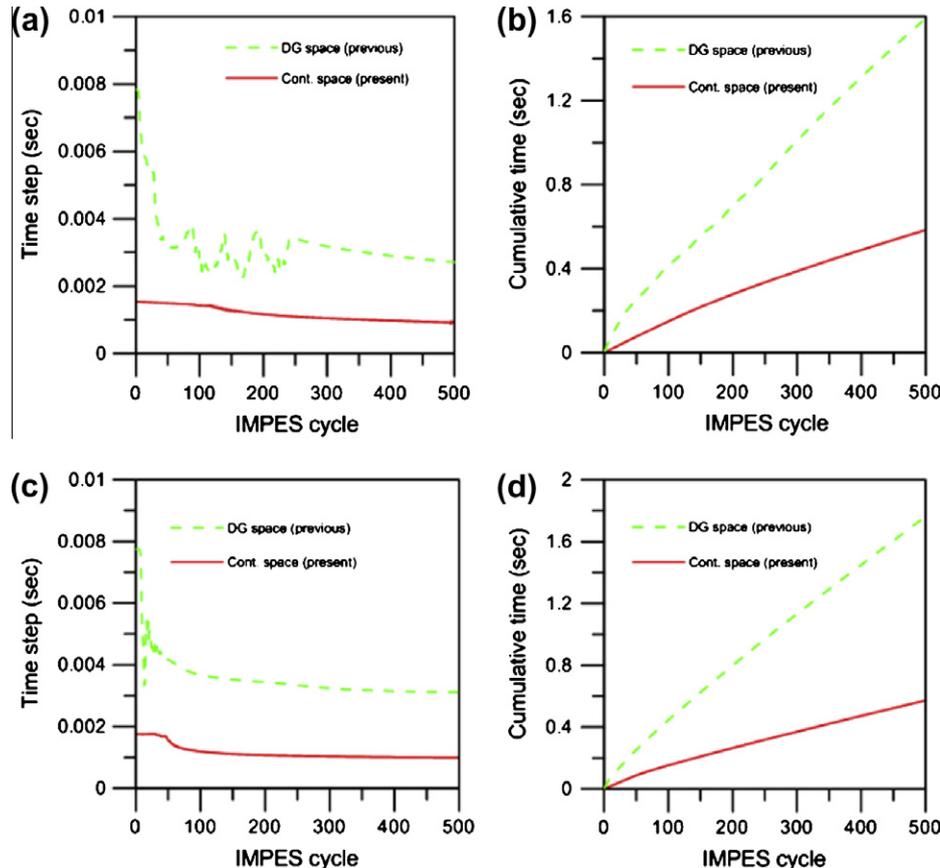


Fig. 3. Comparison of adaptive time step size between the present and previous work by Li and Bangerth [40]. (a) Time step vs. IMPES cycle for the single crack medium. (b) Cumulative time vs. IMPES cycle for the single crack medium. (c) Time step vs. IMPES cycle for the random medium. (d) Cumulative time vs. IMPES cycle for the random medium.

equation, i.e. $S_h \in Q_1$. This choice allows the simple inclusion of a capillary pressure (diffusion) term in the future: The discretization of the diffusion (Laplace) operator using DG methods leads to a significant number of additional terms that need to be integrated on each face between cells. Discontinuous Galerkin finite elements also have the drawback that the use of numerical fluxes introduces an additional numerical diffusion that acts everywhere.

These problems can be avoided using a continuous finite element space, although we still have to add some form of stabilization to make the scheme stable. We add such a stabilization in the form of a nonlinear viscosity to (22), i.e. we solve

$$\epsilon \left(\frac{S^{(n+1)} - S^{(n)}}{\Delta t^{(n+1)}} \right) + \mathbf{u}_t^{(n+1)} \cdot \nabla F(S^{(n)}) - \nabla(v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)})) \nabla S^{(n)} = -F(S^{(n)})q, \tag{26}$$

where $v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)})$ is the artificial viscosity, $S^{(a)}$ is an ‘‘average’’ saturation defined as

$$S^{(a)} = \frac{S^{(n)} + S^{(n-1)}}{2} \tag{27}$$

and $S^{(ex)}$ is an ‘‘extrapolated’’ saturation value given by

$$S^{(ex)} = \left(1 + \frac{\Delta t^{(n)}}{\Delta t^{(n-1)}} \right) S^{(n)} - \frac{\Delta t^{(n)}}{\Delta t^{(n-1)}} S^{(n-1)}. \tag{28}$$

Note that we treat both advection as well as the artificial diffusion term explicitly in time to avoid having to re-assemble the corresponding matrices in each time step. The stabilization factor $v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)})$ is chosen in such a way that, if the discretized saturation S satisfies the original Eq. (5) exactly, the artificial diffusion term is zero. In other words, this term acts primarily in the vicinity of discontinuities in S (or other dependent variables) where in any case the numerical approximation becomes less accurate, but does not affect the areas where the saturation varies smoothly and the numerical approximation is accurate. The literature contains a number of approaches to achieve this. Here we adopt a scheme developed by Guermond and Pasquetti [39] which builds on a suitably defined residual and a limiting procedure for the additional viscosity. The artificial viscosity is a piecewise constant function defined on each cell K with diameter h_K as

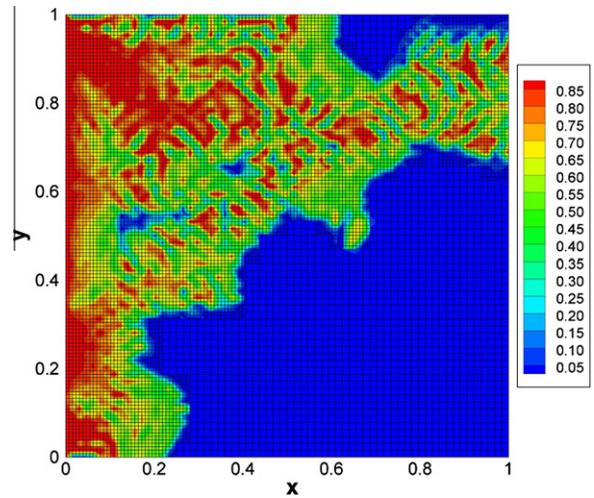


Fig. 4. Spurious oscillations in the saturation solution for the random media case in the absence of a stabilization term.

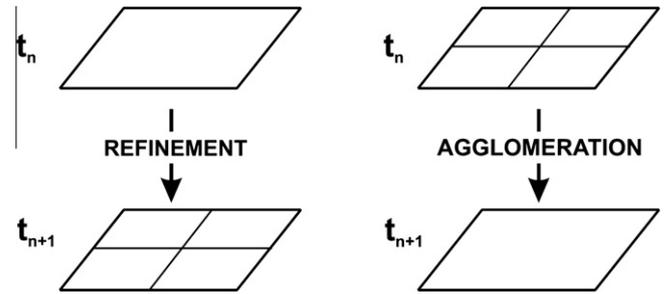


Fig. 5. Illustration of local mesh adaptation process.

$$v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)})|_K = \beta \|\mathbf{u}_t^{(n+1)}\|_{L^\infty(K)} \times \min \left\{ h_K, h_K^\alpha \frac{\|\text{Res}(\mathbf{u}_t^{(n+1)}, S^{(a)})\|_{L^\infty(K)}}{c(\mathbf{u}_t^{(n+1)}, S^{(ex)})} \right\} \tag{29}$$

where α is a stabilization exponent that we chose to be $\alpha = 1$, and β is a user-defined dimensionless stabilization constant that is chosen

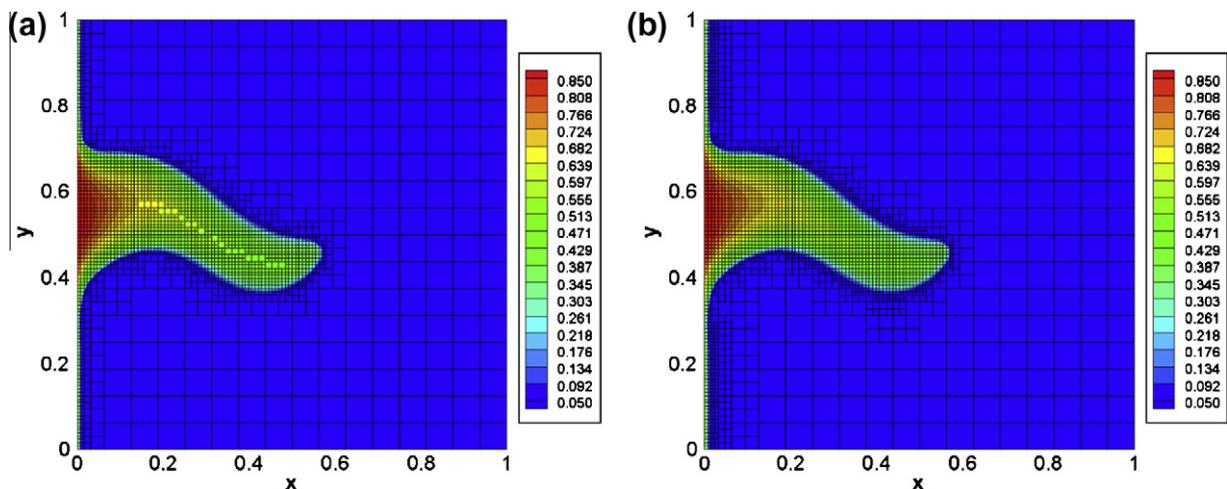


Fig. 6. Effect of refinement indicator threshold: saturation field and adaptive mesh for single crack medium at $t = 0.359$ s. (a) Cells are coarsened if $\eta_K < 1.1$ and refined if $\eta_K > 1.7$ (2608 computational cells). (b) Threshold values for coarsening and refinement are chosen as 0.21 and 0.28 (2938 computational cells).

as small as possible but as large as necessary to avoid unphysical oscillations in the solution. Following Guermond and Pasquetti [39] as well as Kronbichler and Bangerth [44], the velocity and saturation global normalization constant, $c(\mathbf{u}_t^{(n+1)}, S^{(ex)})$, and the residual $\text{Res}(\mathbf{u}_t^{(n+1)}, S^{(a)})$ are given by

$$c(\mathbf{u}_t^{(n+1)}, S^{(ex)}) = c_R \|\mathbf{u}_t^{(n+1)}\|_{L^\infty(\Omega)} \text{var}(S^{(ex)}) |\text{diam}(\Omega)|^{\alpha-2}, \tag{30}$$

$$\text{Res}(\mathbf{u}_t^{(n+1)}, S^{(a)}) = \left(\epsilon \left(\frac{S^{(n)} - S^{(n-1)}}{\Delta t^{(n)}} \right) + \mathbf{u}_t^{(n+1)} \cdot \nabla F(S^{(a)}) + (S^{(a)})^q \right) \cdot (S^{(a)})^{\alpha-1}, \tag{31}$$

where c_R is a dimensionless user-defined normalization constant, $\text{diam}(\Omega)$ is the diameter of the domain and $\text{var}(S^{(ex)})$ is the range

Table 1
Parameters used in the model

Parameter	Symbol	Value	Units
Porosity	ϵ	1.0	–
Viscosity (wetting)	μ_w	0.2	kg m ⁻¹ s ⁻¹
Viscosity (non-wetting)	μ_{nw}	1.0	kg m ⁻¹ s ⁻¹
Stabilization exponent	α	1.0	–
Stabilization constant	β	0.4	–
Normalization constant	c_R	1.0	–

of the extrapolated saturation values in the entire computational domain Ω , given mathematically by

$$\text{var}(S^{(ex)}) = \max_{\Omega} S^{(ex)} - \min_{\Omega} S^{(ex)}. \tag{32}$$

Guermond and Pasquetti [39] demonstrate excellent performance and computational results for this scheme and provide details on the derivation. The stabilization term is critical in order to obtain a saturation field that is oscillation free. Spurious oscillations that occur without the stabilization term are illustrated in Fig. 4. Results discussed subsequently using the above method are free of such unphysical oscillations.

With the artificial diffusion term defined as above, we obtain the weak form of our discrete equations by multiplying Eq. (26) with a test function σ and integrating by parts as necessary. We then obtain

$$\begin{aligned} & (\epsilon S^{(n+1)}, \sigma)_{\Omega} - \Delta t^{(n+1)} (F(S^{(n)}) \mathbf{u}_t^{(n+1)}, \nabla \sigma)_{\Omega} + \Delta t^{(n+1)} (F(S^{(n)}) (\mathbf{n} \cdot \mathbf{u}_t^{(n+1)}), \sigma)_{\partial \Omega} \\ & = (\epsilon S^{(n)}, \sigma)_{\Omega} - \Delta t^{(n+1)} (v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)}) \nabla S^{(n)}, \nabla \sigma)_{\Omega} \\ & \quad + \Delta t^{(n+1)} (\mathbf{n} \cdot v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)}) \nabla S^{(n)}, \sigma)_{\partial \Omega}. \end{aligned} \tag{33}$$

In Eq. (33), there are two boundary integral terms corresponding to Dirichlet and Neumann boundary conditions. In the present work, we require only Dirichlet boundary conditions for saturation,

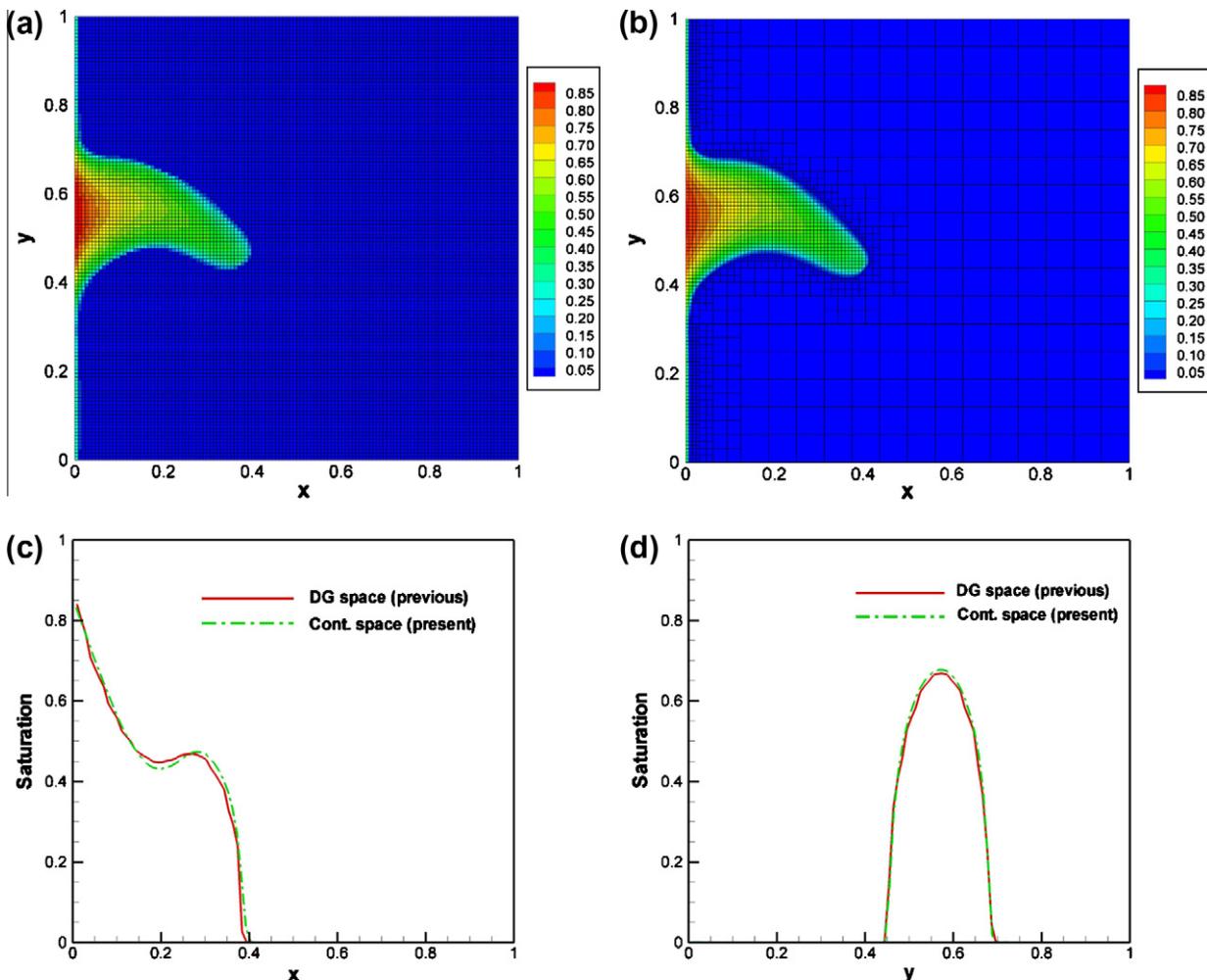


Fig. 7. Numerical results for the single crack medium in comparison with previous work of Li and Bangerth [40] at $t = 0.154$ s. (a) Saturation field using discontinuous Galerkin space (previous work). (b) Saturation field using continuous space with the stabilized term (present work). (c) Saturation profile along $y = 0.5$ (d) Saturation profile along $x = 0.1$.

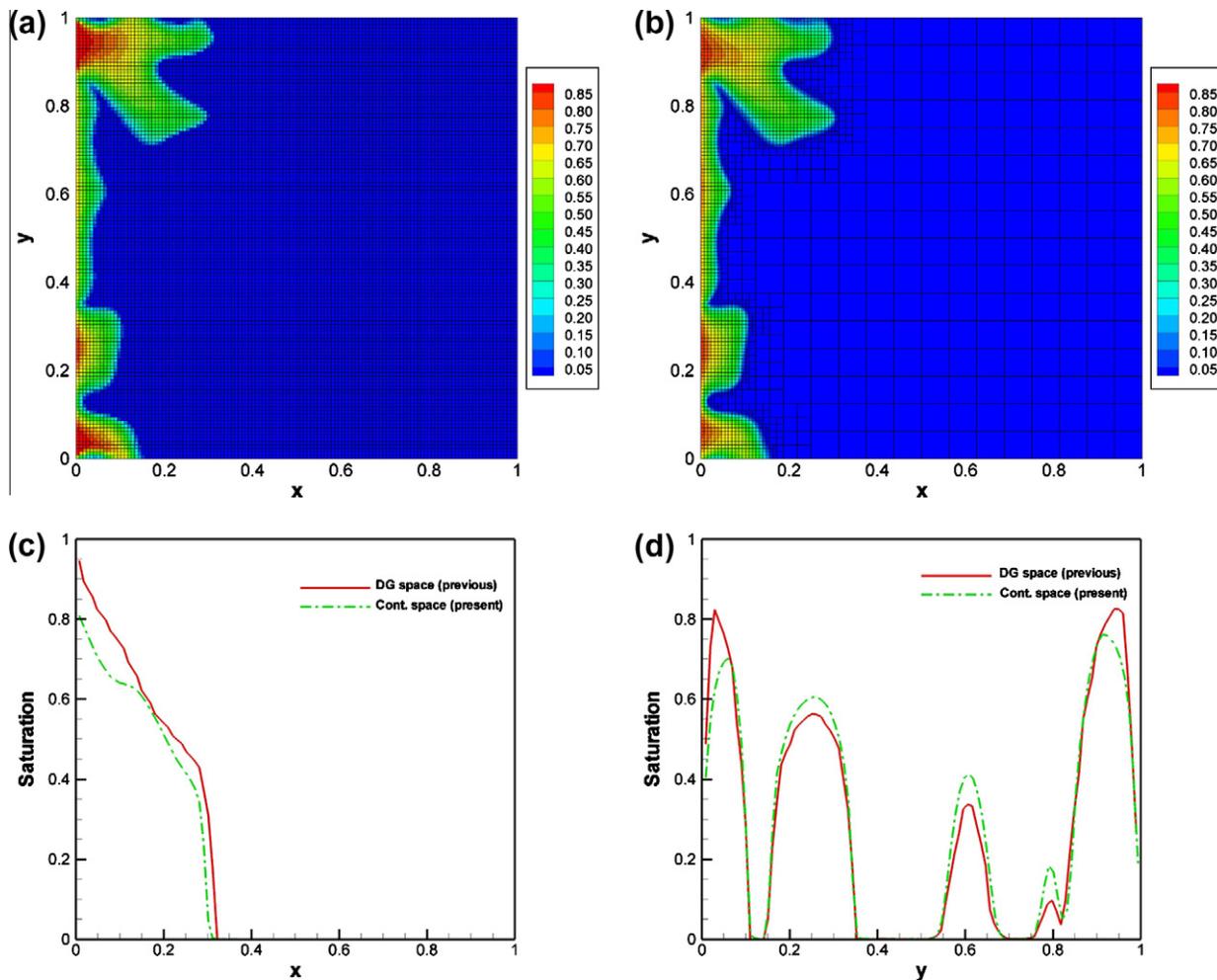


Fig. 8. Numerical results for the random medium in comparison with previous work of Li and Bangerth [40] at $t = 0.249$ s. (a) Saturation field using discontinuous Galerkin space (previous work). (b) Saturation field using continuous space with the stabilized term (present work). (c) Saturation profile along $y = 0.95$. (d) Saturation profile along $x = 0.05$.

as is described in Section 2.2, and therefore the Neumann boundary integral term vanishes.

Since the Dirichlet boundary conditions for saturation are only imposed on the inflow boundaries, the third term on the left hand side of Eq. (33) needs to be split further into two parts:

$$\begin{aligned} & \Delta t^{(n+1)} (F(S^{(n)}) (\mathbf{n} \cdot \mathbf{u}_t^{(n+1)}), \sigma)_{\partial\Omega} \\ &= \Delta t^{(n+1)} (F(S_{(+)}^{(n)}) (\mathbf{n} \cdot \mathbf{u}_{t(+)}^{(n+1)}), \sigma)_{\partial\Omega_{(+)}} + \Delta t^{(n+1)} (F(S_{(-)}^{(n)}) (\mathbf{n} \cdot \mathbf{u}_{t(-)}^{(n+1)}), \sigma)_{\partial\Omega_{(-)}}, \end{aligned} \quad (34)$$

where $\partial\Omega_{(-)} = \{\mathbf{x} \in \partial\Omega : \mathbf{n} \cdot \mathbf{u}_t < 0\}$ and $\partial\Omega_{(+)} = \{\mathbf{x} \in \partial\Omega : \mathbf{n} \cdot \mathbf{u}_t > 0\}$ represent inflow and outflow boundaries, respectively. We choose values using an upwind formulation, i.e. $S_{(+)}^{(n)}$ and $\mathbf{u}_{t(+)}^{(n+1)}$ correspond to the values taken from the present cell, while the values of $S_{(-)}^{(n)}$ and $\mathbf{u}_{t(-)}^{(n+1)}$ are those taken from the neighboring boundary $\partial\Omega_{(-)}$.

3.4. Linear system

Discretization of Eqs. (24), (25) and (33) yields the following linear system that needs to be solved for time step $n + 1$:

$$\begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{H} & \mathbf{0} & \mathbf{M}^S \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n+1)} \\ \mathbf{P}^{(n+1)} \\ \mathbf{S}^{(n+1)} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{F}_2 \\ \mathbf{F}_3 \end{pmatrix}, \quad (35)$$

where the individual matrices and vectors are defined as follows using shape functions \mathbf{v}_i for velocity, and ϕ_i for both pressure and saturation:

$$\mathbf{M}_{ij}^u = ((\mathbf{K}\lambda_t(S^{(n)}))^{-1} \mathbf{v}_i, \mathbf{v}_j)_{\Omega}, \quad \mathbf{M}_{ij}^S = (\epsilon\phi_i, \phi_j)_{\Omega}, \quad (36)$$

$$\mathbf{B}_{ij} = -(\nabla \cdot \mathbf{v}_j, \phi_i)_{\Omega}, \quad \mathbf{H}_{ij} = -\Delta t^{(n+1)} (F(S^{(n)}) \mathbf{v}_i, \nabla \phi_j)_{\Omega}, \quad (37)$$

$$(\mathbf{F}_2)_i = -(F(S^{(n)}) q, \phi_i)_{\Omega}, \quad (38)$$

and

$$(\mathbf{F}_3)_i = (\epsilon S^{(n)}, \phi_i)_{\Omega} - \Delta t^{(n+1)} (v(\mathbf{u}_t^{(n+1)}, S^{(a)}, S^{(ex)}) \nabla S^{(n)}, \nabla \phi_i)_{\Omega} \quad (39)$$

We will solve these equations by first computing the updated pressures $\mathbf{P}^{(n+1)}$ and then the velocities $\mathbf{U}^{(n+1)}$. The final step is to solve for updated saturation $\mathbf{S}^{(n+1)}$. To this end, we can form the Schur complement (i.e., do a block elimination, see [45]) of the top left 2×2 block of the matrix to obtain

$$\mathbf{B}(\mathbf{M}^u)^{-1} \mathbf{B}^T \mathbf{P}^{(n+1)} = -\mathbf{F}_2, \quad (40)$$

$$(\mathbf{M}^u) \mathbf{U}^{(n+1)} = -\mathbf{B}^T \mathbf{P}^{(n+1)}. \quad (41)$$

We can therefore obtain the pressure and velocity solution by first inverting the Schur complement matrix $\mathbf{S} = \mathbf{B}(\mathbf{M}^u)^{-1} \mathbf{B}^T$ and next the mass matrix \mathbf{M}^u on the velocity space. Both matrices are symmetric and positive definite and therefore amenable to the

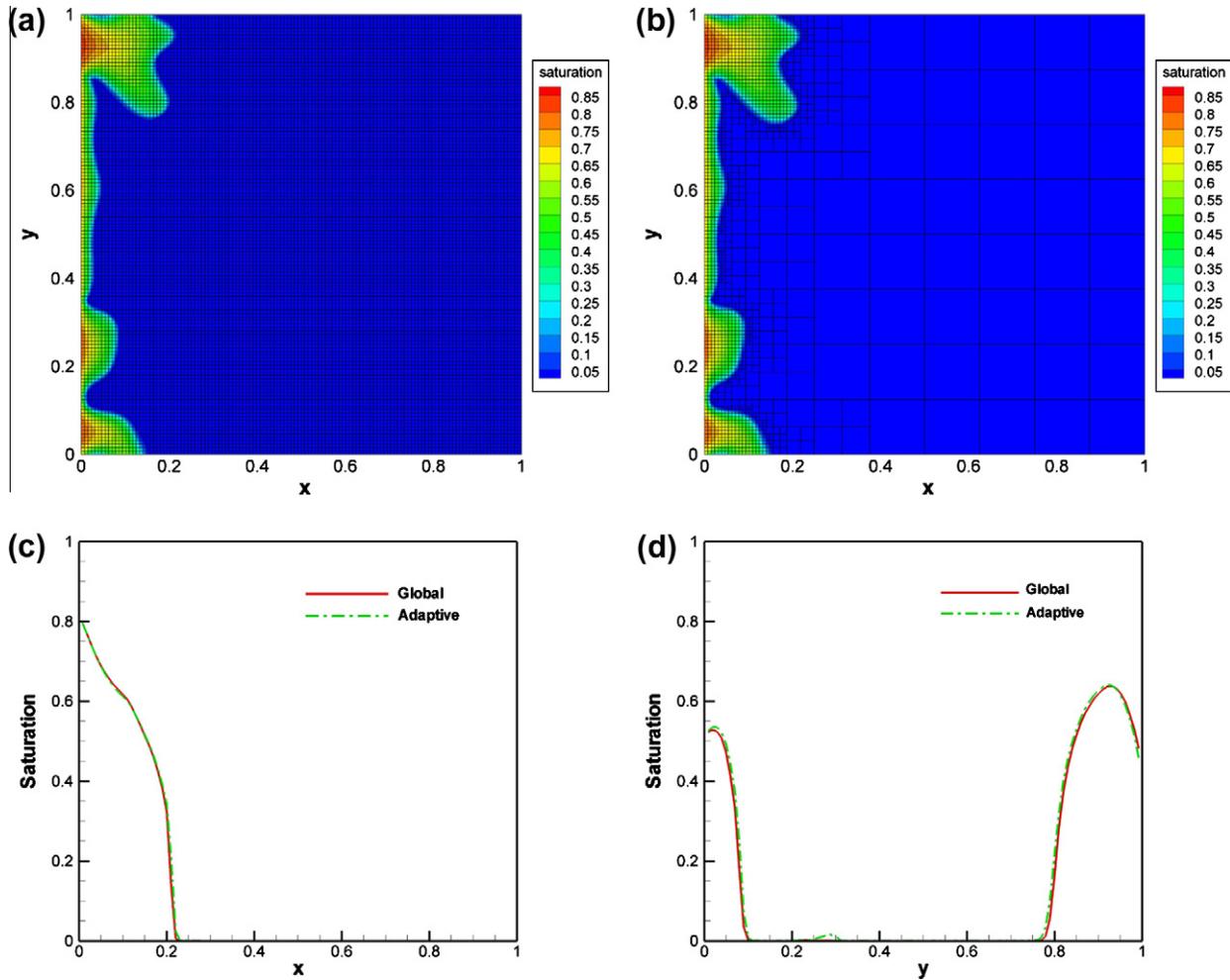


Fig. 9. Comparison of computational mesh distribution and saturation fields at $t = 0.278$ s using the continuous discretization. (a) Uniform mesh with 16,384 elements and 165,380 degrees of freedom. (b) Adaptive mesh with 1981 elements and 7924 degrees of freedom. (c) Saturation profile along $y = 0.95$. (d) Saturation profile along $x = 0.15$.

conjugate gradient method. There are various strategies to precondition the Schur complement solve, using the fact that \mathbf{S} is a discrete representation of the Laplace operator. We choose to precondition with $\tilde{\mathbf{S}}^{-1} = [\mathbf{B}(\text{diag} \mathbf{M}^u)^{-1} \mathbf{B}^T]^{-1}$ since $\tilde{\mathbf{S}}$ is a matrix that is cheap to apply and therefore significantly simpler to invert than \mathbf{S} . At the same time, the preconditioned matrix $\tilde{\mathbf{S}}^{-1} \mathbf{S}$ has a nearly constant condition number and can consequently be solved in a constant number of iterations, irrespective of the refinement level [46]. An alternative is to precondition with an (approximate) inverse of a matrix obtained from a primal (non-mixed) discretization of the Laplace operator.

Once the velocity is available, we can assemble \mathbf{H} and \mathbf{F}_3 (which depend on $\mathbf{u}^{(n+1)}$ implicitly) and solve for the saturations as

$$\mathbf{M}^S \mathbf{S}^{(n+1)} = \mathbf{F}_3 - \mathbf{H} \mathbf{U}^{(n+1)}. \quad (42)$$

4. Adaptive mesh refinement strategy

In analogy with widely used schemes in compressible flow problems, the adaptive refinement technique adopted here is based upon the transient saturation gradient. Once a solution $S^{(n)}$ is computed on a given mesh, we compute a refinement indicator

$$\eta_K = |\nabla_p S_p^{(n+1)}(\mathbf{x}_K)| \quad (43)$$

where \mathbf{x}_K is the cell center. Here,

$$S_p^{(n+1)} = S^{(n)} + \Delta t^{(n+1)} \frac{S^{(n)} - S^{(n-1)}}{\Delta t^{(n+1)}} = 2S^{(n)} - S^{(n-1)} \quad (44)$$

is a predictor for the saturation profile in the next time step. We choose a predicted saturation profile for our refinement indicator since the mesh so created will be used to discretize future time steps, not the current one: we want to refine ahead of a front, rather than propagating into a coarse set of elements and thereby smearing the saturation solutions.

With the refinement indicators η_K so defined, refinement proceeds along the following algorithm [33,42]:

- (1) If η_K is larger than a threshold θ_r , then mark the cell K for refinement if the resulting children will not exceed the chosen maximum refinement level.
- (2) If η_K is smaller than a threshold θ_c , then mark the cell K for coarsening if the resulting cell is not coarser than the initial mesh.
- (3) If for a non-active cell not all children are marked for coarsening, then remove coarsening flags from all children.
- (4) Mark additional cells for refinement to ensure that the resulting mesh has no interfaces between cells that differ in refinement by more than one level.
- (5) Refine and coarsen all cells marked as above.

This adaptive procedure is illustrated graphically in Fig. 5.

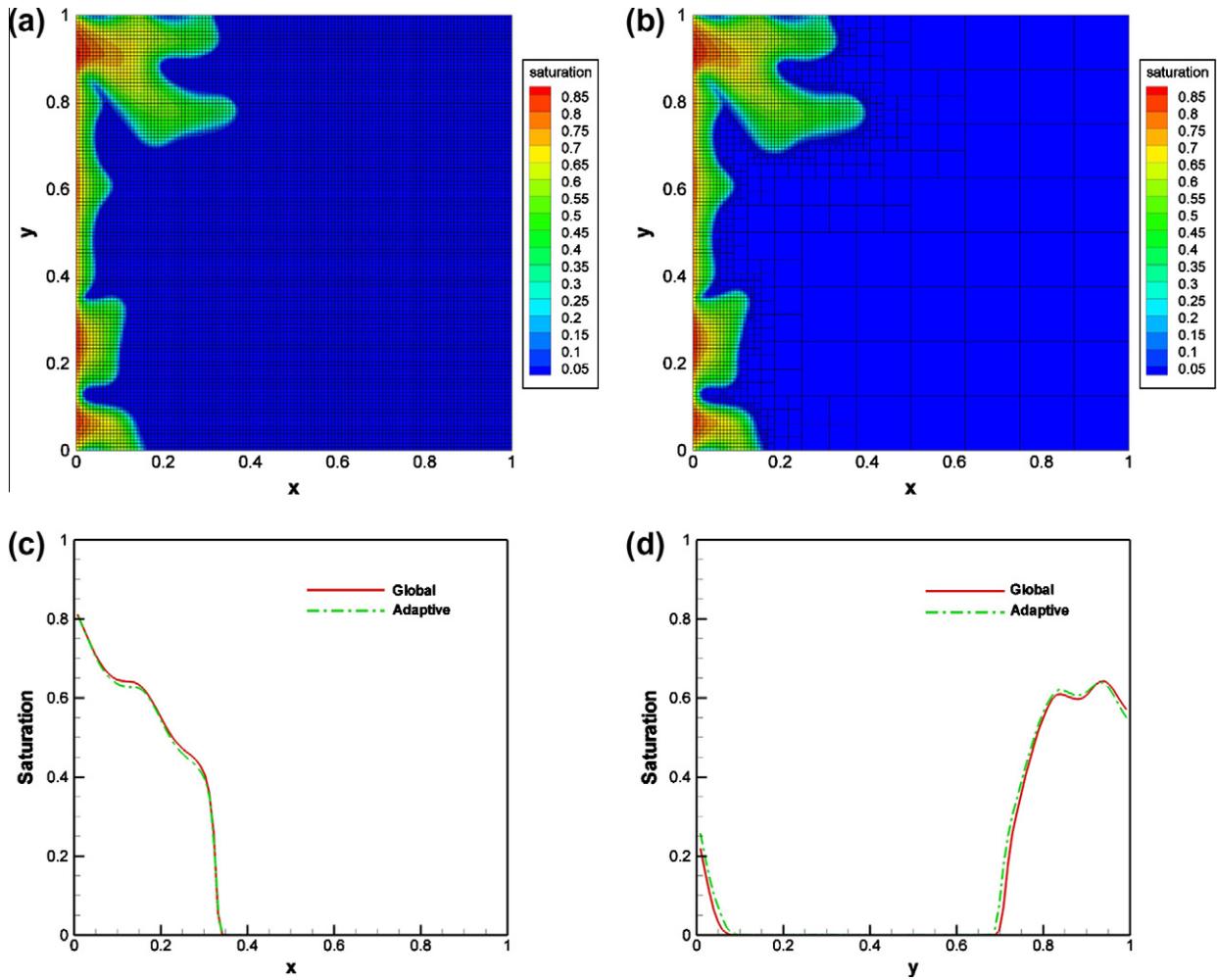


Fig. 10. Comparison of computational mesh distribution and saturation fields at $t = 0.488$ s using the continuous discretization. (a) Uniform mesh with 16,384 elements and 165,380 degrees of freedom. (b) Adaptive mesh with 3220 elements and 12,880 degrees of freedom. (c) Saturation profile along $y = 0.95$. (d) Saturation profile along $x = 0.15$.

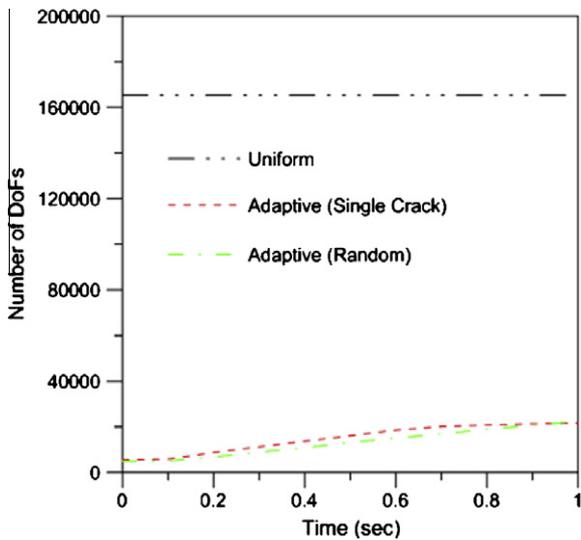


Fig. 11. Comparison of the number of degrees of freedom in the single crack and random media cases with the corresponding number for uniform grids. The adaptive meshes need between 86% and 96% fewer degrees of freedom.

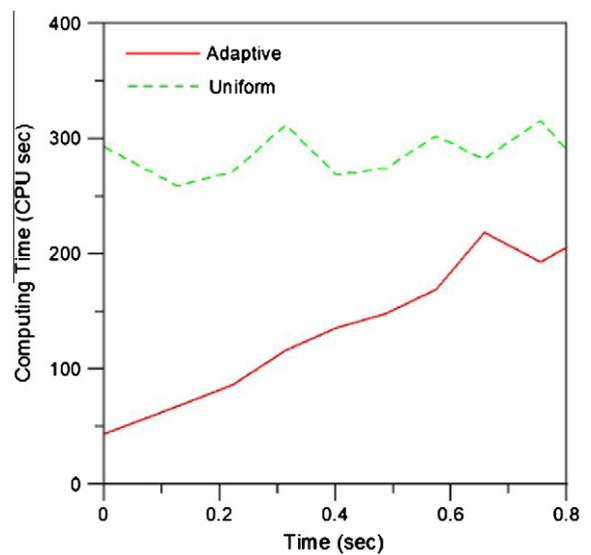


Fig. 12. CPU time per IMPES step for the adaptive and uniform grid simulations.

To illustrate the effect of selecting different threshold levels, results obtained for two simulations under otherwise identical conditions are shown in Fig. 6. In the first simulation, cells are

coarsened if $\eta_K < \theta_c = 1.1$ and refined if $\eta_K > \theta_r = 1.7$. In the second, these thresholds are chosen as $\theta_c = 0.21$ and $\theta_r = 0.28$. As can be seen, the saturation fields are essentially identical, but the less stringent, first set of threshold values result in more coarsening,

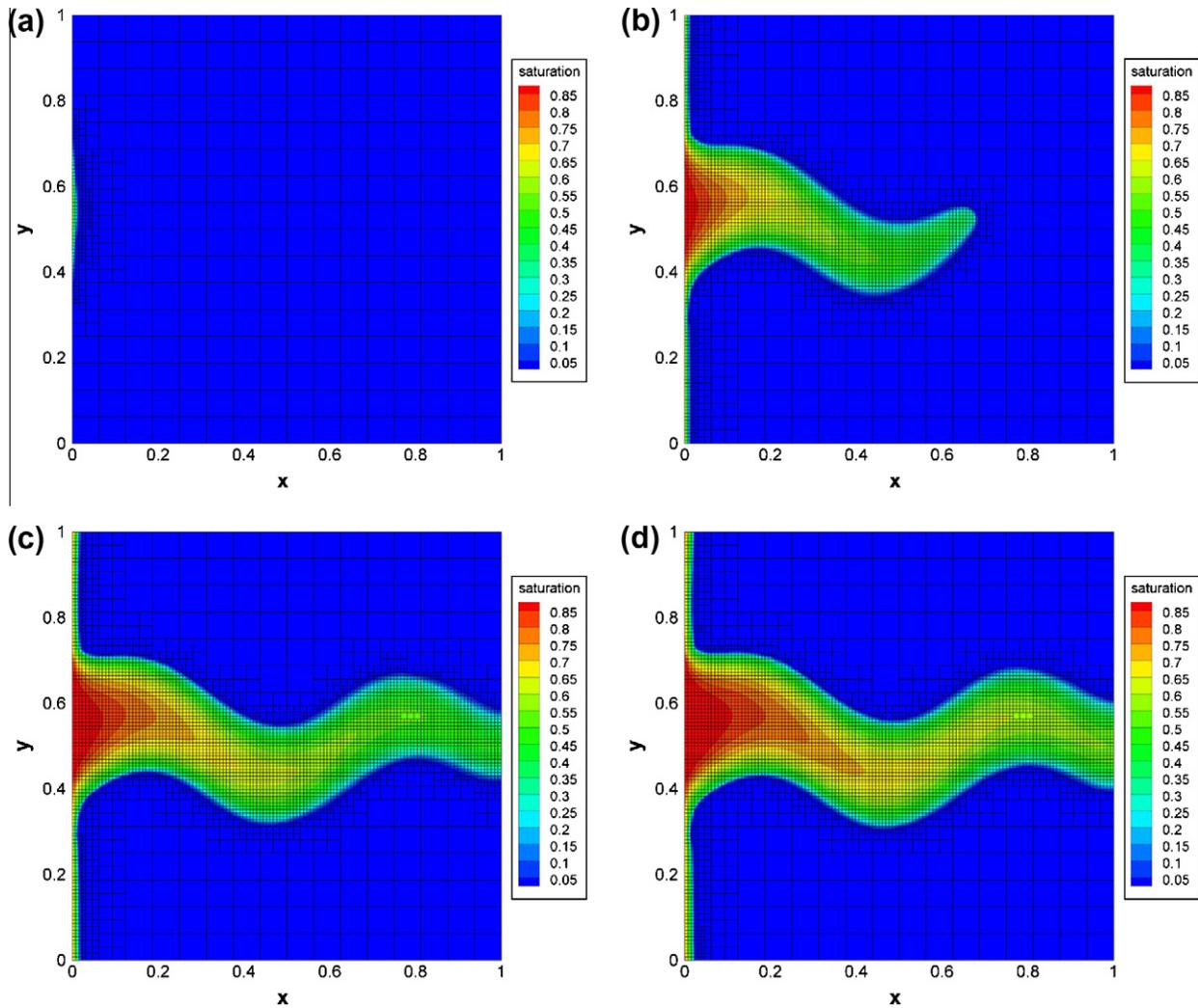


Fig. 13. Saturation fields for the single crack medium. (a) $t = 0.002$ s with 520 elements and 2080 degrees of freedom. (b) $t = 0.427$ s with 3424 elements and 13,696 degrees of freedom. (c) $t = 0.731$ s with 5185 elements and 20,740 degrees of freedom. (d) $t = 0.979$ s with 5551 elements and 22,204 degrees of freedom.

particularly along the center of the crack as well as along the inlet boundary. This yields a computational mesh with about 12% less cells. To ensure high accuracy, all following computations were done with the second set of thresholds, however.

5. Results and discussion

The simulations presented and analyzed in this section were performed using the spatial permeability distributions given in Section 2.1, and with the input parameters listed in Table 1. It should be noted that in the validation section, the porosity is set to 0.6 rather than the value of 1.0 used in all other cases. Our implementation uses the Open Source finite element library deal.II [42] written in C++.

5.1. Numerical validation

Here, for numerical validation, we compare the present results obtained with the adaptive grid method to previous work using uniform grids in which Raviart–Thomas finite elements are adopted for velocity and discontinuous Galerkin (DG) elements for pressure and saturation [40].

Fig. 7a and b compares saturation contours in the single crack case between these two discretizations at $t = 0.154$ s in the single

crack case. Fig. 7c and d shows saturation profiles along $y = 0.5$ and $x = 0.1$, respectively. The results are essentially identical.

In the case of a random medium, there are a couple of small but more noticeable differences as shown in the results at $t = 0.249$ s presented in Fig. 8. The extent of the penetration of the invading fluid is predicted to be higher in the DG simulations (see top and bottom fingers in Fig. 8a), and there are differences of up to 10% in the saturation as shown in both the longitudinal and transverse profiles in Fig. 8c and d. These differences are due to the lower accuracy and higher diffusivity of the low-order DG approximation. In contrast, the discretization used here shows a more pronounced and less diffused appearance, a consequence of the less heavy-handed stabilization even though we use far fewer degrees of freedom compared to the uniformly refined grid used for the DG method.

5.2. Comparison of global vs. adaptive refinement

Figs. 9 and 10 show the saturation distribution in the random medium obtained using global and locally adaptive refinement at $t = 0.278$ s and $t = 0.488$ s respectively. Well resolved saturation distributions that are quasi-identical can be obtained using both refinement techniques. The effectiveness of the adaptive grid method is highlighted by the fact that, at time $t = 0.278$ s for instance (Fig. 9), only 7924 degrees of freedom (DoFs) are required

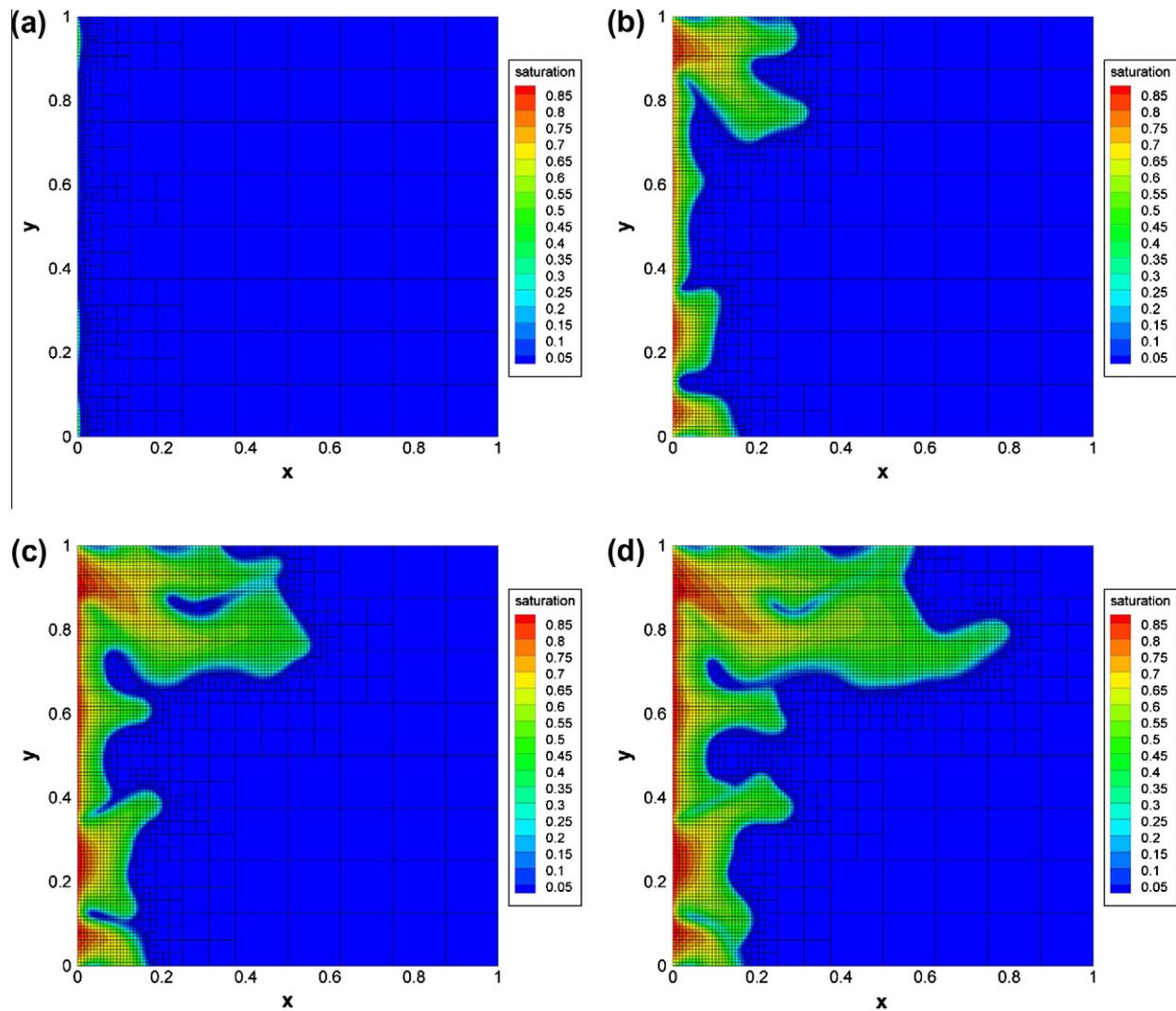


Fig. 14. Saturation fields for the random medium. (a) $t = 0.003$ s with 544 elements and 2176 degrees of freedom. (b) $t = 0.411$ s with 2722 elements and 10,888 degrees of freedom. (c) $t = 0.755$ s with 4744 elements and 18,976 degrees of freedom. (d) $t = 1.085$ s with 6361 elements and 25,444 degrees of freedom.

with local adaptation compared to 165,380 DoFs for the globally refined uniform grid. The total number of degrees of freedom for the local adaptive method increases with time: as the wetting phase (water) gradually invades the areas initially occupied by the non-wetting phase (oil) a correspondingly larger part of the computational domain is refined. As the gradients abate in the water invaded portion of the domain, the mesh is coarsened again. Overall, the required number of DoFs was reduced by 86 to 96% ($t = 0$ and $t = 1$) for both single crack and random media as shown in Fig. 11. Fig. 12 compares the computing (CPU) time per IMPES step as a function of the process simulation time; significant computational performance gains are achieved with the locally adaptive method, resulting in an overall reduction of the aggregate CPU time for the entire simulation of 42% without any reduction in accuracy in the solution. Experience shows that the savings in both number of unknowns as well as CPU time would be even larger for three dimensional simulations.

5.3. Saturation distribution in a domain with a single crack

Having established the accuracy of the adaptive mesh two-phase flow method, further analysis of the single crack simulations with the permeability given by Eq. (12) are presented here. Such

simulations can be considered an idealized representation of situations arising in fuel cell fibrous media when carbon fibres break due to over-compression [47–49]; in geo-science, a porous medium with a single crack distribution can be found in sandstone reservoir rocks [5,2].

The saturation distribution as a function of time is shown in Fig. 13. The mobile fluid slowly meanders its way along the central areas where permeability is highest. In this case, due to the higher saturation gradients in the central area, locally refined cells are concentrated in the middle of the domain and propagate in time with the saturation front.

5.4. Saturation distribution in a porous medium with random permeability

Simulations presented in this section were performed for a medium with a permeability distribution given by Eq. (13), see Fig. 2. The saturation distribution is shown in Fig. 14. Fig. 14b shows that the more mobile displacing fluid seeks pathways formed by inter-connected high-permeability zones, faults and fractures. The fluid appears to form fingers, a phenomenon frequently found in both oil reservoirs and also reported in porous diffusion media of fuel cells [50].

6. Conclusions and future work

A numerical method allowing high resolution simulations of two-phase flow in porous media was presented. The method is based on a finite element discretization using continuous elements. It incorporates a new entropy based stabilizing term that ensures accuracy while providing stability, and locally adaptive refinement allowing highly resolved time dependent simulations. The effectiveness of the numerical method was demonstrated by performing simulations corresponding to two-phase flow in two types of porous media (single crack and heterogeneous).

The implementation used an implicit pressure and explicit saturation (IMPES) formulation. Compared to a simple finite volume (or DG scheme), the higher order of approximation used for the saturation variable, in conjunction with the stabilizing dissipative term, makes the standard CFL restriction (i.e. $\Delta t \leq h/|\mathbf{u}|$) no longer suitable for evaluating the adaptive time step for the modified Eq. (26). In order to further improve computational efficiency, future work will focus on the development and implementation of more efficient time stepping schemes, and a particularly attractive avenue is the use of an adaptive time stepping strategy that takes advantage of the slower variation of pressure compared to saturation [29,30], and allows larger time steps for the computationally costly pressure equation. Such an algorithm would make it more practical to perform 3D simulations including enhanced physical modelling, such as capillary diffusion, that would enhance the applicability of the method to a broader range of multi-phase transport problems of practical interest.

Acknowledgments

The work of the first, second and fourth authors was funded through the Canada Research Chairs Program and the MITACS Network of Centres of Excellence. Parts of the work by the third author (WB) were funded through Award No. KUS-C1-016-04, made by the King Abdullah University of Science and Technology, and through an Alfred P. Sloan Research Fellowship.

Thanks are due to Dr. Zhangxin Chen (University of Calgary) for his invaluable insights on finite element theory and to Mrs. Hsiao-Yun Janette Cheng (PhD candidate, University of Sheffield), for helping with the graphics.

References

- [1] Bear J. Dynamics of fluids in porous media. American Elsevier; 1972.
- [2] Nield DA, Bejan A. Convection in porous media. New York: Springer; 2006.
- [3] Peaceman DW. Fundamentals of numerical reservoir simulation. Amsterdam: Elsevier; 1977.
- [4] Aziz K, Settari A. Petroleum reservoir simulation. London: Applied Science Publishers Ltd.; 1979.
- [5] Gerritsen MG, Durlafsky LJ. Modeling fluid flow in oil reservoirs. *Annu Rev Fluid Mech* 2005;37(1):211–38.
- [6] Branets LV, Ghai SS, Lyons SL, Wu XH. Challenges and technologies in reservoir modeling. *Commun Computat Phys* 2009;6(1):1–23.
- [7] Lister S, Djilali N. Two-phase transport in porous gas diffusion electrodes. In: Faghri M, Sundén B, editors. Southampton (UK): WIT Press; 2005.
- [8] Djilali N. Computational modelling of polymer electrolyte membrane (PEM) fuel cells: challenges and opportunities. *Energy* 2007;32(4):269–80.
- [9] Djilali N, Sui PC. Transport phenomena in fuel cells: from microscale to macroscale. *Int J Computat Fluid Dynam* 2008;22(1–2):115–33.
- [10] Kumbur EC, Sharp KV, Mench MM. Validated Leverett approach for multiphase flow in PEFC diffusion media. *J Electrochem Soc* 2007;154(12):B1295–304.
- [11] Kumbur EC, Sharp KV, Mench MM. Validated Leverett approach for multiphase flow in PEFC diffusion media. *J Electrochem Soc* 2007;154(12):B1305–14.
- [12] Kumbur EC, Sharp KV, Mench MM. Validated Leverett approach for multiphase flow in PEFC diffusion media. *J Electrochem Soc* 2007;154(12):B1315–24.
- [13] Niessner J, Helmig R. Multi-scale modeling of three-phase-three-component processes in heterogeneous porous media. *Adv Water Resour* 2007;30(11):2309–25.
- [14] Niessner J, Helmig R, Jakobs H, Roberts JE. Interface condition and linearization schemes in the newton iterations for two-phase flow in heterogeneous porous media. *Adv Water Resour* 2005;28(7):671–87.
- [15] Kees CE, Miller CT. Higher order time integration methods for two-phase flow. *Adv Water Resour* 2002;25(2):159–77.
- [16] Binning P, Celia MA. Practical implementation of the fractional flow approach to multi-phase flow simulation. *Adv Water Resour* 1999;22(5):461–78.
- [17] Tseng PH, Zyvoloski GA. A reduced degree of freedom method for simulating non-isothermal multi-phase flow in a porous medium. *Adv Water Resour* 2000;23(7):731–45.
- [18] Piri M, Blunt MJ. Three-dimensional mixed-wet random pore-scale network modeling of two- and three-phase flow in porous media. I. Model description. *Phys Rev E* 2005;71(2):026301.
- [19] Christov I, Popov B. New non-oscillatory central schemes on unstructured triangulations for hyperbolic systems of conservation laws. *J Computat Phys* 2008;227(11):5736–57.
- [20] Hou T, Wu XH. A multiscale finite element method for elliptic problems in composite materials and porous media. *J Computat Phys* 1997;134(1):169–89.
- [21] Durlafsky LJ, Efendiev Y, Ginting V. An adaptive local-global multiscale finite volume element method for two-phase flow simulations. *Adv Water Resour* 2007;30(3):576–88.
- [22] Efendiev Y, Ginting V, Hou T, Ewing R. Accurate multiscale finite element methods for two-phase flow simulations. *J Computat Phys* 2006;220(1):155–74.
- [23] Efendiev Y, Hou T. Multiscale finite element methods for porous media flows and their applications. *Appl Numer Math* 2007;57(5–7):577–96.
- [24] Jenny P, Lee SH, Tchelepi HA. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *J Computat Phys* 2003;187(1):47–67.
- [25] Helmig R, Niessner J, Class H. Recent advances in finite element methods for multi-phase flow processes in porous media. *Int J Computat Fluid Dynam* 2006;20(3/4):245–52.
- [26] Sheldon JW, Zondek B, Cardwell WT. One-dimensional, incompressible, non-capillary, two-phase fluid flow in a porous medium. *Trans SPE AIME* 1959;216:290–6.
- [27] Stone HL, Garder Jr. Analysis of gas-cap or dissolved-gas reservoirs. *Trans SPE AIME* 1961;222:92–104.
- [28] Douglas Jr J, Peaceman DW, Rachford Jr HH. A method for calculating multi-dimensional immiscible displacement. *Trans SPE AIME* 1961;216:297–306.
- [29] Chen Z. Computational methods for multi-phase flows in porous media. SIAM 2006.
- [30] Chen Z, Huan G, Li B. An improved IMPES method for two-phase flow in porous media. *Transport Porous Med* 2004;54:361–76.
- [31] Ainsworth M, Oden JT. A posteriori error estimation in finite element analysis. John Wiley and Sons; 2000.
- [32] Verfürth R. A review of a posteriori error estimation and adaptive mesh refinement techniques. New York, Stuttgart: Wiley/Teubner; 1996.
- [33] Carey GF. Computational grids: generation, adaptation and solution strategies. Taylor & Francis; 1997.
- [34] Kröner D. Numerical schemes for conservation laws. Wiley/Teubner; 1997.
- [35] Noelle S, Rosenbaum W, Rumpf M. 3D adaptive central schemes: part I. Algorithms for assembling the dual mesh. *Appl Numer Math* 2006;56(6):778–99.
- [36] Karni S, Kurganov A, Petrova G. A smoothness indicator for adaptive algorithms for hyperbolic systems. *J Computat Phys* 2002;178(2):323–41.
- [37] Tang H, Tang T. Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM J Numer Anal* 2003;41(2):487.
- [38] Chen Z. Finite element methods and their applications. Springer; 2005.
- [39] Guermont JL, Pasquetti R. Entropy-based nonlinear viscosity for fourier approximations of conservation laws. *CR Math* 2008;346(13–14):801–6.
- [40] Li Y, Bangerth W. The deal.II tutorial: step-21. <http://dealii.org/developer/doxygen/deal.II/step_21.html>.
- [41] Brezzi F, Fortin M. Mixed and hybrid finite element methods. Springer-Verlag; 1991.
- [42] Bangerth W, Hartmann R, Kanschat G. deal.II – a general purpose object oriented finite element library. *ACM Trans Math Softw* 2007;33(4):24/1–24/27.
- [43] Kronbichler M, Bangerth W. The deal.II tutorial: step-31. <http://dealii.org/developer/doxygen/deal.II/step_31.html>.
- [44] Neckels D, Bangerth W. The deal.II tutorial, step-33. <http://dealii.org/developer/doxygen/deal.II/step_33.html>.
- [45] Zhang F. The Schur complement and its applications. Springer; 2005.
- [46] Bangerth W. The deal.II tutorial: step-20. <http://dealii.org/developer/doxygen/deal.II/step_20.html>.
- [47] Nitta I, Hottinen T, Himanen O, Mikkola M. Inhomogeneous compression of PEMFC gas diffusion layer: part I. Experimental. *J Power Source* 2007;171(1):26–36.
- [48] Hottinen T, Himanen O, Karvonen S, Nitta I. Inhomogeneous compression of PEMFC gas diffusion layer: part II. Modeling the effect. *J Power Source* 2007;171(1):113–21.
- [49] Bazylak A, Sinton D, Liu ZS, Djilali N. Effect of compression on liquid water transport and microstructure of PEMFC gas diffusion layers. *J Power Source* 2007;163(2):784–92.
- [50] Lister S, Sinton D, Djilali N. Ex situ visualization of liquid water transport in PEM fuel cell gas diffusion layers. *J Power Source* 2006;154(1):95–105.

Appendix B

An H-adaptive Operator Splitting Method for Two-phase Flow in 3D Heterogeneous Porous Media

An H-adaptive Operator Splitting Method for Two-phase Flow in 3D Heterogeneous Porous Media

C.C. Chueh^a, W. Bangerth^{b,*}, N. Djilali^{a,*}

^a*Institute for Integrated Energy Systems and Department of Mechanical Engineering,
University of Victoria, PO Box 3055 STN CSC, Victoria, BC, Canada, V8W 3P6*

^b*Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA*

Abstract

The simulation of multiphase flow in porous media is a ubiquitous problem, but it is computationally costly. This paper presents an interconnected set of algorithms which we show can accelerate computations by more than two orders of magnitude compared to traditional techniques. Our method uses: an adaptive operator splitting technique based on an a posteriori criterion to separate the flow from the transport equations; adaptive meshing to reduce the size of the discretized problem; efficient block preconditioned solver techniques for fast solution of the discrete equations; and a recently developed artificial diffusion strategy to stabilize the numerical solution of the transport equation. We demonstrate the accuracy and efficiency of our approach using numerical experiments in 2D and 3D.

Key words: Adaptive mesh refinement; Stabilized finite element method; Operator splitting; Preconditioning; Two-phase flow; Heterogeneous porous media; Fuel Cells

* Corresponding authors.

Email addresses: bangerth@math.tamu.edu (W. Bangerth), ndjilali@uvic.ca (N. Djilali).

1 Introduction

Multiphase flow models in porous media are used in a wide variety of fields such as oil reservoirs [1–3], the flow of magma in the earth crust [4] or transport processes in fuel cells [5,6]. In the latter example, current interest is primarily focused on polymer electrolyte membrane (PEM) fuel cells in which the gas diffusion layer (GDL) provides pathways for gaseous fuel to reach the catalyst sites, for electron produced electrochemically at the catalyst sites [7] to be conducted to the current collector, and, very importantly, for excess liquid water to exit the system. Excess water can curtail transport of reactant gases [8] as well as exacerbate degradation [9]. Improvements in PEM fuel cells are therefore contingent upon advancements in water management, which can be facilitated by the simulation of realistic, complex two-phase transport phenomena in 3D heterogeneous porous media.

For fuel cells as well as other applications involving porous media, the bottleneck for simulations is the speed with which complex three-dimensional problems can be solved. In this regard, classical finite element (FE) and finite volume (FV) methods – while having the potential to accurately represent all the salient physics – do not always compare favorably against other methods such as pore network modelling [10–14] or Lattice Boltzmann Method (LBM) [15,16]. Increasing the computational efficiency of the FE/FV numerical methods while retaining their accuracy remains a central challenge.

There are at least five areas which a high performance simulator needs to address to improve computational speed for a given level of accuracy:

- Higher order spatial discretizations can yield the same accuracy at smaller computational cost, but need to incorporate stabilization mechanisms for hyperbolic problems such as those representing multiphase flow;
- Adaptive mesh refinement that can vastly reduce the number of cells required to resolve

the flow field;

- Adaptive time stepping methods that allow the use of the large time step limited solely by the physical time scale rather than numerical stability;
- Operator splitting methods for coupled problems to transform a complex problem into a sequence of simpler problems for which more efficient solver techniques are available;
- Efficient solver and preconditioning methods that can accelerate the solution of the linear problems.

In this paper, we discuss a framework that addresses all of these issues and demonstrate that, with judicious algorithm choice, we can accelerate the numerical simulation of multi-phase flow problems by more than two orders of magnitude, making possible the simulation previously inaccessible scenarios. The implementation of these methods is described in detail and the associated code is made available as part of the widely used Open Source finite element library `deal.II` [17,18].

To place the proposed techniques into their proper context, in the remainder of this introduction, we summarize the main contributions of this paper and provide an overview of the related literature. The mathematical formulation and algorithm description are discussed in the following Sections.

Higher order discretizations and stabilization issues. Stability issues arising from the discretization of advection-dominated problems have long led to the belief that finite volume or upwinding schemes [19] (or their modern incarnation in the form of discontinuous Galerkin (DG) finite element methods [20]) combined with the usual set of flux/slope limiters are the only choice; in other words, upwinding schemes should be preferred to the introduction of artificial diffusion (see, e.g., Kröner [21]). The primary reason for this belief is that diffusive stabilization often excessively smoothes out sharp fronts (e.g. Helmig

[22] and Paul [23]).

On the other hand, it is difficult to derive upwind schemes of higher order, and finite volume methods lack a systematic framework to derive high order discretizations that finite element theory provides. Higher order methods are attractive because – up to a certain point – they typically provide the same accuracy with fewer degrees of freedom and less numerical work. Consequently, the development of stabilization methods for higher order methods applied to transport dominated problems is an important step to achieving higher computational efficiency. We have previously validated and demonstrated an artificial diffusion method that, when used with a general continuous finite element discretization for the saturation transport (advection) equation in two-phase flow in porous media, ensures stability and accuracy of the solution [24]. The method uses an entropy-based diffusion term, proposed by Guermond and Pasquetti [25], and is able to efficiently damp unphysical oscillations while providing the same amount of resolution of saturation field as a low-order DG method [26] without upwinding. The trick is to ensure that the artificial viscosity term acts only in the vicinity of strong gradients in the saturation and other discontinuities, while disabling it in regions where the solution is smooth. The scheme offers higher order accuracy at least in smooth regions while providing stability where necessary [25].

Adaptive mesh refinement. In many engineering problems involving fluid dynamics, structural mechanics etc, automated adaptive mesh refinement (AMR) has been used to obtain numerical solutions with higher accuracy and resolution, while requiring less memory and shorter computational times. Adaptivity, which can be traced back to the late 1970s [27], is based on the idea that, in order to improve numerical accuracy, a fine mesh is not necessarily required everywhere but only in areas where the solution is eventful. A variety of AMR methods have been proposed depending on the type of

physical problem and associated partial differential equations (PDE), and a large body of literature exists for these methods (see, for example, [28–31] for a general overview, and [32,33] for overviews in multiphase flows). Adaptive methods can be grouped into a number of categories: h -adaptive methods change the local element size h in response to smoothness properties of the solution; p -adaptive methods vary the local polynomial order p of the finite element space; hp -adaptive methods combine the previous two methods; r -adaptive methods move mesh points (see, e.g., Tang and Tang [34], Tan *et al.* [35,36], Di *et al.* [37–39], Li and Tang [40] and Zhang *et al.* [41]); and subgrid methods superimpose finer meshes over the original grid where necessary [42]. Despite this variety, h -adaptive methods are the most commonly used in engineering applications today. We note that the literature on the use of adaptive meshes remains relatively limited for *transient* multiphase flow in porous media.

Adaptivity is driven by refinement indicators. These may be based on error estimates for the underlying partial differential equation [28,29,31]; however, for many applications, simpler error indicators based on the smoothness of the solution, such as those proposed in [21], are often sufficient to refine and coarsen the mesh at each time step. For hyperbolic problems, most choice of refinement criterion will yield fine grids and consequently high accuracy of physical features such as sharp fronts as they evolve in space and time, while simultaneously reducing computational cost by keeping the mesh coarse in areas where fine meshes are not required.

For the purpose of this work, we use a relatively simple refinement criterion that is based on the gradient of the saturation. This obviously ensures the accurate resolution of sharp fronts in the saturation profile but also ensures good resolution of the other variables due to the strong couplings.

Operator splitting techniques. Multi-phase porous media flow is a coupled problem in which the flow and pressure fields affect the transport of individual phases, but the phase composition also nonlinearly affects the flow and pressure fields. Such couplings complicate significantly the numerical solution. Operator splitting methods alleviate the problem by breaking up the coupled problem into a sequence of simpler problems whose solution can, if necessary, be iterated until sufficient accuracy is reached. Operator splitting techniques have a long history. Since alternating-direction methods were introduced by Douglas, Peaceman and Rachford [43–47] and fractional step methods by D’jakonov and Yanenko [48,49], these procedures have been applied in the numerical simulation of many important problems.

In the context of non-reactive systems of multi-phase flow in porous media, two kinds of operator splitting are typically used: splitting the pressure-velocity calculation from the saturation calculation, and splitting the saturation calculation into convection and diffusion components. The latter strategy has been developed and applied in Espedal and Ewing [50], Dahle [51], Espedal and Karlsen [52] and Karlsen *et al.* [53] and is motivated by the fact that it is easy to combine modern methods developed within the hyperbolic community (convection) for tracking discontinuous solutions, with efficient methods (e.g. multi-grid) for the implicit discretization of the parabolic (diffusive) part.

Splitting the velocity and pressure equations from the equation for the saturation also has a long history. The time stepping schemes most commonly used are of IMPES (implicit pressure, explicit saturation) type and date back to Sheldon *et al.* [54] and Stone and Gardner [55]). In these schemes, one first solves the implicit, time independent pressure/velocity system with the current saturation values, and then uses an explicit time stepping scheme to advance the saturation by one time step. While this effectively decouples the two equations, one is left with the important problem that solving the equations for flow velocity and pressure is much more expensive than solving that for the saturation.

On the other hand, the effect of the saturation on the flow field is typically weak, i.e. the flow field only evolves slowly while the saturation changes significantly at each time step. Consequently, we will here explore the possibility of not solving the computationally expensive flow equations at every time step but only when necessary, thereby significantly reducing computing time.

Such methods have previously been developed (see, for example, Chen *et al.* [56,57] and Abreu *et al.* [58]) but with a fixed number of saturation time steps between each solution of the flow field. We will here make the timing of solving the flow equations adaptive based on an a posteriori criterion. Finally, we note that for reactive transport, an additional splitting can be added to incorporate general reaction terms as a system of coupled ordinary differential equations [59–63] which also often have their own time scales independent of the flow or saturation fields.

Solver and preconditioner techniques The result of operator splitting methods is a sequence of linear systems – frequently ill-conditioned, non-symmetric and/or indefinite – that need to be solved efficiently. Their size typically leaves Krylov subspace methods as the only viable choice, hence requiring good preconditioners for efficient solution. In the context of the mixed formulation of porous media flow, the saddle point structure of the flow problem presents a particular complication. The literature discussing solutions for this problem is vast; to name only a few, Rusten and Winther [64] developed a preconditioned iterative method for saddle point problems; Arnold, Falk and Winther [65] recommended a block diagonal preconditioner; and more recently, Powell and Silvester [66,67] conceived a parameter-free preconditioning method. We will here adapt a method originally proposed for the Stokes system by Silvester and Wathen [68] that leads to very efficient and adjustable preconditioners.

Overview of the paper. The remainder of the paper presents the mathematical formulation of the methods outlined above and shows how they can be combined to form a method that achieves a speed-up of more than two orders of magnitude compared to a traditional two-phase porous medium solver. Section 2 presents the mathematical flow model, followed in Section 3 by a discussion of the time stepping, discretization, operator splitting and stabilization methods; Section 4 introduces a numerical test case; Section 5 presents numerical results related to the accuracy and efficiency of our methods, and discusses the implications. We conclude in Section 6.

The code that implements our methods is available as part of the `deal.II` library [17], together with detailed documentaion and a tutorial to allow further developments by others.

2 Mathematical model of two-phase porous media flow

We consider the flow of a two-phase immiscible, incompressible fluid. Capillary and gravity effects are neglected, and that viscous effects are assumed dominant. The governing equations for such a flow are then [3]

$$\mathbf{u}_t = -\mathbf{K}\lambda_t(S)\nabla p, \quad (1)$$

$$\nabla \cdot \mathbf{u}_t = q, \quad (2)$$

$$\epsilon \frac{\partial S}{\partial t} + \nabla \cdot (\mathbf{u}_t F(S)) = 0, \quad (3)$$

where S is the saturation (volume fraction) of the second (wetting) phase, p is the pressure, \mathbf{K} is the permeability tensor, λ_t is the total mobility, ϵ is the porosity, F is the fractional flow of the wetting phase, q is the source term and \mathbf{u}_t is the total velocity. The total

mobility, fractional flow of the wetting phase and total velocity are respectively given by

$$\lambda_t(S) = \lambda_w + \lambda_{nw} = \frac{k_{rw}(S)}{\mu_w} + \frac{k_{rnw}(S)}{\mu_{nw}}, \quad (4)$$

$$F(S) = \frac{\lambda_w}{\lambda_t} = \frac{\lambda_w}{\lambda_w + \lambda_{nw}} = \frac{k_{rw}(S)/\mu_w}{k_{rw}(S)/\mu_w + k_{rnw}(S)/\mu_{nw}}, \quad (5)$$

$$\mathbf{u}_t = \mathbf{u}_w + \mathbf{u}_{nw} = -\lambda_t(S)\mathbf{K} \cdot \nabla p, \quad (6)$$

where subscripts w, nw represent the wetting and non-wetting phases, respectively.

For convenience, the porosity ϵ in equation (3), which can be considered a scaling factor, is set to one. Following a commonly used prescription for the dependence of the relative permeabilities k_{rw} and k_{rnw} on saturation, we use

$$k_{rw} = S^2, \quad k_{rnw} = (1 - S)^2. \quad (7)$$

Equations (1)–(3) are augmented by initial conditions for the saturation and boundary conditions for the pressure. Since saturation and the gradient of the pressure uniquely determine the velocity, no boundary conditions are necessary for the velocity. Since the flow equations do not contain time derivatives, initial conditions for the velocity and pressure variables are not required. The flow field separates the boundary into inflow or outflow parts. Specifically,

$$\mathbf{\Gamma}_{in}(t) = \{\mathbf{x} \in \partial\Omega : \mathbf{n} \cdot \mathbf{u}_t < 0\}, \quad (8)$$

and we arrive at a complete model by also imposing boundary values for the saturation variable on the inflow boundary $\mathbf{\Gamma}_{in}$.

3 Numerical methods

As noted in the Introduction, the numerical solution of equations (1)–(3) is computationally costly. Efficient state-of-the-art algorithms are described in detail in the following subsections.

3.1 Adaptive operator splitting and time stepping

The time stepping schemes most commonly used to solve equations (1)–(3) of the kind are of IMPES (*implicit pressure, explicit saturation*) type in which one first solves the implicit pressure/velocity system (1)–(2) with the current saturation values, and then uses an explicit time stepping scheme to advance the saturation by one time step using (3). The vast majority of computing time in such schemes is spent in the implicit solver for the pressure and velocity variables. Computing efforts may be significantly reduced by noting that the pressure and velocity fields depend only weakly on the saturation, and therefore do not change significantly between time steps, whereas saturation fronts typically move by one cell in each step. Intuitively, a net reduction in computing time should be achievable by rebalancing the computing efforts between the saturation and the pressure/velocity system. An operator splitting approach to achieve this consists of solving for the saturation at every time step, and only updating the velocity and pressure whenever necessary (we will call the intervals between such updates “macro timesteps”). Such a method is described by Abreu *et al.* in [58] where the pressure system is only solved once per fixed number of saturation time steps. A better approach would let the length of the macro timesteps depend *adaptively* on the changes incurred in the saturation since the last update.

To derive such a scheme, let superscripts in parentheses denote the number of the time

step at which a quantity is defined. Furthermore, let $n_p < n$ be the last step at which we actually computed the velocity and pressure variables, and $n_{pp} < n_p$ be the next-to-last step at which this happened. For the coupling in the saturation equation, we only need the velocity $\mathbf{u}_t^{(n)}$; however, we don't want to compute $\mathbf{u}_t^{(n)}$ at each time step but rather use a quantity already computed. Consequently, we need a criterion that tells us when this approximation becomes inadequate, i.e. we need to estimate $\|\mathbf{u}_t^{(n)} - \mathbf{u}_t^{(n_p)}\|$ without actually computing $\mathbf{u}_t^{(n)}$.

We will not attempt to derive such an estimate at the level of the original partial differential equation. Rather, let us assume (as discussed further down below) that we have discretized the velocity-pressure equations with one of the usual finite element methods. Then, if we were to solve for the discretized pressure and velocity in time step n , they would have to satisfy the system of equations

$$\begin{pmatrix} \mathbf{M}^u(S^{(n-1)}) & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n)} \\ \mathbf{P}^{(n)} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{F}_2^{(n)} \end{pmatrix}, \quad (9)$$

where \mathbf{B}, \mathbf{B}^T correspond to the operators $-\text{div}$ and ∇ , respectively, and $\mathbf{M}^u(S)$ corresponds to $\mathbf{K}^{-1}\lambda_t(S)^{-1}$. Obviously, $\mathbf{U}^{(n_p)}, \mathbf{P}^{(n_p)}$ have to satisfy a corresponding set of equations. Then we have the following lemma:

Lemma 1 *Let $\mathbf{U}^{(n_p)}, \mathbf{P}^{(n_p)}$ and $\mathbf{U}^{(n)}, \mathbf{P}^{(n)}$ be solutions to (9) at time steps n_p and n , respectively, and assume that the source terms are constant in time, i.e. $\mathbf{F}_2^{(n_p)} = \mathbf{F}_2^{(n)} = \mathbf{F}_2$. Then there exists a constant C so that*

$$\|\mathbf{U}^{(n)} - \mathbf{U}^{(n_p)}\| \leq C \|\mathbf{M}^u(S^{(n-1)}) - \mathbf{M}^u(S^{(n_p-1)})\| \quad (10)$$

for any vector norm on the left hand side and associated matrix norm on the right.

Proof Let

$$A(S) = \begin{pmatrix} \mathbf{M}^u(S) \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix}, \quad \mathbf{V} = \begin{pmatrix} \mathbf{U} \\ \mathbf{P} \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} 0 \\ \mathbf{F}_2 \end{pmatrix}.$$

Then $\mathbf{V}^{(n)} = A(S^{(n-1)})^{-1} \mathbf{G}$ and $\mathbf{V}^{(n_p)} = A(S^{(n_p-1)})^{-1} \mathbf{G}$. Consequently,

$$\begin{aligned} \|\mathbf{U}^{(n)} - \mathbf{U}^{(n_p)}\| &\leq \|\mathbf{V}^{(n)} - \mathbf{V}^{(n_p)}\| \\ &= \|[A(S^{(n-1)})^{-1} - A(S^{(n_p-1)})^{-1}] \mathbf{G}\| \\ &\leq \|A(S^{(n-1)})^{-1} - A(S^{(n_p-1)})^{-1}\| \|\mathbf{G}\| \\ &= \|A(S^{(n-1)})^{-1} [A(S^{(n_p-1)}) - A(S^{(n-1)})] A(S^{(n_p-1)})^{-1}\| \|\mathbf{G}\| \\ &\leq \|A(S^{(n-1)})^{-1}\| \|A(S^{(n_p-1)})^{-1}\| \|A(S^{(n_p-1)}) - A(S^{(n-1)})\| \|\mathbf{G}\|. \end{aligned}$$

The claim follows by observing that among the components of A only M^u depends on the saturation, i.e. $\|A(S^{(n_p-1)}) - A(S^{(n-1)})\| = \|M^u(S^{(n_p-1)}) - M^u(S^{(n-1)})\|$, and setting $C = \sup_S \|A(S)^{-1}\|^2 \|\mathbf{F}_2\|$, assuming uniform boundedness of $A(S)$ as a function of saturation S .

Since the preceding lemma holds for any vector and associated matrix norm, we may choose one that allows for the convenient evaluation of the above expression. This leads to the following statement:

Theorem 2 *Under the same assumptions as in the previous lemma, we have*

$$\|\mathbf{U}^{(n)} - \mathbf{U}^{(n_p)}\|_1 \leq C_1 \max_{\kappa \in \mathbb{T}} |\kappa| \left\| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right\|_{L^\infty(\kappa)} \|\|\mathbf{K}^{-1}\|_1\|_{L^\infty(\kappa)}, \quad (11)$$

$$\|\mathbf{U}^{(n)} - \mathbf{U}^{(n_p)}\|_\infty \leq C_\infty \max_{\kappa \in \mathbb{T}} |\kappa| \left\| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right\|_{L^\infty(\kappa)} \|\|\mathbf{K}^{-1}\|_\infty\|_{L^\infty(\kappa)}, \quad (12)$$

Proof Recall that $\mathbf{M}^u(S)_{ij} = ((\mathbf{K}\lambda_t(S))^{-1} \mathbf{v}_i, \mathbf{v}_j)_\Omega$. Furthermore, for all common finite element shape functions, we have $\sum_j \mathbf{v}_j = \mathbf{e}$ with $\mathbf{e} = (1, 1)^T$ in 2D and similar in 3D.

Then

$$\begin{aligned} \left\| \mathbf{M}^{\mathbf{u}}(S^{(n-1)}) - \mathbf{M}^{\mathbf{u}}(S^{(n_p-1)}) \right\|_{\infty} &= \max_i \sum_j \left| \left(\left[\left(\mathbf{K} \lambda_t(S^{(n-1)}) \right)^{-1} - \left(\mathbf{K} \lambda_t(S^{(n_p-1)}) \right)^{-1} \right] \mathbf{v}_i, \mathbf{v}_j \right) \right|_{\Omega} \\ &= \max_i \int_{\Omega} \left| \left[\frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right] \mathbf{e}^T \mathbf{K}^{-1} \mathbf{v}_i \right| d\Omega. \end{aligned}$$

By our choice of finite element space, each of the shape functions \mathbf{v}_i is nonzero in only one vector component. Furthermore, it is nonzero only on a patch $\omega_i \subset \Omega$ that is the union of a number of cells. Let n_v be the maximal number of cells meeting at any one vertex of the triangulation \mathbb{T} , then

$$\begin{aligned} \left\| \mathbf{M}^{\mathbf{u}}(S^{(n-1)}) - \mathbf{M}^{\mathbf{u}}(S^{(n_p-1)}) \right\|_{\infty} &\leq \max_i \int_{\omega_i} \left| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right| \left\| \mathbf{K}^{-1} \right\|_1 d\omega_i \\ &\leq n_v \max_{\kappa \in \mathbb{T}} |\kappa| \left\| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right\|_{L^{\infty}(\kappa)} \left\| \mathbf{K}^{-1} \right\|_1 \Big|_{L^{\infty}(\kappa)}. \quad (13) \end{aligned}$$

The first statement of the theorem follows from this by the previous lemma, combining the norm dependent constant C and n_v into C_1 . The second claim of the theorem is proven in exactly the same way.

Remark 3 *Intuitively, one would expect $\left\| \mathbf{U}^{(n)} - \mathbf{U}^{(n_p)} \right\|_1$ to grow with the distance of $S^{(n-1)}$ from $S^{(n_p-1)}$ but be more or less independent of the spatial discretization. On the other hand, the previous theorem seems to indicate that the difference in velocities decreases with the size of mesh cells $\kappa \in \mathbb{T}$ since $h^d = |\kappa|$ appears on the right hand side (with $d = \dim \Omega$). This latter conclusion is a fallacy, however: the constants C in the previous lemma contains grows like h^{-d} , canceling this factor. Consequently, it is indeed the change in inverse mobilities $\lambda_t^{-1}(S)$ along with the permeability that determines the change in velocities.*

Following this reasoning, let us define the following indicator function derived from the first estimate of the theorem above:

$$\theta(n, n_p) = \max_{\kappa \in \mathbb{T}} \left(\left\| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right\|_{L^\infty(\kappa)} \left\| \mathbf{K}^{-1} \right\|_{L^\infty(\kappa)} \right). \quad (14)$$

(Due to the symmetry of \mathbf{K} , we have $\|\mathbf{K}^{-1}\|_1 = \|\mathbf{K}^{-1}\|_\infty$ and the second estimate of the theorem yields the same formula.) This quantity is easily and cheaply evaluated in each time step n . We then update the velocities and pressures by solving (9) whenever we find that $\theta(n, n_p) \geq \theta^*$ with θ^* a user-provided threshold.

Remark 4 *When using adaptive meshes, one could argue that the constant C and the cell size $|\kappa|$ in the estimates of the theorem do not cancel, but instead should combine to form an indicator of the kind*

$$\theta(n, n_p) = \max_{\kappa \in \mathbb{T}} \left(\frac{h_\kappa^d}{\hat{h}^d} \left\| \frac{1}{\lambda_t(S^{(n-1)})} - \frac{1}{\lambda_t(S^{(n_p-1)})} \right\|_{L^\infty(\kappa)} \left\| \mathbf{K}^{-1} \right\|_{L^\infty(\kappa)} \right), \quad (15)$$

where $h_\kappa = \text{diam } \kappa$ and \hat{h} is either an average cell diameter or the diameter of the smallest cell in the domain. Such an indicator function does make sense: it deemphasizes changes in the mobility due to changes in the saturation on small cells since small cells have less influence on the global flow field. On the other hand, one can argue that we care more about small changes on small cells since we refined them adaptively in the first place. Following this latter line of reasoning, our code uses the original criterion (14).

Remark 5 *In the common case of a diagonal permeability tensor $\mathbf{K}(\mathbf{x}) = k(\mathbf{x})\mathbf{1}$ the evaluation of (14) can be simplified using $\|\mathbf{K}^{-1}\|_{L^\infty(\kappa)} = \|k^{-1}\|_{L^\infty(\kappa)}$.*

In summary, the algorithm outlined above allows us to perform a number of saturation time steps of length $\Delta t_c^{(n)}$ until criterion (14) tells us to re-compute velocity and pressure

variables, leading to a macro time step of length

$$\Delta t_p^{(n)} = \sum_{i=n_p+1}^n \Delta t_c^{(i)}.$$

We choose the length of (micro) steps subject to the Courant-Friedrichs-Lewy (CFL) restriction according to the criterion

$$\Delta t_c = \frac{\min_K h_K}{7 \|\mathbf{u}_t\|_{L^\infty(\Omega)}}, \quad (16)$$

which we have confirmed to be stable for the choice of finite element and time stepping scheme for the saturation equation discussed below (h_K denotes the diameter of cell K). The result is a scheme where neither micro nor macro time steps are of uniform length, and both are chosen adaptively.

As an implementation detail, we note that we always solve the pressure-velocity part in the first three micro time steps to ensure accuracy at the beginning of computation, and to provide starting data to linearly extrapolate previously computed velocities to the current time step. A detailed description of the overall algorithm is shown in Algorithm 1.

Algorithm 1 Adaptive operator splitting for time steps $n \geq 3$

 $n_{pp} \leftarrow 1;$ $n_p \leftarrow 2;$ **for** $n \leftarrow 3$ **to** l **do** Compute the indicator $\theta(n, n_p)$ for recomputing \mathbf{u}_t, p from (14); **if** $\theta > \theta^*$ **then** Solve the pressure-velocity system for $\mathbf{u}_t^{(n)}, p^{(n)}$; $\mathbf{u}_t^* \leftarrow \mathbf{u}_t^{(n)}$; $n_{pp} \leftarrow n_p$; $n_p \leftarrow n$; **else** Compute \mathbf{u}_t^* by linear extrapolation to $t^{(n)}$ of $\mathbf{u}_t^{(n_p)}, \mathbf{u}_t^{(n_{pp})}$ defined at $t^{(n_p)}, t^{(n_{pp})}$; **end** Solve the saturation transport equation for $S^{(n)}$ using \mathbf{u}_t^* ;**end**

3.2 Spatial discretization and adaptive mesh refinement strategy

The IMPES scheme described above requires the separate solution of velocity/pressure and saturation equations. We discretize these on the same mesh composed of quadrilaterals or hexahedra, using continuous Q_2 elements for the velocity and Q_1 elements for the pressure; this choice satisfies the usual Ladyzhenskaya-Babuska-Brezzi (LBB) conditions [69,20]. We use continuous Q_1 finite elements to discretize the saturation equation.

As mentioned in the introduction, choosing meshes adaptively to resolve sharp saturation fronts is an essential ingredient to achieve efficiency in our algorithm. Here, we use the same shock-type refinement approach used in [24] to select those cells that should be refined or coarsened. The refinement indicator for each cell K of the triangulation is

computed by

$$\eta_K = |\nabla S_h(\mathbf{x}_K)| \quad (17)$$

where $S_h(\mathbf{x}_K)$ is the discrete saturation variable evaluated at the center of cell K . This approach is analogous to ones frequently used in compressible flow problems, where density gradients are used to indicate refinement. Cells are coarsened if $\eta_K < \theta_c = 1.1$ and refined if $\eta_K > \theta_r = 1.7$, see also [24]. Meshes resulting from this choice can be seen, for example, in Figures 2 and 5 below.

3.3 Artificial diffusion stabilization of the saturation equation

The chosen Q_1 elements for the saturation equation do not lead to a stable discretization without upwinding or other kinds of stabilization, and spurious oscillations will appear in the numerical solution. Adding an artificial diffusion term is one approach to eliminating these oscillations [20]. On the other hand, adding too much diffusion smears sharp fronts in the solution and suffers from grid-orientation difficulties [20]. To avoid these effects, we use the artificial diffusion term proposed by Guermond and Pasquetti [25] and already validated in [24].

This method modifies the (discrete) weak form of the saturation equation (3) to read

$$\left(\epsilon \frac{\partial S_h}{\partial t}, \varphi_h \right) - (\mathbf{u}_t F(S_h), \nabla \varphi) + (\mathbf{n} \cdot \mathbf{u}_t \hat{F}(S_h), \varphi)_{\partial \Omega} + (\nu(S_h) \nabla S_h, \nabla \varphi_h) = 0 \quad \forall \varphi_h, \quad (18)$$

where ν is the artificial diffusion parameter and \hat{F} is an appropriately chosen numerical flux on the boundary of the domain (we choose the obvious full upwind flux for this).

Following Guermond and Pasquetti [25], we use the parameter as a piecewise constant

function set on each cell K with the diameter h_K as

$$\nu(S)|_K = \beta \|\mathbf{u}_t\|_{L^\infty(K)} \min \left\{ h_K, h_K^\alpha \frac{\|\text{Res}(S)\|_{L^\infty(K)}}{c(\mathbf{u}_t, S)} \right\} \quad (19)$$

where α is a stabilization exponent and β is a dimensionless user-defined stabilization constant. Following Guermond and Pasquetti [25] as well as Kronbichler and Bangerth [70], the velocity and saturation global normalization constant, $c(\mathbf{u}_t, S)$, and the residual $\text{Res}(S)$ are respectively given by

$$c(\mathbf{u}_t, S) = c_R \|\mathbf{u}_t\|_{L^\infty(\Omega)} \text{var}(S) |\text{diam}(\Omega)|^{\alpha-2} \quad (20)$$

and

$$\text{Res}(S) = \left(\epsilon \frac{\partial S}{\partial t} + \mathbf{u}_t \cdot \nabla F(S) + F(S)q \right) \cdot S^{\alpha-1} \quad (21)$$

where c_R is a second dimensionless user-defined constant, $\text{diam}(\Omega)$ is the diameter of the domain and $\text{var}(S) = \max_\Omega S - \min_\Omega S$ is the range of the present saturation values in the entire computational domain Ω .

This stabilization scheme has a number of advantages over simpler schemes such as finite volume (or discontinuous Galerkin) methods or streamline upwind Petrov Galerkin (SUPG) discretizations. In particular, the artificial diffusion term acts primarily in the vicinity of discontinuities since the residual is small in areas where the saturation is smooth. It therefore provides for a higher degree of accuracy. On the other hand, it is nonlinear since ν depends on the saturation S . We avoid this difficulty by using an explicit Euler time stepping method, which leads to the following fully discrete problem at time step n :

$$\begin{aligned} (\epsilon S_h^{(n)}, \varphi_h) &= (\epsilon S_h^{(n-1)}, \varphi_h) + \Delta t_c^{(n)} \left(\mathbf{u}_t^* F(S_h^{(n-1)}), \nabla \varphi \right) \\ &\quad - \Delta t_c^{(n)} \left(\mathbf{n} \cdot \mathbf{u}_t^* \hat{F}(S_h^{(n-1)}), \varphi \right)_{\partial\Omega} - \Delta t_c^{(n)} (\nu(S_h^{(n-1)}) \nabla S_h^{(n-1)}, \nabla \varphi_h) \quad \forall \varphi_h, \end{aligned} \quad (22)$$

where \mathbf{u}_t^* is the velocity linearly extrapolated from $\mathbf{u}_t^{(n_p)}$ and $\mathbf{u}_t^{(n_{pp})}$ to the current time

$t^{(n)}$ (see Algorithm 1). Consequently, the equation is linear in $S_h^{(n)}$ and all that is required is to solve with a mass matrix on the saturation space.

3.4 Linear system and its preconditioning

Following the discretization of the governing equations (1)–(3) discussed above, we obtain a linear system of equations in time step (n) of the following form:

$$\begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{H} & \mathbf{0} & \mathbf{M}^S \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n)} \\ \mathbf{P}^{(n)} \\ \mathbf{S}^{(n)} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{F}_2 \\ \mathbf{F}_3 \end{pmatrix} \quad (23)$$

where the individual matrices and vectors are defined as follows using shape functions \mathbf{v}_i for velocity, and ϕ_i for both pressure and saturation:

$$\mathbf{M}_{ij}^u = \left((\mathbf{K}\lambda_t(S^{(n-1)}))^{-1} \mathbf{v}_i, \mathbf{v}_j \right)_\Omega, \quad \mathbf{M}_{ij}^S = (\epsilon\phi_i, \phi_j)_\Omega \quad (24)$$

$$\mathbf{B}_{ij} = -(\nabla \cdot \mathbf{v}_j, \phi_i)_\Omega, \quad \mathbf{H}_{ij} = -\Delta t_c^{(n)} \left(F(S^{(n-1)}) \mathbf{v}_i, \nabla \phi_j \right)_\Omega \quad (25)$$

$$(\mathbf{F}_2)_i = -\left(F(S^{(n-1)}) q, \phi_i \right)_\Omega, \quad (26)$$

and \mathbf{F}_3 as given in (22).

The linear system above is of block triangular form if we consider the top left 2×2 panel of matrices as one block. We can therefore first solve for the velocity and pressure (unless we decide to use $\mathbf{U}^{(n_p)}$ in place of the velocity, as discussed in Section 3.1) followed by a

solve for the saturation variable. The first of these steps requires us to solve

$$\begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n)} \\ \mathbf{P}^{(n)} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{F}_2 \end{pmatrix} \quad (27)$$

We apply the Generalized Minimal Residual (GMRES) method (Saad and Schultz [71]) to this linear system. The ideal preconditioner for the system (27) is

$$\mathbf{P} = \begin{pmatrix} \mathbf{M}^u & \mathbf{0} \\ \mathbf{B} & -\mathbf{S} \end{pmatrix}, \quad \mathbf{P}^{-1} = \begin{pmatrix} (\mathbf{M}^u)^{-1} & \mathbf{0} \\ \mathbf{S}^{-1}\mathbf{B}(\mathbf{M}^u)^{-1} & -\mathbf{S}^{-1} \end{pmatrix} \quad (28)$$

where $\mathbf{S} = \mathbf{B}(\mathbf{M}^u)^{-1}\mathbf{B}^T$ is the Schur complement [72] of the system. This preconditioner is optimal since

$$\mathbf{P}^{-1} \begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & (\mathbf{M}^u)^{-1}\mathbf{B}^T \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \quad (29)$$

and consequently all eigenvalues are equal to one. GMRES with this preconditioner would then converge in one iteration.

However, we cannot of course expect to use exact inverses of the velocity mass matrix and the Schur complement. We therefore follow the approach by Silvester and Wathen [68] originally proposed for the Stokes system. Adapting it to the current set of equations yield the preconditioner

$$\tilde{\mathbf{P}}^{-1} = \begin{pmatrix} \widetilde{(\mathbf{M}^u)^{-1}} & \mathbf{0} \\ \widetilde{\mathbf{S}^{-1}\mathbf{B}(\mathbf{M}^u)^{-1}} & -\widetilde{\mathbf{S}^{-1}} \end{pmatrix} \quad (30)$$

where a tilde indicates an approximation of the exact inverse matrix. In particular, since

$(\mathbf{M}^u)^{-1} = \left((\mathbf{K}\lambda_t)^{-1} \mathbf{v}_i, \mathbf{v}_j \right)_\Omega$ is a sparse symmetric and positive definite matrix, we choose for $\widetilde{(\mathbf{M}^u)^{-1}}$ a single application of a sparse incomplete Cholesky decomposition of this matrix [73]. We note that the Schur complement that corresponds to the porous media flow operator in non-mixed form, $-\nabla \cdot [\mathbf{K}\lambda_t(S)]\nabla$ and $\tilde{\mathbf{S}} = ((\mathbf{K}\lambda_t) \nabla\phi_i, \nabla\phi_j)_\Omega$ should be a good approximation of the actual Schur complement matrix \mathbf{S} . Since both of these matrices are again symmetric and positive definite, we use an incomplete Cholesky decomposition of $\tilde{\mathbf{S}}$ for $\widetilde{\mathbf{S}^{-1}}$. It is important to note that $\tilde{\mathbf{S}}$ needs to be built with Dirichlet boundary conditions to ensure its invertibility.

Once the velocity is available $\mathbf{U}^{(n)} \equiv \mathbf{u}_t^*$ (see operator splitting Algorithm 1), we can assemble \mathbf{H} and \mathbf{F}_3 and solve for the saturations using

$$\mathbf{M}^S \mathbf{S}^{(n)} = \mathbf{F}_3 - \mathbf{H}\mathbf{U}^{(n)}. \quad (31)$$

where the mass matrix \mathbf{M}^S is solved by the conjugate gradient method, using an incomplete Cholesky decomposition as preconditioner once more.

4 Numerical test cases

We have implemented our methods in a program based on the C++ `deal.II` library [17,18]. The implementation discussed here uses and extends parts of the step-21, step-31 and step-33 tutorial programs of this library [26,74,70]. We use the implementation of the incomplete Cholesky decomposition provided by the `Trilinos` library [75].

In the Introduction, we show numerical results that illustrate the efficiency and accuracy of our combined methods in solving Equations (1)–(3) augmented by appropriate initial and boundary in conjunction with two different choices of the permeability model. In the problems considered, there is no internal source term ($q = 0$).

For simplicity, we choose $\Omega = [0, 1]^d$, $d = 2, 3$, though all methods (as well as our implementation) should work equally well on general unstructured meshes.

4.1 Initial and boundary conditions

Initial conditions are only required for the saturation variable, and we choose $S(\mathbf{x}, 0) = 0$, i.e. the porous medium is initially filled by the non-wetting phase. We prescribe a linear pressure on the boundaries:

$$p(\mathbf{x}, t) = 1 - x \quad \text{on} \quad \partial\Omega \times [0, T]. \quad (32)$$

As discussed in Section 2, pressure and saturation uniquely determine a velocity, and the velocity determines whether a boundary segment is an inflow or outflow boundary. On the inflow part of the boundary, $\Gamma_{in}(t)$, we impose

$$S(\mathbf{x}, t) = 1 \quad \text{on} \quad \Gamma_{in}(t) \cap \{x = 0\}, \quad (33)$$

$$S(\mathbf{x}, t) = 0 \quad \text{on} \quad \Gamma_{in}(t) \setminus \{x = 0\}. \quad (34)$$

In other words, the domain is flooded by the wetting phase from the left. No boundary conditions for the saturation are required for the outflow parts of the boundary.

4.2 Absolute permeability model

In Section 5, we consider heterogeneous but isotropic porous media, i.e. we choose the permeability tensor as

$$\mathbf{K}(\mathbf{x}) = k(\mathbf{x}) \cdot \mathbf{1}. \quad (35)$$

We consider the two porous medium models used previously by Li and Bangerth [26]. The first test case corresponds to a single crack along a sine curve:

$$k(\mathbf{x}) = k_{sm}(\mathbf{x}) = \max \left\{ \exp \left(- \left(\frac{y - 0.5 - 0.1 \cdot \sin(10x)}{0.1} \right)^2 \right), 0.01 \right\}. \quad (36)$$

The second test case corresponds to a porous medium with random permeability prescribed using

$$k(\mathbf{x}) = k_{rm}(\mathbf{x}) = \min \left\{ \max \left\{ \sum_{l=1}^N \Psi_l(\mathbf{x}), 0.01 \right\}, 4 \right\}, \quad (37)$$

$$\Psi_l(\mathbf{x}) = \exp \left(- \left(\frac{|\mathbf{x} - \mathbf{x}_l|}{0.05} \right)^2 \right).$$

The N centers \mathbf{x}_l of high permeability regions are randomly chosen in Ω . We will present results for $N = 50$, and $N = 200$. The corresponding three permeability fields are shown in Figure 1. The two permeability fields are truncated from above and below to ensure a reasonable limit on the heterogeneous contrast.

All the numerical and physical parameters used for the 3D cases are listed in Table 1.

5 Results and discussion

In this section, we first validate the methods by solving a two-dimensional test case and comparing solutions to those previously obtained with a simple finite element method. We then demonstrate the approach for three-dimensional computations and in particular show the speedup obtained from our methods.

Table 1

Parameters used in the model

PARAMETER	SYMBOL	VALUE	UNIT
Porosity	ϵ	1.0	-
Viscosity (wetting)	μ_w	0.2	$kg \cdot m^{-1} \cdot sec^{-1}$
Viscosity (nonwetting)	μ_{nw}	1.0	$kg \cdot m^{-1} \cdot sec^{-1}$
Stabilization exponent	α	1.0	-
Stabilization constant	β	0.27	-
Normalization constant	c_R	1.0	-
Number of high-permeability regions	N	50; 200	-
Operator splitting threshold	θ^*	5.0	-

5.1 2D validation

We compare results obtained with the adaptive grid and operator splitting methods to previous work using uniform grids in which Raviart-Thomas finite elements [76] are used for velocity and piecewise constant, discontinuous Galerkin (DG) elements for pressure and saturation. This combination satisfies the usual stability condition [69]. The results were obtained by the program used in [26].

We consider the 2D version of the first test case laid out in the previous section. Parameters used in our program are those listed in Table 1 except that the stabilization constant β which is chosen as 0.3.

First, consider the case when the pressure-velocity system is solved once every 10 saturation time steps (fixed operator splitting). Figures 2a and 2b compare saturation contours in the single crack case between these two discretizations at $t = 0.589$ in the single crack case. Figures 2c and 2d show saturation profiles along $y = 0.5$ and $x = 0.2$, respectively. The results are essentially identical. Figure 3 shows the same saturation profiles but where the pressure-velocity system is solved only once every 30, 60, and 90 saturation time steps; it is obvious that the errors are getting more noticeable as the frequency of solving the pressure-velocity part decreases.

In contrast to this, when we drive the operator splitting by criterion (14) with $\theta^* = 5$, the difference between the two methods vanishes (see Figure 4), justifying our proposed indicator for operator splitting.

5.2 3D experiments

We now consider three-dimensional simulations. The number of the cells in each direction is initially set to 8 (512 total cells) and the number of degrees of freedom (DoFs) is 16,197 in this simulation. Figures 5 and 6 show the saturation distribution and their corresponding adaptive grids as simulations progress. Due to the heterogeneity of the porous media, only a few areas of higher wetting-phase saturation are formed in high-permeability regions. Figures 7 and 8 show volume renderings of the invading wetting phase using iso-surface contours.

5.3 Performance comparison

The number of saturation time steps after which the velocity and pressure variables should be solved again is shown in Figure 9 as determined by criterion (14) for two different values

of the threshold θ^* . Not surprisingly, choosing a larger value for the threshold relaxes the constraints and leads to more saturation time steps between macro solves. In both cases these numbers increase as the simulation progresses. This can be explained by observing that flow initially happens along only a few fingers but then spreads out across higher permeability areas; the average velocity of the front therefore declines, leading to smaller saturation changes per time step.

To illustrate the performance of our combination of h -adaptive meshes, adaptive operator splitting, and block matrix preconditioning methods, we compare the run time of different combinations of these methods (adaptive mesh, adaptive mesh + adaptive operating splitting, and adaptive mesh + adaptive operating splitting + block matrix preconditioning) in 2D and 3D simulations. The number of degrees of freedom ranges from 2,176 to 41,732 in 2D, and from 63,490 to 449,999 in 3D. All other conditions, including the use of the same CPU specifications, are identical between simulations, and we use the same parameters as well as the same minimum element size h . The base line code uses a uniform mesh, solves the velocity/pressure system in each saturation time step, and uses a Schur complement solver for this [26].

Table 2 shows run times for the various combinations. Mesh adaptivity alone improves performance in the 2D/3D cases by about 43%. Operator splitting plus adaptive grids (third column), the case in Table 2 shows an up to 92% reduction in computing time compared with using uniform grids and solving the velocity/pressure system in each time step. Finally, the combination of all methods considered herein reduces compute time by a factor of 160–180, making computations possible that would otherwise be unachievable. Using well known techniques, these excellent compute times could be further accelerated by parallelizing operations across multiple cores of a single computer, or across the nodes of a cluster.

Table 2

Comparison of performance (testing total wall clock time from $t = 0$ to $t = 2$ in the case of $N = 200$ and $\theta^* = 5$) between four numerical combinations: Uniform Mesh (“UM”), Adaptive Mesh (“AM”), Adaptive Mesh plus Adaptive Operator Splitting (“AM+AOS”) and Adaptive Mesh plus Adaptive Operator Splitting plus Block Preconditioning (“AM+AOS+BP”).

	UM	AM	AM+AOS	AM+AOS +BP
Compute time in 2D (days)	7.2	4.2	0.2	0.04
Compute time in 3D (days)	96.5	55.9	7.5	0.58

6 Conclusions and Future Work

Motivated by computational expense of the methods that are traditionally used for this task, we have presented efficient integrated numerical methods allowing high-resolution simulation of two-phase flow in porous media. The methods are based on a finite element discretization using continuous elements, and consist of (i) an adaptive operator splitting method that only recomputes the velocity and pressure variables whenever necessary, as determined by a newly proposed objective indicator; (ii) block matrix preconditioning methods that greatly reduce the compute time needs; (iii) an entropy-based stabilizing term that preserves accuracy and ensures stability; (iv) locally adaptive refinement allowing highly resolved time dependent simulations. The robustness and effectiveness of the numerical methods were demonstrated through 2D/3D simulations and comparing both the accuracy of solutions as well as the required CPU time.

More complex multiphase flows involve additional physical processes, such as multiple interacting phases or capillary effects with a capillary pressure relationship that is fundamen-

tally dependent on material properties and operating conditions, saturation, wettability, and other material properties [77–79]. The addition of such terms obviously complicates the computation of solutions, but we believe that the techniques described herein can readily be extended to such problems. In particular, following a similar mathematical procedure to that described in Section 3.1, it should be possible to extend our operator splitting indicator to these more complex situations.

7 Acknowledgments

The work of the first and third authors was funded through the Canada Research Chairs Program and the MITACS Network of Centres of Excellence. Parts of the work by the second author were funded through Award No. KUS-C1-016-04, made by the King Abdullah University of Science and Technology, and through an Alfred P. Sloan Research Fellowship. Some computational resources used here were also funded by NSF award DMS-0922866.

Thanks are due to Dr. Zhangxin Chen (University of Calgary) for his invaluable insights on finite element theory, to Dr. Sergey Mashchenko (McMaster University) for installing the relevant libraries, and to the first author’s wife, Mrs. Hsiao-Yun Janette Cheng (PhD candidate, University of Sheffield), for helping with the graphics.

Access to the Shared Hierarchical Academic Research Computing Network (SHARCNET, <http://www.sharcnet.ca>) and Compute/Calcul Canada is gratefully acknowledged.

References

- [1] K Aziz and A Settari. *Petroleum Reservoir Simulation*. Applied Science Publishers LTD, London, 1979.

- [2] DW Peaceman. *Fundamentals of Numerical Reservoir Simulation*. Elsevier, Amsterdam, 1977.
- [3] MG Gerritsen and LJ Durlofsky. Modeling fluid flow in oil reservoirs. *Annual Review of Fluid Mechanics*, 37(1):211–238, 2005.
- [4] D McKenzie. The extraction of magma from the crust and mantle. *Earth Planet. Sc. Letters*, 74:81–91, 1985.
- [5] J Larminie and A Dicks. *Fuel Cell Systems Explained*. Wiley, 2006.
- [6] MM Mench. *Fuel Cell Engines*. Wiley, 2008.
- [7] N Djilali. Computational modelling of polymer electrolyte membrane (PEM) fuel cells: challenges and opportunities. *Energy*, 32(4):269–280, 2007.
- [8] T Berning and N Djilali. A 3D, multiphase, multicomponent model of the cathode and anode of a PEM fuel cell. *Journal of The Electrochemical Society*, 150(12):A1589–A1598, 2003.
- [9] A Taniguchi, T Akita, K Yasuda, and Y Miyazaki. Analysis of electrocatalyst degradation in PEMFC caused by cell reversal during fuel starvation. *Journal of Power Sources*, 130(1-2):42–49, 2004.
- [10] M Piri and MJ Blunt. Three-dimensional mixed-wet random pore-scale network modeling of two- and three-phase flow in porous media. I. Model description. *Phys. Rev. E*, 71(2):026301, Feb 2005.
- [11] M Piri and MJ Blunt. Three-dimensional mixed-wet random pore-scale network modeling of two- and three-phase flow in porous media. II. Results. *Phys. Rev. E*, 71(2):026302, Feb 2005.
- [12] MJ Blunt. Flow in porous media – pore-network models and multiphase flow. *Current Opinion in Colloid and Interface Science*, 6(3):197–207, 2001.

- [13] B Markicevic, A Bazylak, and N Djilali. Determination of transport parameters for multiphase flow in porous gas diffusion electrodes using a capillary network model. *Journal of Power Sources*, 171(2):706–717, 2007.
- [14] A Bazylak, V Berejnov, B Markicevic, D Sinton, and N Djilali. Numerical and microfluidic pore networks: Towards designs for directed water transport in GDLs. *Electrochimica Acta*, 53(26):7630–7637, 2008.
- [15] DA Wolf-Gladrow. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models: An Introduction*. Lecture Notes in Mathematics, Springer-Verlag, 2000.
- [16] S Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Oxford Science Press, 2001.
- [17] W Bangerth, R Hartmann, and G Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [18] W Bangerth, R Hartmann, and G Kanschat. deal.II *Differential Equations Analysis Library, Technical Reference*, 2010. <http://www.dealii.org/>.
- [19] B van Leer. Upwinding and high-resolution methods for compressible flow: from donor cell to residual-distribution schemes. *Communications in Computational Physics*, 1:192–206, 2006.
- [20] Z Chen. *Finite Element Methods and Their Applications*. Springer, 2005.
- [21] D Kröner. *Numerical Schemes for Conservation Laws*. Wiley/Teubner, 1997.
- [22] R Helmig. *Multiphase Flow and Transport Processes in the Subsurface*. Springer-Verlag, Heidelberg, 1997.
- [23] M von Paul. *Simulation of Two-phase Flow Processes in Heterogeneous Porous Media with Adaptive Methods*. Dissertation, Universität Stuttgart, Institut für Wasserbau, 2003.

- [24] CC Chueh, M Secanell, W Bangerth, and N Djilali. Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media. *Computers and Fluids*, 39:1585–1596, 2010.
- [25] JL Guermond and R Pasquetti. Entropy-based nonlinear viscosity for fourier approximations of conservation laws. *Comptes Rendus Mathematique*, 346(13-14):801–806, 2008.
- [26] Y Li and W Bangerth. *The deal.II tutorial manual: step-21*. http://dealii.org/developer/doxygen/deal.II/step_21.html.
- [27] I Babuška and WC Rheinboldt. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis*, 15(4):736–754, 1978.
- [28] M Ainsworth and JT Oden. *A Posteriori Error Estimation in Finite Element Analysis*. John Wiley and Sons, 2000.
- [29] R Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques*. Wiley/Teubner, New York, Stuttgart, 1996.
- [30] GF Carey. *Computational Grids: Generation, Adaptation and Solution Strategies*. Taylor & Francis, 1997.
- [31] W Bangerth and R Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser Verlag, 2003.
- [32] P Ellsiepen. *Zeit- und ortsadaptive Verfahren angewandt auf Mehrphasenprobleme poröser Medien*. Dissertation, Universität Stuttgart, Institut für Mechanik, Lehrstuhl II, 1999.
- [33] R Hinkelmann. *Efficient Numerical Methods and Information-Processing Techniques in Environment Water*. Habilitation, Universität Stuttgart, 2002.
- [34] H Tang and T Tang. Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 41(2):487, 2003.
- [35] Z Tan, Z Zhang, Y Huang, and T Tang. Moving mesh methods with locally varying time steps. *Journal of Computational Physics*, 200(1):347–367, 2004.

- [36] Z Tan, T Tang, and Z Zhang. A simple moving mesh method for one- and two-dimensional phase-field equations. *Journal of Computational and Applied Mathematics*, 190(1-2):252–269, 2006.
- [37] Y Di, R Li, T Tang, and P Zhang. Moving mesh finite element methods for the incompressible Navier-Stokes equations. *SIAM Journal on Scientific Computing*, 26(3):1036–1056, 2005.
- [38] Y Di, R Li, T Tang, and P Zhang. Moving mesh methods for singular problems on a sphere using perturbed harmonic mappings. *SIAM Journal on Scientific Computing*, 28(4):1490–1508, 2006.
- [39] Y Di, R Li, and T Tang. A general moving mesh framework in 3d and its application for simulating the mixture of multi-phase flows. *Communications in Computational Physics*, 3:582–602, 2008.
- [40] R Li and T Tang. Moving mesh discontinuous galerkin method for hyperbolic conservation laws. *Journal of Scientific Computing*, 27:347–363, 2006.
- [41] YB Zhang, H Wang, and T Tang. Simulating two-phase viscoelastic flows using moving finite element methods. *Communications in Computational Physics*, 7:333–349, 2010.
- [42] MJ Berger and P Colella. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, 82(1):64–84, 1989.
- [43] DW Peaceman and HH Rachford. The numerical solution of parabolic and elliptic equations. *J. Soc. Indust. Appl. Math.*, 3:28–41, 1955.
- [44] J Douglas. On the numerical integration of $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{\partial u}{\partial t}$. *J. Soc. Indust. Appl. Math.*, 3:42–65, 1955.
- [45] J Douglas and DW Peaceman. Numerical solution of two-dimensional heat flow problems. *American Institute of Chemical Engineering Journal*, 1:505–512, 1955.
- [46] J Douglas and DW Peaceman. On the numerical solution of heat conduction problems in two and three space variables. *Transaction of the American Mathematical Society*, 82:421–439,

1960.

- [47] J Douglas, DW Peaceman, and HH Rachford. A method for calculating multi-dimensional immiscible displacement. *Trans. of AIME*, 216:297–308, 1959.
- [48] EG D'yakonov. On an iterative method for the solution of a system of finite-difference equations. *DAN SSSR*, 143:522–525, 1961.
- [49] NN Yanenko. *Méthode a Pas Fractionnaires*. Librairie Armand Colin, Paris, 1968.
- [50] MS Espedal and RE Ewing. Characteristic petrov-galerkin subdomain methods for two-phase immiscible flow. *Computer Methods in Applied Mechanics and Engineering*, 64(1-3):113–135, 1987.
- [51] HK Dahle. *Operator Splitting Techniques for Convection-Dominated Diffusion Problems in One and Two Space Dimensions*. PhD Thesis, Department of Mathematics, University of Bergen, Norway, 1988.
- [52] MS Espedal and KH Karlsen. *Numerical Simulation of Reservoir Flow Models Based on Large Time Step Operator Splitting Algorithms*, volume 1734. Lecture Notes in Mathematics (Springer-Verlag, New York), 2000.
- [53] KH Karlsen, KA Lie, JR Natvig, HF Nordhaug, and HK Dahle. Operator splitting methods for systems of convection-diffusion equations: Nonlinear error mechanisms and correction strategies. *Journal of Computational Physics*, 173(2):636–663, 2001.
- [54] JW Sheldon, B Zondek, and WT Cardwell. One-dimensional, incompressible, non-capillary, two-phase fluid flow in a porous medium. *T. SPE AIME*, 216:290–296, 1959.
- [55] HL Stone and Jr. Garder. Analysis of gas-cap or dissolved-gas reservoirs. *T. SPE AIME*, 222:92–104, 1961.
- [56] Z Chen. *Computational Methods for Multi-phase Flows in Porous Media*. SIAM, 2006.
- [57] Z Chen, G Huan, and B Li. An improved impes method for two-phase flow in porous media. *Transport in Porous Media*, 54:361–376, 2004.

- [58] E Abreu, J Douglas, F Furtado, and F Pereira. Operating splitting for three-phase flow in heterogeneous porous media. *Communications in Computational Physics*, 6:72–84, 2009.
- [59] GA Cederberg, RL Street, and JO Leckie. A groundwater mass transport and equilibrium chemistry model for multicomponent systems. *Water Resources Research*, 21(8):1095–1104, 1985.
- [60] AL Walter, EO Frind, DW Blowes, CJ Ptacek, and JW Molson. Modeling of multicomponent reactive transport in groundwater 1. model development and evaluation. *Water Resources Research*, 30(11):3137–3148, 1994.
- [61] A Zysset, F Stauffer, and T Dracos. Modeling of chemically reactive groundwater transport. *Water Resources Research*, 30(7):2217–2228, 1994.
- [62] LSJ Bell and PJ Binning. A split operator approach to reactive transport with the forward particle tracking eulerian lagrangian localized adjoint method. *Advances in Water Resources*, 27(4):323–334, 2004.
- [63] AI James and JW Jawitz. Modeling two-dimensional reactive transport using a godunov-mixed finite element method. *Journal of Hydrology*, 338(1-2):28–41, 2007.
- [64] T Rusten and R Winther. A preconditioned iterative method for saddlepoint problems. *SIAM Journal on Matrix Analysis and Applications*, 13(3):887–904, 1992.
- [65] DN Arnold, RS Falk, and R Winther. Preconditioning $H(\text{div})$ and applications. *Mathematics of Computation*, 66:957–984, 1997.
- [66] CE Powell and D Silvester. Optimal preconditioning for raviart–thomas mixed formulation of second-order elliptic problems. *SIAM Journal on Matrix Analysis and Applications*, 25(3):718–738, 2003.
- [67] CE Powell. Parameter-free $H(\text{div})$ preconditioning for a mixed finite element formulation of diffusion problems. *IMA J Numer Anal*, 25(4):783–796, 2005.

- [68] D Silvester and A Wathen. Fast iterative solution of stabilised Stokes systems part ii: Using general block preconditioners. *SIAM Journal on Numerical Analysis*, 31(5):1352–1367, 1994.
- [69] F Brezzi and M Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, 1991.
- [70] D Neckels and W Bangerth. *The deal.II tutorial manual, step-33*. http://dealii.org/developer/doxygen/deal.II/step_33.html.
- [71] Y Saad and MH Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [72] F Zhang. *The Schur Complement and its Applications*. Springer, 2005.
- [73] GH Golub and CF van Loan. *Matrix Computations*. 3rd Edition, Johns Hopkins, 1996.
- [74] M Kronbichler and W Bangerth. *The deal.II tutorial manual: step-31*. http://dealii.org/developer/doxygen/deal.II/step_31.html.
- [75] MA Heroux, RA Bartlett, VE Howle, RJ Hoekstra, JJ Hu, TG Kolda, RB Lehoucq, KR Long, RP Pawlowski, ET Phipps, AG Salinger, HK Thornquist, RS Tuminaro, JM Willenbring, A Williams, and KS Stanley. An overview of the trilinos project. *ACM Trans. Math. Softw.*, 31(3):397–423, 2005.
- [76] PA Raviart and JM Thomas. *A Mixed Finite Element Method for 2nd Order Elliptic Problems*, volume 606. In: *Mathematical Aspects of the Finite Element Method Lecture Notes in Mathematics* (Springer-Verlag, Berlin), 1977.
- [77] EC Kumbur, KV Sharp, and MM Mench. Validated leverett approach for multiphase flow in pefc diffusion media I. Hydrophobicity effect. *Journal of The Electrochemical Society*, 154(12):B1295–B1304, 2007.
- [78] EC Kumbur, KV Sharp, and MM Mench. Validated leverett approach for multiphase flow in pefc diffusion media II. Compression effect. *Journal of The Electrochemical Society*, 154(12):B1305–B1314, 2007.

[79] EC Kumbur, KV Sharp, and MM Mench. Validated leverett approach for multiphase flow in pefc diffusion media III. Temperature effect and unified approach. *Journal of The Electrochemical Society*, 154(12):B1315–B1324, 2007.

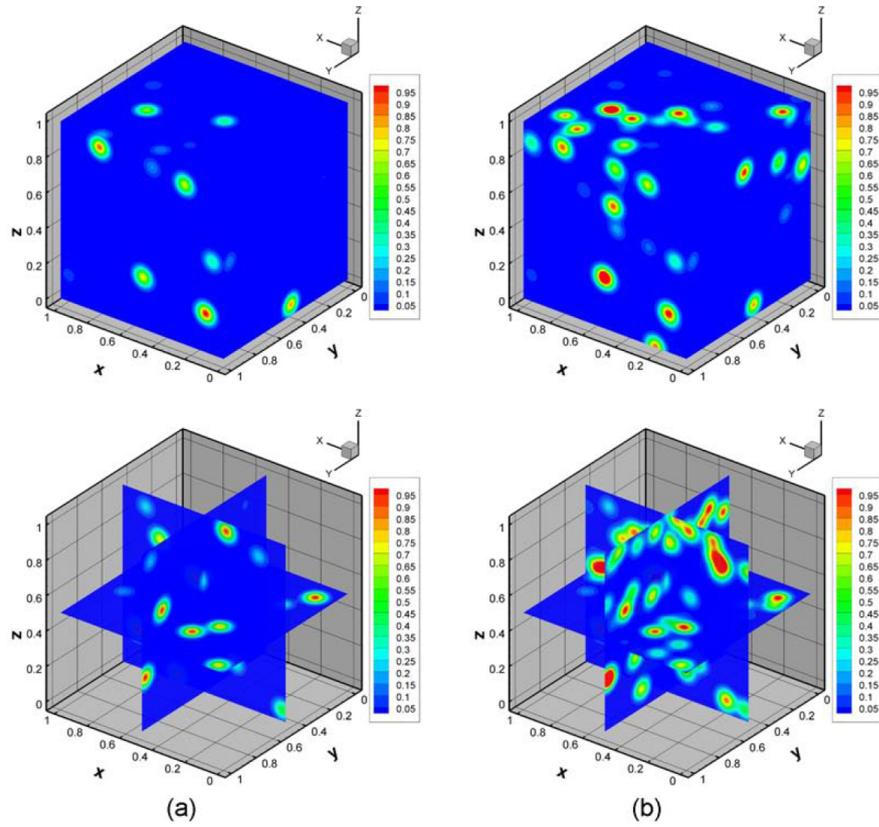


Fig. 1. Permeability fields of random porous media (top: 3D contours, bottom: surface contours inside the entire 3D domain at $x = 0.5$ of $Y - Z$ plane, $y = 0.5$ of $X - Z$ plane, $z = 0.5$ of $X - Y$ plane) (a) $N = 50$. (b) $N = 200$.

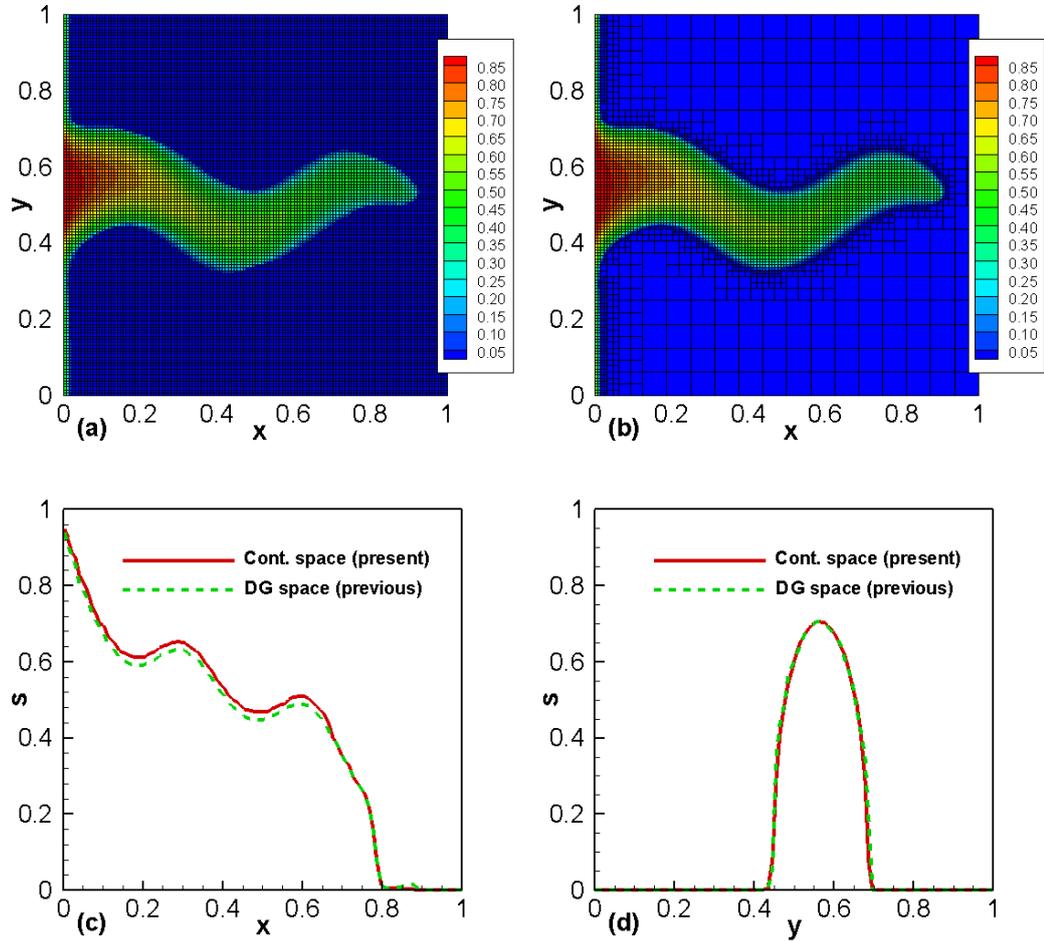


Fig. 2. Numerical results for the single crack medium in comparison with previous work of Li and Bangerth [26] at $t = 0.589$ when the pressure-velocity system is solved once every 10 saturation time steps (fixed operator splitting). (a) Saturation field using discontinuous Galerkin space (previous work). (b) Saturation field using continuous space with the stabilizing term and operator splitting (present work). (c) Saturation profile along $y = 0.5$. (d) Saturation profile along $x = 0.2$.

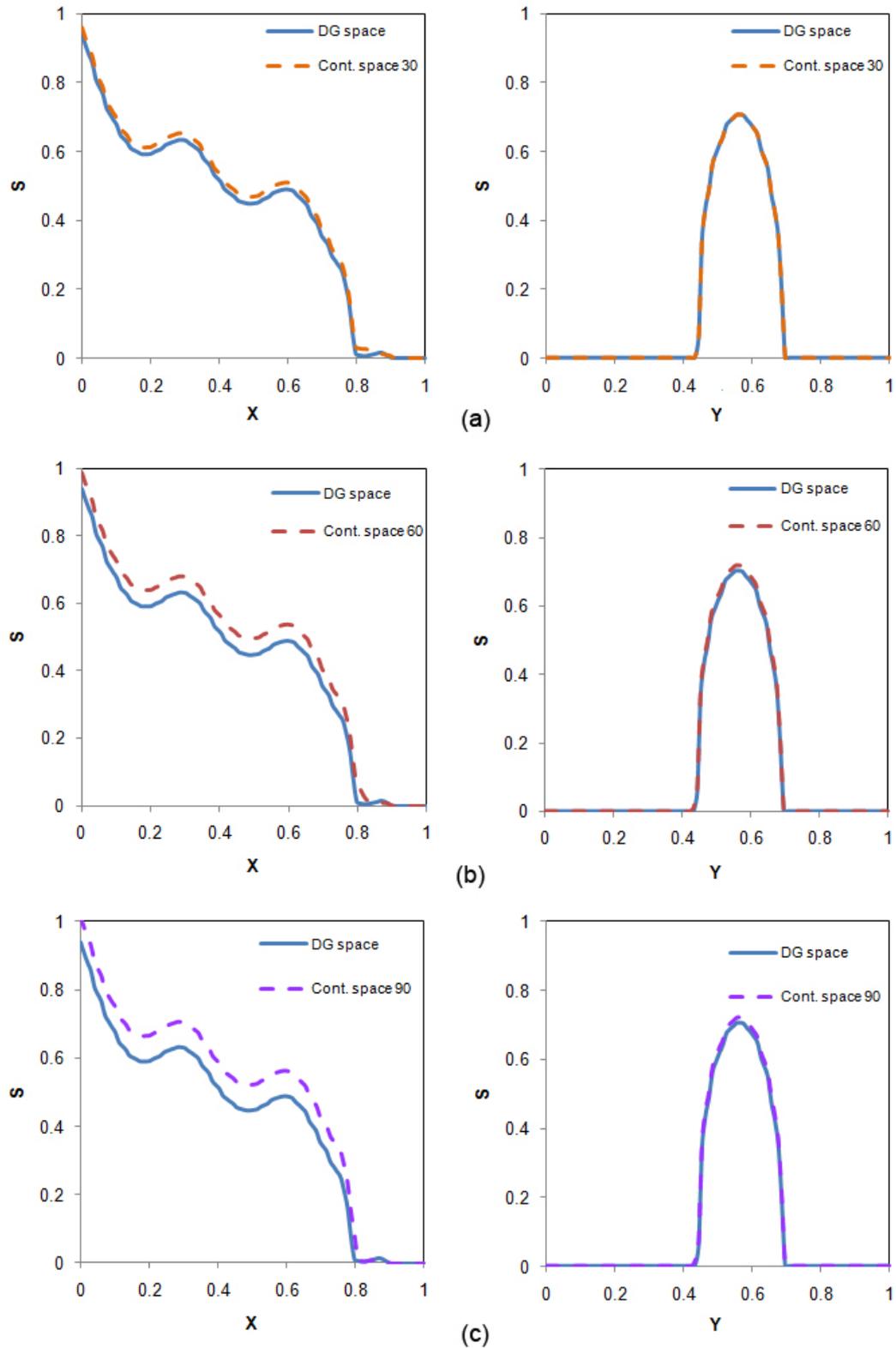


Fig. 3. Saturation profiles along $y = 0.5$ (left) and along $x = 0.2$ (right) for the single crack medium in comparison with previous work of Li and Bangerth [26] at $t = 0.589$ when the pressure-velocity system is solved once every (a) 30, (b) 60, or (c) 90 saturation time steps (fixed operator splitting).

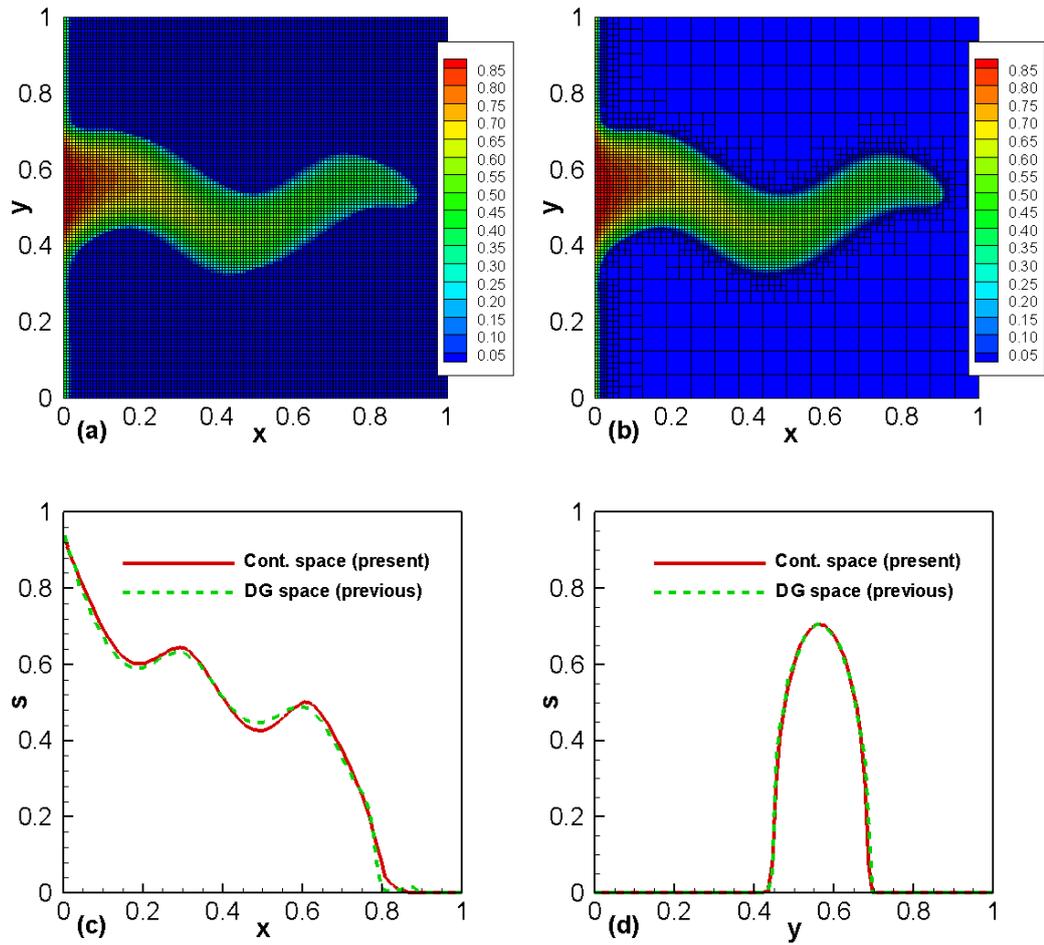


Fig. 4. Numerical results for the same situation as in Fig. 2 but when the decision to solve the pressure-velocity system is made adaptively using criterion (14) with $\theta^* = 5$.

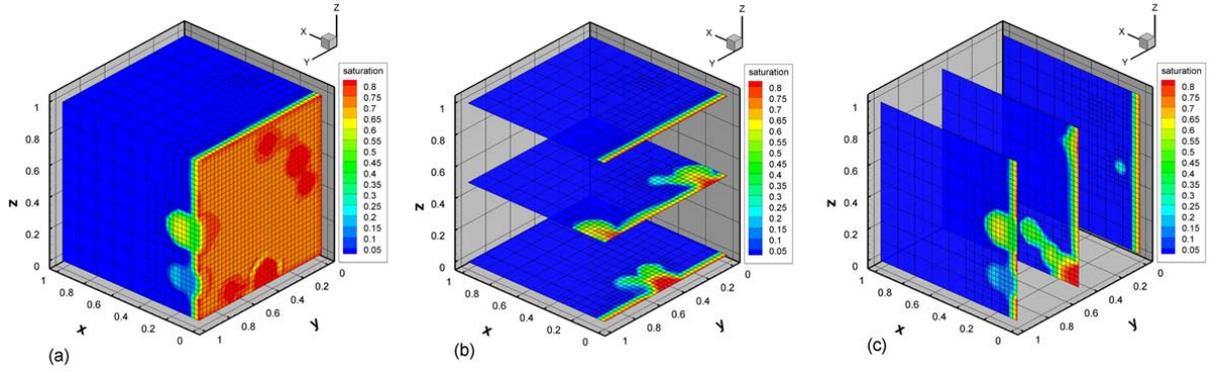


Fig. 5. Saturation distribution contours in a random porous medium (eq. (37), $N = 50$) with corresponding adaptive grids at $t = 2.58$ with 7,820 elements and 242,771 degrees of freedom: (a) 3D contour. (b) Surface contours of $X - Y$ plane at $z = 0.0, 0.5, 1.0$. (c) Surface contours of $X - Z$ planes at $y = 0.0, 0.5, 1.0$.

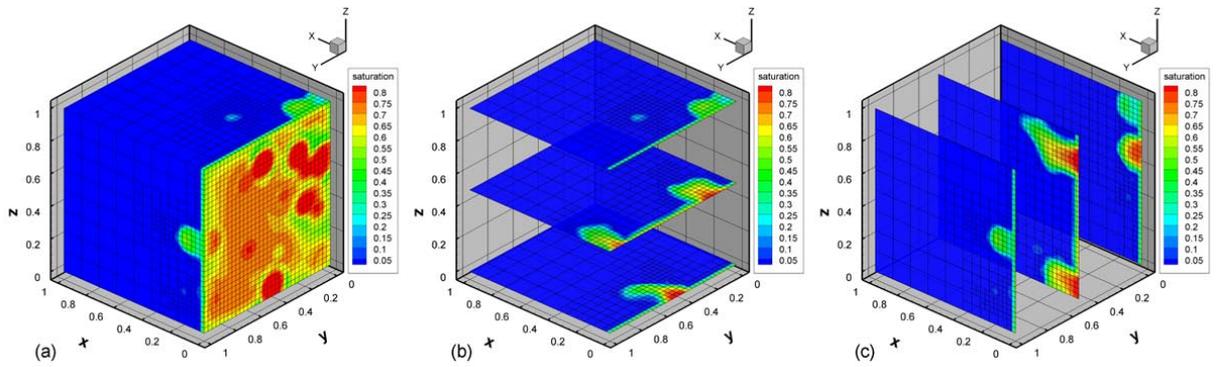


Fig. 6. Saturation distribution contours in a random porous medium (eq. (37), $N = 200$) with corresponding adaptive grids at $t = 0.75$ with 8,464 elements and 262,949 degrees of freedom: (a) 3D contour. (b) Surface contours of $X - Y$ plane at $z = 0.0, 0.5, 1.0$. (c) Surface contours of $X - Z$ planes at $y = 0.0, 0.5, 1.0$.

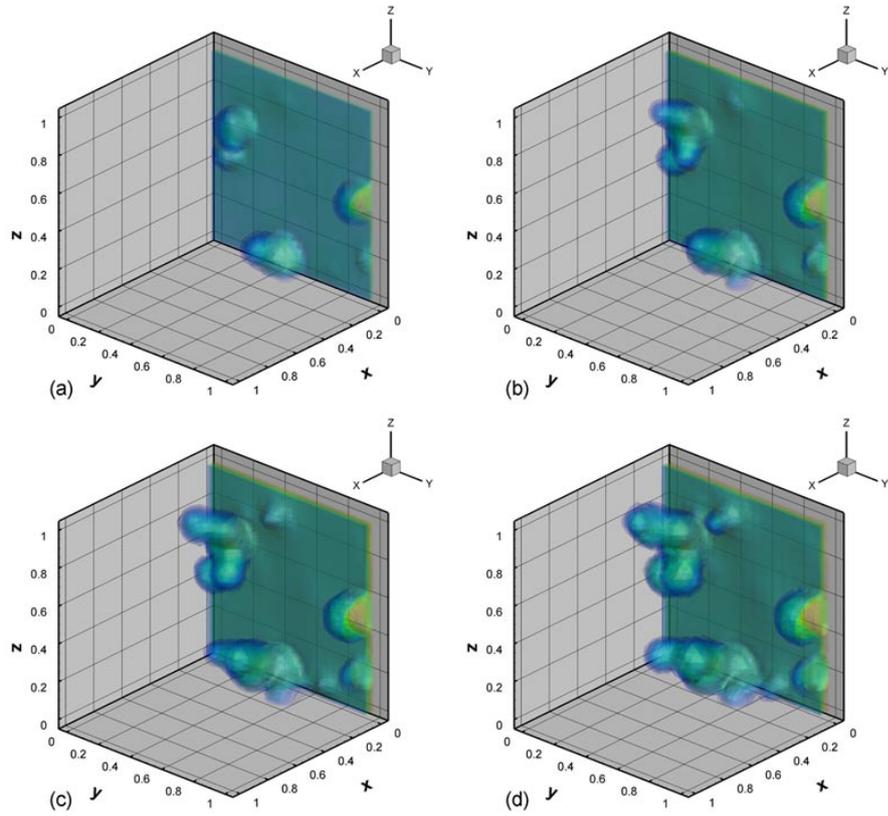


Fig. 7. Time evolution of the invading wetting phase front saturation into a random porous medium (eq. (37), $N = 50$) initially filled with non-wetting phase (a) $t = 0.64$. (b) $t = 1.28$. (c) $t = 1.92$. (d) $t = 2.59$.

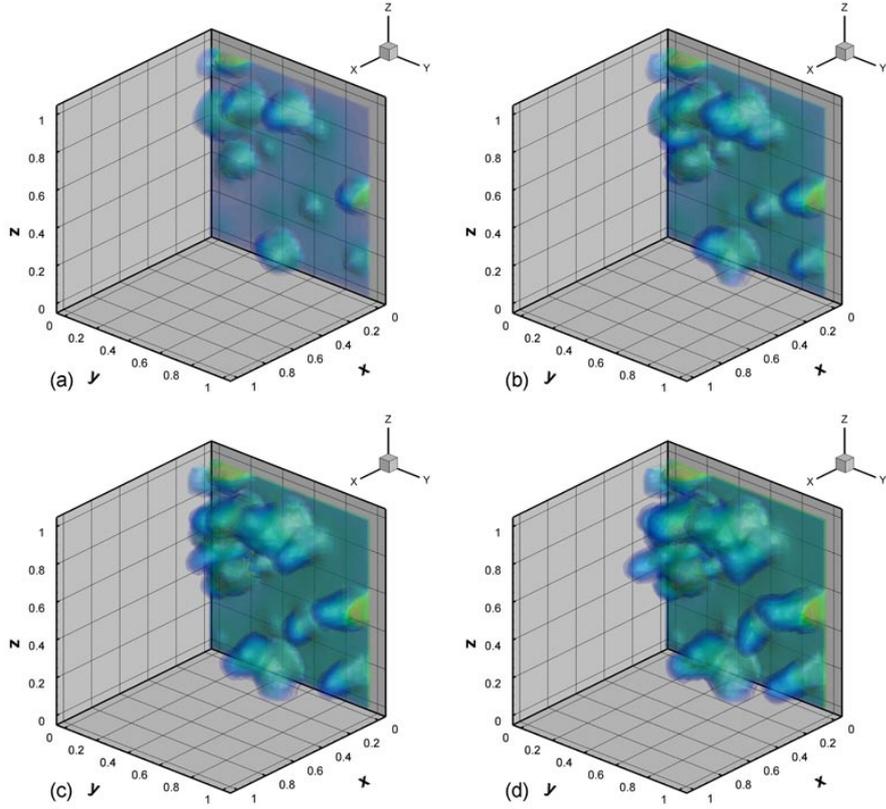


Fig. 8. Time evolution of the invading wetting phase front saturation into a random porous medium (eq. (37), $N = 200$) initially filled with non-wetting phase (a) $t = 0.19$. (b) $t = 0.38$. (c) $t = 0.57$. (d) $t = 0.74$.

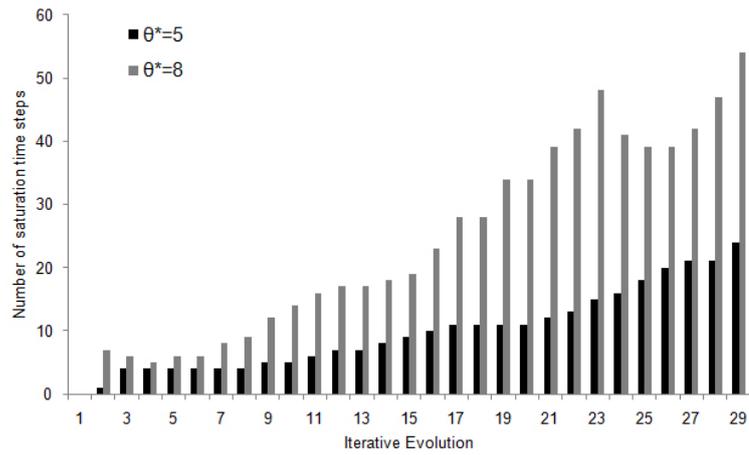


Fig. 9. Numbers of saturation time steps between two solutions of the velocity-pressure system (i.e. between n_p and n_{pp}) for two different thresholds ($\theta^* = 5$ and $\theta^* = 8$) in criterion (14).

Appendix C

Saturation and Permeability Gradient Effects in Two-phase Flows in Heterogeneous Porous Media

Saturation and Permeability Gradient Effects in Two-phase Flows in Heterogeneous Porous Media

C.C. Chueh ^a, N. Djilali ^{a,*}, W. Bangerth ^b

^a*Institute for Integrated Energy Systems and Department of Mechanical Engineering,
University of Victoria, PO Box 3055 STN CSC, Victoria, BC, Canada, V8W 3P6*

^b*Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA*

Abstract

In this work, there are two major aspects that we are trying to improve for numerical simulation of two-phase flow in heterogeneous porous media: one for the numerical part and the other for the modeling part. For the numerical part, when the mixture model of two-phase flow in porous media is used in places where there is a small amount of capillary transport, the saturation transport reduces to an advection-dominated problem, with saturation discontinuities, which is difficult to solve numerically. Without appropriate numerical methods, it is hard for scientists or engineers in the field of two-phase flow in heterogeneous porous media to better understand the formation mechanisms of complex transport phenomena. For the modeling part, there is no paper dealing with the effect of the permeability gradient on capillary transport; however, this factor is critical in order to provide realistic predictions for two-phase fluid flow in porous media. Based on these two important issues, we implement a combination of the effects of the saturation gradient and permeability gradient on capillary pressure while also considering the immobile or irreducible saturation by using state-of-the-art numerical methods. The numerical methods (the stabilized finite element method, adaptive operator splitting, adaptive mesh refinement and block matrix preconditioning) we use here maintain accuracy and stability, resolve fine-scale flow behaviors in heterogeneous porous media and reduce the computational cost. Along with

our numerical predictions, for the first time, we discuss the complex two-phase interactions in heterogeneous porous media as well as the effects of different capillary intensities. In addition, physical explanations are given for the formation of complex transport phenomena.

Key words: Adaptive mesh refinement; Stabilized finite element method; Operator splitting; Preconditioning; Two-phase flow; Heterogeneous porous media; Fuel cells

1 Introduction

There are several natural systems and industrial applications that involve the two-phase flow in heterogeneous structures of porous media. One of the examples on which current interest is primarily focused is polymer electrolyte membrane (PEM) fuel cells in which effective water management is vital for the successful operation, performance and durability. However, achieving an optimal level of water requires a delicate balance in PEM fuel cells. Water produced from the electrochemical reactions combined with water from humidified inlet gases contribute towards necessary membrane hydration. However, the accumulation of excess liquid water in the gas diffusion layer (GDL) and gas flow channels leads to oxidant starvation and performance loss [1]. In addition to these mass transport limitations, excess liquid water can also lead to non-homogeneous current density [2], ineffective heat removal, and membrane swelling [3]. Therefore, liquid water within the fuel cells must be managed well, which can be facilitated by the simulation of realistic, complex two-phase flow process in heterogeneous porous media.

Simulations can help scientists or engineers give a prognosis of whether or not a porous medium with its own material properties has the capabilities in making excess liquid water

* Corresponding author.

Email address: ndjilali@uvic.ca (N. Djilali).

flow out of it more efficiently and then, based on the results obtained from the simulations, provide useful suggestions on manufacturing process of the porous media of PEM fuel cells. As we will show below, one possible way to achieve this is to simulate two-phase flow in a porous medium with a certain permeability gradient effect generating additional capillary forces to make excess water exit the system. Despite the obvious advantage, in the existing literature, to the best of our knowledge no one has already dealt with this issue that may be crucial because, for the most part, a heterogeneous porous medium considered is constituted by a few different patches with different constant permeability values (i.e. $\nabla k = 0$).

Based on what has been mentioned above, the current study is motivated by the need to understand the effects of saturation and permeability gradients on capillary transport in heterogeneous porous media. The objective of the present work is to develop a transient, adaptive model to help predict and understand the physical formation of two-phase complex transport phenomena in *heterogeneous* porous media by using a state-of-the-art numerical combination of adaptive mesh refinement (AMR), adaptive operator splitting, block preconditioning and a stabilization approach, which numerical combination has been developed for an advection-dominated two-phase flow problem in our previous papers [4,5].

This paper is structured as follows: Section 2 presents the governing equations, boundary conditions and numerical method. Section 5 presents insights from the results about the dependence of capillary pressure on saturation and permeability. Finally, the paper is concluded in Section 6.

2 Mixture Model of Two-phase Flow in Heterogeneous Porous Media

In the following, let us first derive the governing equations of the mixture model used here, taking into account in particular the heterogeneity of the rock matrix. We will also provide physical and mathematical interpretations; define relevant physical parameters, coefficients, boundary and initial conditions; introduce an absolute permeability model. All of the involved parameters are listed in Table 1.

2.1 Mixture model formulation

Neglecting gravitational effects, a two-phase immiscible, incompressible, isothermal fluid can be described through mass balance equations for wetting and nonwetting phases:

$$\epsilon \frac{\partial (\rho_w S_w)}{\partial t} + \nabla \cdot (\rho_w \mathbf{u}_w) = \rho_w q_w, \quad (1)$$

$$\epsilon \frac{\partial (\rho_{nw} S_{nw})}{\partial t} + \nabla \cdot (\rho_{nw} \mathbf{u}_{nw}) = \rho_{nw} q_{nw}. \quad (2)$$

Here, S is the saturation (volume fraction) indicating the composition of the mixture of fluids and ρ is the density, ϵ is the porosity, q is the source term, \mathbf{u} is the velocity. Subscripts w and nw denote the wetting and non-wetting phases, respectively. We will assume incompressibility, i.e., ρ_w, ρ_{nw} are both constant. We model the velocities \mathbf{u}_w and \mathbf{u}_{nw} by Darcy's Law,

$$\mathbf{u}_w = -\frac{k_{rw}}{\mu_w} \mathbf{K} \nabla p_w = -\lambda_w \mathbf{K} \nabla p_w, \quad (3)$$

$$\mathbf{u}_{nw} = -\frac{k_{rnw}}{\mu_{nw}} \mathbf{K} \nabla p_{nw} = -\lambda_{nw} \mathbf{K} \nabla p_{nw}, \quad (4)$$

where λ is the fluid mobility, k_r is the relative permeability, μ is the viscosity and p is the pressure. We assume that the pore space is completely filled by the mixture, yielding the closure relation $S_w + S_{nw} = 1$. \mathbf{K} is the permeability tensor and we will in the following

assume isotropic permeabilities of the form $\mathbf{K} = k\mathbf{I}$. Furthermore, the two pressures are related by the capillary pressure, $p_c = p_{nw} - p_w$.

Our goal is to derive a single equation for the saturation by eliminating $S_{nw} = 1 - S_w$. Since we then only deal with a single saturation variable, S_w , we will henceforth drop the index to simplify notation and refer to it by S . Then, let us define the total mobility λ_t and fractional flow F of the wetting phase by

$$\lambda_t(S) = \lambda_w(S) + \lambda_{nw}(S) = \frac{k_{rw}(S)}{\mu_w} + \frac{k_{rnw}(S)}{\mu_{nw}}, \quad (5)$$

$$F(S) = \frac{\lambda_w(S)}{\lambda_t(S)} = \frac{\lambda_w(S)}{\lambda_w(S) + \lambda_{nw}(S)} = \frac{k_{rw}(S)/\mu_w}{k_{rw}(S)/\mu_w + k_{rnw}(S)/\mu_{nw}}. \quad (6)$$

Combining the equations above then yields (see also [6])

$$\mathbf{u}_t = -\lambda_t(S)\mathbf{K}\nabla p_w - \lambda_{nw}(S)\mathbf{K}\nabla p_c, \quad (7)$$

$$\nabla \cdot \mathbf{u}_t = q_w + q_{nw}, \quad (8)$$

$$\epsilon \frac{\partial S}{\partial t} + \nabla \cdot (F(S)\mathbf{u}_t) + \nabla \cdot (\lambda_{nw}(S)F(S)\mathbf{K}\nabla p_c) = q_w. \quad (9)$$

There remains the task of modeling the capillary pressure, p_c . We follow a suggestion by Leverett [7,8] to use the so-called Leverett J -function and set

$$p_c = \frac{\sigma_c \cos(\theta_c)}{(k/\epsilon)^{1/2}} J(S_e), \quad (10)$$

where σ_c is the surface tension, k is the permeability and θ_c is the contact angle which varies between 0° and 90° for the hydrophilic case and between 90° and 180° for hydrophobic case. S_e is the effective saturation [9–11], defined as

$$S_e = \frac{S - S_{\text{irr}}}{1 - S_{\text{irr}}} \quad (11)$$

where S_{irr} is the so-called immobile or irreducible saturation [9,12], representing the amount of entrapped liquid fluid. It can be regarded as a threshold point below which the liquid remains immobile. The immobile or irreducible saturation depends on the pore

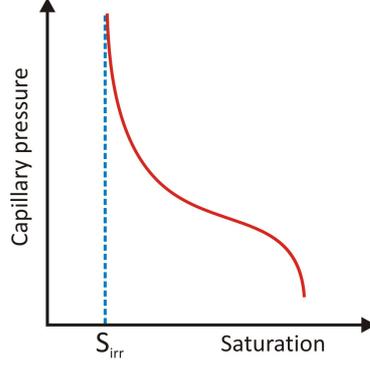


Fig. 1. General form of the capillary pressure curve of hydrophilic porous media.

structure of the medium [12] and the ratio of viscous to surface tension forces (often expressed as the capillary number $Ca = \frac{u_{inv}\mu}{\sigma_c}$ where u_{inv} is an invading phase velocity, see [13,10]). For hydrophilic porous media, the general form of the capillary pressure-saturation curve is shown in Figure 1 [9,10]. In this work, we use the J -function proposed in [11]:

$$J(S_e) = 1.417(1 - S_e) - 2.120(1 - S_e)^2 + 1.263(1 - S_e)^3 \quad (12)$$

Using this model for p_c , we can express equations (7)–(9) in their final form as

$$\mathbf{u}_t = -\lambda_t(S)\mathbf{K}\nabla p_w - \lambda_{nw}(S)\mathbf{K}\frac{\partial p_c}{\partial S}\nabla S - \lambda_{nw}(S)\mathbf{K}\frac{\partial p_c}{\partial k}\nabla k, \quad (13)$$

$$\nabla \cdot \mathbf{u}_t = q_w + q_{nw}, \quad (14)$$

$$\epsilon \frac{\partial S}{\partial t} + \nabla \cdot \left[F(S) \left(\mathbf{u}_t + \lambda_{nw}(S)\mathbf{K}\frac{\partial p_c}{\partial k}\nabla k \right) \right] + \nabla \cdot \left(\lambda_{nw}(S)F(S)\mathbf{K}\frac{\partial p_c}{\partial S}\nabla S \right) = q_w. \quad (15)$$

Note that $\frac{\partial p_c}{\partial S} < 0$, ensuring well-posedness of the last equation.

2.2 Physical and mathematical interpretation

Capillary forces act on the movement of both phases through the dependence of the total velocity \mathbf{u}_t on the capillary pressure in (13) and the dependence of the saturation S in equation (15). More specifically, however, we see from equations (1) and (3) that the

transport of the wetting phase is entirely determined by the wetting phase pressure p_w . In models without capillary pressure, p_w is determined by boundary values and the source term $q_w + q_{nw}$. However, in the presence of capillary forces, $p_{nw} = p_w + p_c$ takes on this role. Consequently, the forces driving the motion of the wetting fluid result from the difference $p_w = p_{nw} - p_c$ of externally applied forces and capillary pressure. The wetting fluid velocity is then given by

$$\mathbf{u}_w = -\lambda_w \mathbf{K} \nabla p_w = -\lambda_w \mathbf{K} \nabla p_{nw} + \lambda_w \mathbf{K} \left[\frac{\partial p_c}{\partial S} \nabla S + \frac{\partial p_c}{\partial k} \nabla k \right].$$

Since both partial derivatives of p_c are negative, this equation implies transport (i) from high to low pressure, (ii) from high saturation areas to low saturation areas, and (iii) from areas with high permeability into areas of low permeability. The second of these effects gives rise to the diffusion term in (15).

While block wise heterogenous permeabilities have been considered in the literature through interface conditions [14], we have been unable to find references that would describe differentially heterogenous media as we have done above. Our formulation generalizes that found elsewhere in that a discontinuous permeability k would lead to a delta-like contribution to the velocity field. An alternative viewpoint is that there is an entry pressure that has to be overcome before fluids can enter one medium from another. This is in accordance with, for example, the discussion in [14].

3 Numerical Methods

We use the finite element method to solve equations (13)–(15) and follow the general outline of the algorithms described and validated in [4] and [5]. In particular, we use (i) a stable choice of continuous elements for all variables [15,16], (ii) the nonlinear artificial viscosity method by Guermond and Pasquetti’s [17] to stabilize the transport equation,

(iii) adaptive mesh refinement to resolve regions with large gradients, (iv) block preconditioners to accelerate the solution of the velocity-pressure system, (v) an adaptive operator split method that only recomputes the pressure and velocity whenever necessary while solving for the saturation in every step.

Our implementation of these methods is based on the deal.II library [18,19] and uses parts of the step-21, step-31 and step-33 tutorial programs of this library [20–22].

4 Numerical Testcases

In the following, let us briefly describe the general setup of the testcases we will use in the numerical experiments shown in Section 5.

4.1 Relative permeabilities and other coefficients

To close the system of partial differential equations (13), (14), (15) we need to define the relative permeabilities. For the purpose of this paper, we choose them as follows [23]:

$$\begin{aligned} k_{rw}(S_e) &= S_e^3, \\ k_{rnw}(S_e) &= (1 - S_e)^3. \end{aligned}$$

The fractional flow of the wetting phase F in (6) then becomes

$$F = \frac{\frac{S_e^3}{\mu_w}}{\frac{S_e^3}{\mu_w} + \frac{(1-S_e)^3}{\mu_{nw}}} = \frac{1}{1 + \frac{(1-S_e)^3}{S_e^3} \frac{\mu_w}{\mu_{nw}}}.$$

All the other physical parameters used in our computations are shown in Table 1.

Table 1

Parameters used in the model

PARAMETER	SYMBOL	VALUE	UNITS
Porosity	ϵ	0.6	-
Viscosity (wetting)	μ_w	0.2	$g \cdot cm^{-1} \cdot sec^{-1}$
Viscosity (nonwetting)	μ_{nw}	1.0	$g \cdot cm^{-1} \cdot sec^{-1}$
Immobile or irreducible saturation	S_{irr}	0.0 or 0.18	-
Surface tension	σ_c	10^{-4} or 10^{-2}	$g \cdot sec^{-2}$
Contact angle	θ_c	72°	-
Number of high-permeability centers	N_p	40	-
Permeability magnitude scaling factor	γ_m	10^0	-
Bandwidth	B_w	0.05	-
Number of source/sink terms	N_s	1 or 4	-
The i -th wetting-phase source/sink intensity	$q_w^{(i)}$	2.0	sec^{-1}

4.2 Source terms

We here use the following form for the source terms q_w and q_{nw} in equations (14)–(15):

$$q_w = \sum_{i=1}^{N_s} q_w^{(i)} \chi_{A_s^{(i)}}(\mathbf{x}), \quad q_{nw} = 0,$$

where $q_w^{(i)}$ indicates the volume density of the fluid produced or injected per unit time at the i -th source/sink, N_s is the number of the sources/sinks and $\chi_{A_s^{(i)}}(\mathbf{x})$ is the indicator

function of the i -th sub-domain $A_s^{(i)}$, i.e. $\chi_A(\mathbf{x}) = 1$ if $\mathbf{x} \in A$ and $\chi_A(\mathbf{x}) = 0$ otherwise.

In our numerical examples, we use $q_w^{(i)} = 2$ (1/sec), $N_s = 1$ or $N_s = 4$, and each set $A_s^{(i)}$ equal to circles of radius 0.125 cm centered at $\mathbf{x}_c^{(i)}$. If $N = 1$, then $\mathbf{x}_c^{(1)} = (0.5, 0.5)$; for $N_s = 4$, we use $\mathbf{x}_c^{(1)} = (0.25, 0.25)$, $\mathbf{x}_c^{(2)} = (0.75, 0.25)$, $\mathbf{x}_c^{(3)} = (0.25, 0.75)$, and $\mathbf{x}_c^{(4)} = (0.75, 0.75)$.

4.3 Absolute permeability model

In many fields, realistic media are often modeled by generating random permeabilities [24]. Following [20], we choose an isotropic permeability $\mathbf{K} = k\mathbf{I}$, where

$$k(\mathbf{x}) = \gamma_m \min \left\{ \max \left\{ \sum_{l=1}^{N_p} \Psi_l(\mathbf{x}), 0.01 \right\}, 4 \right\} \quad (16)$$

with

$$\Psi_l(\mathbf{x}) = \exp \left(- \left(\frac{|\mathbf{x} - \mathbf{x}_l|}{B_w} \right)^2 \right). \quad (17)$$

Here, γ_m is a magnitude scaling factor. The centers \mathbf{x}_l are N_p randomly chosen locations inside the domain and B_w is the bandwidth of the exponential function. This function models a domain in which there are N_p regions of higher permeability embedded in a matrix of more pristine, unperturbed background rock. The permeability function is bounded both above and below to limit the size of the condition number. Figure 2 shows this permeability field with $N_p = 40$.

4.4 Computational domain, initial and boundary conditions

We perform simulations on the domain $\Omega = [0, 1]^2 \subset \mathbb{R}^2$ for $t \in [0, T]$. The initial conditions are $S_w(\mathbf{x}, 0) = 0$, i.e. the reservoir is filled by the non-wetting phase. The boundary conditions for the velocity/pressure equations (13) and (14) are given on all the boundaries

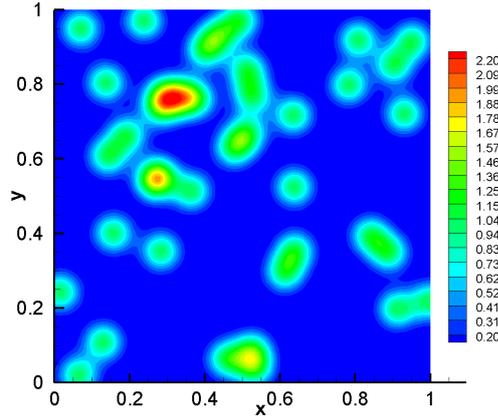


Fig. 2. Permeability field used in the present work

with a fixed gauge pressure of the wetting phase as

$$p_w(\mathbf{x}, t) = 0 \quad \text{on} \quad \partial\Omega \quad (18)$$

The boundary conditions for the saturation transport equation (9) are only imposed on inflow boundaries:

$$S(\mathbf{x}, t) = 0 \quad \text{on} \quad \Gamma_{in}(t) = \{\mathbf{x} \in \partial\Omega : \mathbf{n} \cdot \mathbf{u}_t < 0\}. \quad (19)$$

5 Numerical Results and Discussion

At the beginning, we first show the numerical validation of an advection-dominated two-phase flow in *uniform* porous media. Then, we discuss and compare five different effects in the following order: the effect of the dependence of capillary pressure on S with $S_{\text{irr}} = 0$, the effect of the dependence of capillary pressure on S with $S_{\text{irr}} = 0.18$, the effect of the dependence of capillary pressure on S and k with $S_{\text{irr}} = 0$, the effect of the dependence of capillary pressure on S and k with $S_{\text{irr}} = 0.18$, and the effect of different capillary intensities. It is noted that, since we employ more than one value for three parameters: irreducible saturation S_{irr} , surface tension σ_c , the number of source/sinks N_s , these three

parameters are indicated at the beginning of sections 5.2–5.5. All other parameters are shown in Table 1.

5.1 Numerical validation of an advection-dominated two-phase flow in uniform media

Table 2

Parameters used in the Buckley-Leverett problem

PARAMETER	SYMBOL	VALUE	UNIT
Porosity	ϵ	1.0	-
Viscosity (wetting)	μ_w	0.2	$g \cdot cm^{-1} \cdot sec^{-1}$
Viscosity (nonwetting)	μ_{nw}	1.0	$g \cdot cm^{-1} \cdot sec^{-1}$
permeability (constant)	k	0.1	cm^2
Immobile or irreducible saturation	S_{irr}	0.0	-

Here, for numerical validation, we compare the present results obtained with the adaptive grid method to previous work using uniform grids in which Raviart-Thomas finite elements are adopted for velocity and discontinuous Galerkin (DG) elements for pressure and saturation [20]. The parameters used in this validation are shown in Table 2. Initial conditions are only required for the saturation variable, and we choose $S(\mathbf{x}, 0) = 0$, i.e. the porous medium is initially filled by the non-wetting phase. We prescribe a linear pressure on the boundaries:

$$p(\mathbf{x}, t) = 1 - x \quad \text{on} \quad \partial\Omega \times [0, T]. \quad (20)$$

Pressure and saturation uniquely determine a velocity, and the velocity determines whether a boundary segment is an inflow or outflow boundary. On the inflow part of the boundary,

$\Gamma_{in}(t)$, we impose

$$S(\mathbf{x}, t) = 1 \quad \text{on} \quad \Gamma_{in}(t) \cap \{x = 0\}, \quad (21)$$

$$S(\mathbf{x}, t) = 0 \quad \text{on} \quad \Gamma_{in}(t) \setminus \{x = 0\}. \quad (22)$$

In other words, the domain is flooded by the wetting phase from the left. No boundary conditions for the saturation are required for the outflow parts of the boundary.

Following a commonly used prescription for the dependence of the relative permeabilities k_{rw} and k_{rnw} on saturation, we use

$$k_{rw} = S_e^2, \quad k_{rnw} = (1 - S_e)^2. \quad (23)$$

As shown in Figure 3, results obtained from the simulation for the Buckley-Leverett problem are essentially identical to those in the work of Li and Bangerth [20]. Since the Buckley-Leverett problem has been recognized as one of the shock-type problems, both simulations have a saturation shock or discontinuity around $x = 0.58$. This agreement shows that the diffusive stabilization, proposed by Guermond and Pasquetti [17], has the powerful ability to efficiently damp unphysical oscillations while ensuring the accuracy of the solutions with no upwinding. For more validation about this diffusive scheme, the reader is referred to previous works [4,5].

5.2 *Effect of the dependence of capillary pressure on S only with $S_{irr} = 0$*

For this case, $N_s = 1$ as described in section 4.1. The immobile saturation S_{irr} is set to zero, the surface tension σ_c is set to $10^{-2} g \cdot sec^{-2}$ and the one permeability gradient term appearing in both equations (13) or (15) is not considered. Thus, there is no permeability gradient effect but only a saturation gradient effect on capillary transport.

Figure 4 shows the time evolution of wetting-phase saturation contours with their cor-

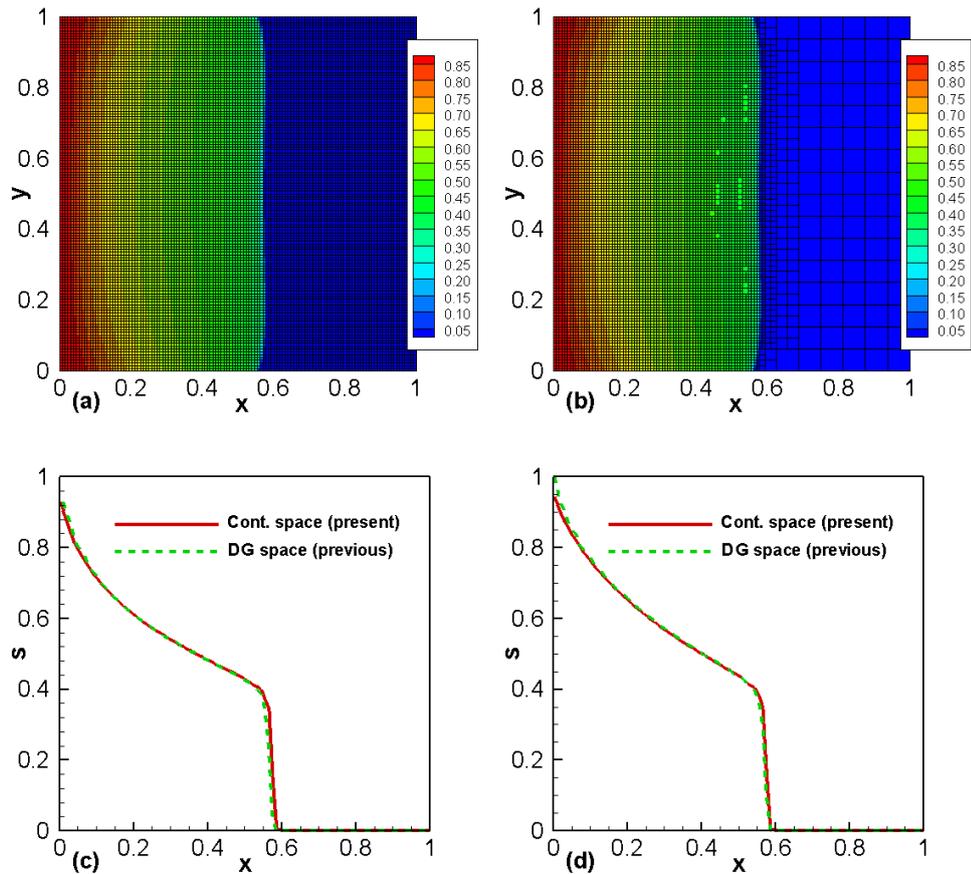


Fig. 3. Numerical results for uniform media in comparison with previous work of Li and Bangerth [20] at $t = 2.78 \text{ sec}$ when the pressure-velocity system is solved adaptively saturation time steps (adaptive operator splitting: $\theta^* = 1.0$). (a) Saturation field using discontinuous Galerkin space (previous work). (b) Saturation field using continuous space with the stabilizing term and operator splitting (present work). (c) Saturation profile along $y = 0.5$. (d) Saturation profile along $y = 0.9$.

responding adaptive refinement meshes. The wetting-phase fluid is injected into a hydrophilic sample initially filled with the non-wetting phase fluid. There is a distorted maple leaf shaped high-saturation formation which appears in Figure 4b. As time passes, the formation gets bigger gradually and eventually becomes a calabash-like formation in Figure 4c. In addition, it is clearly seen that our adaptive grids change in response to transient two-phase transport phenomena. In fact, we refine only the areas where wetting-phase

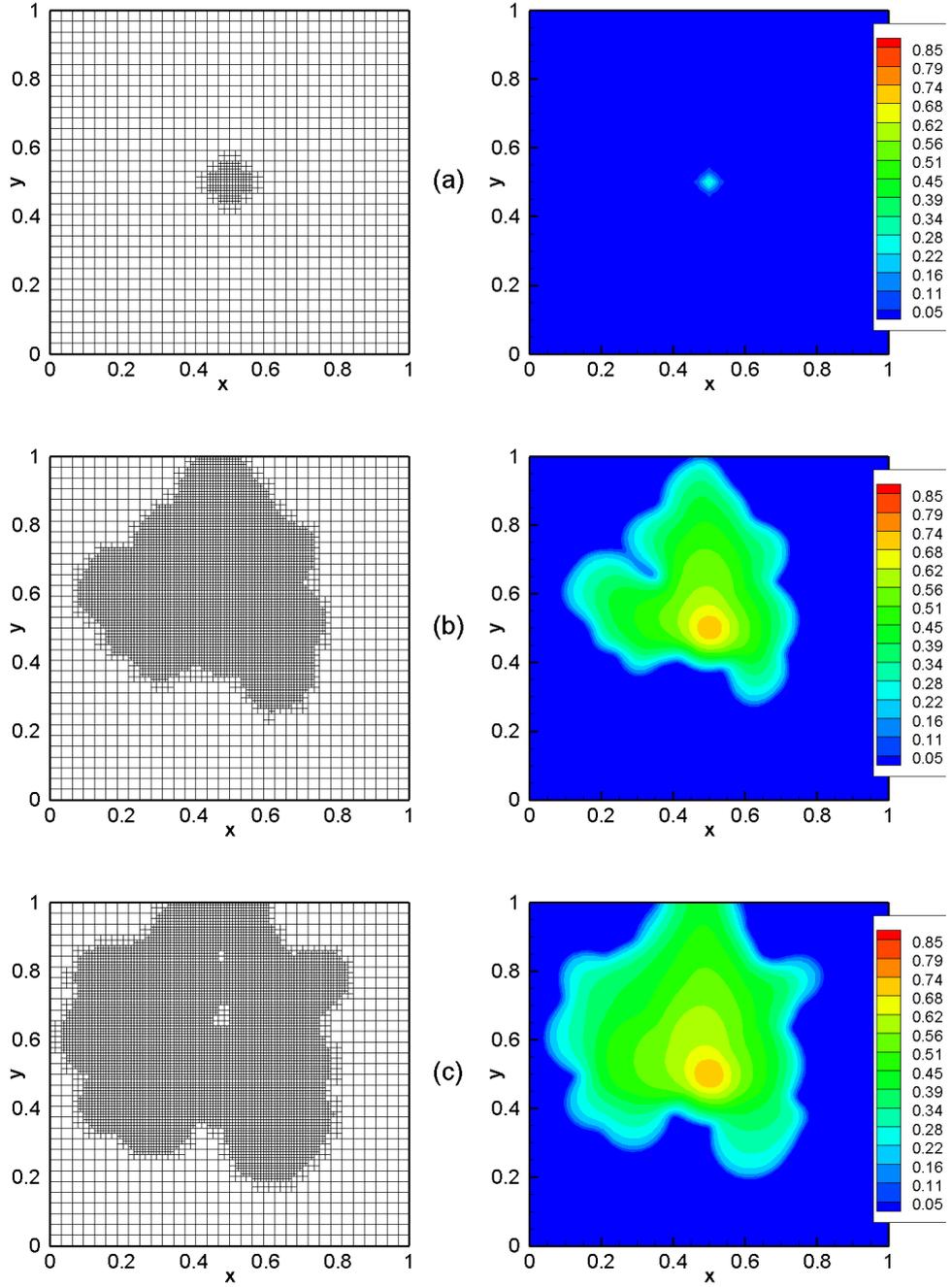


Fig. 4. Time evolution of wetting-phase saturation contours (right) with their corresponding grids (left) for capillary pressure ($N_s = 1$, $\sigma_c = 10^{-2} g \cdot sec^{-2}$ and $S_{irr} = 0$) dependent on S only at (a) $t = 0.016 sec$ with 1240 cells and 13068 DoFs (b) $t = 10.189 sec$ with 6166 cells and 63480 DoFs (c) $t = 19.095 sec$ with 8380 cells and 85994 DoFs

saturation solutions are not continuous, rather than globally refining the entire computational domain. The number of degrees of freedom (DoFs) is greatly reduced, resulting in an increase in computational efficiency.

Figure 5 is used to explain the mechanism of complex interactions between two gradients: the wetting-phase pressure and the wetting-phase saturation (no permeability gradient effects considered in the present case).

Consider the two fluids together (equations (13) and (14)). The pressure forces push the two-phase fluid forward (from high to low pressure) while the saturation gradient effect forces it backward (from low to high saturation). The pressure gradient forces (Figure 5c) overwhelm the saturation gradient effect (Figure 5d) in three directions: approximately 90° East of North, North and 90° West of North from the wetting-phase source's position (0.5, 0.5). This makes the two-phase fluid move in the three ways. It is noted, though, that the quantities of the pressure gradients (Figure 5c) do not appear to be large enough to overcome the saturation gradients (Figure 5d). The reason for this counter-intuition is that in equation (13), the pressure gradient term is multiplied by the total mobility λ_t but the saturation gradient term is multiplied by a scaling factor $(\partial p_c / \partial S) \cdot \lambda_{nw}$. Therefore, based on the fact, i.e. $\lambda_t > (\partial p_c / \partial S) \cdot \lambda_{nw}$, pressure gradient effect is stronger than saturation gradient effect in the three directions.

Consider the wetting-phase fluid alone (equation (15)). As mentioned in a previous subsection, the wetting-phase fluid advances using two kinds of movement: convective and diffusive transport. The convective part is determined by the total velocity and the diffusive term by the capillary pressure. Those two kinds of transport are restricted by porous heterogeneities. The wetting-phase fluid flows only through high-permeability regions: Figure 5d and Figure 5e together show that the wetting-phase fluid goes mainly in three directions in which the high-permeability regions take place.

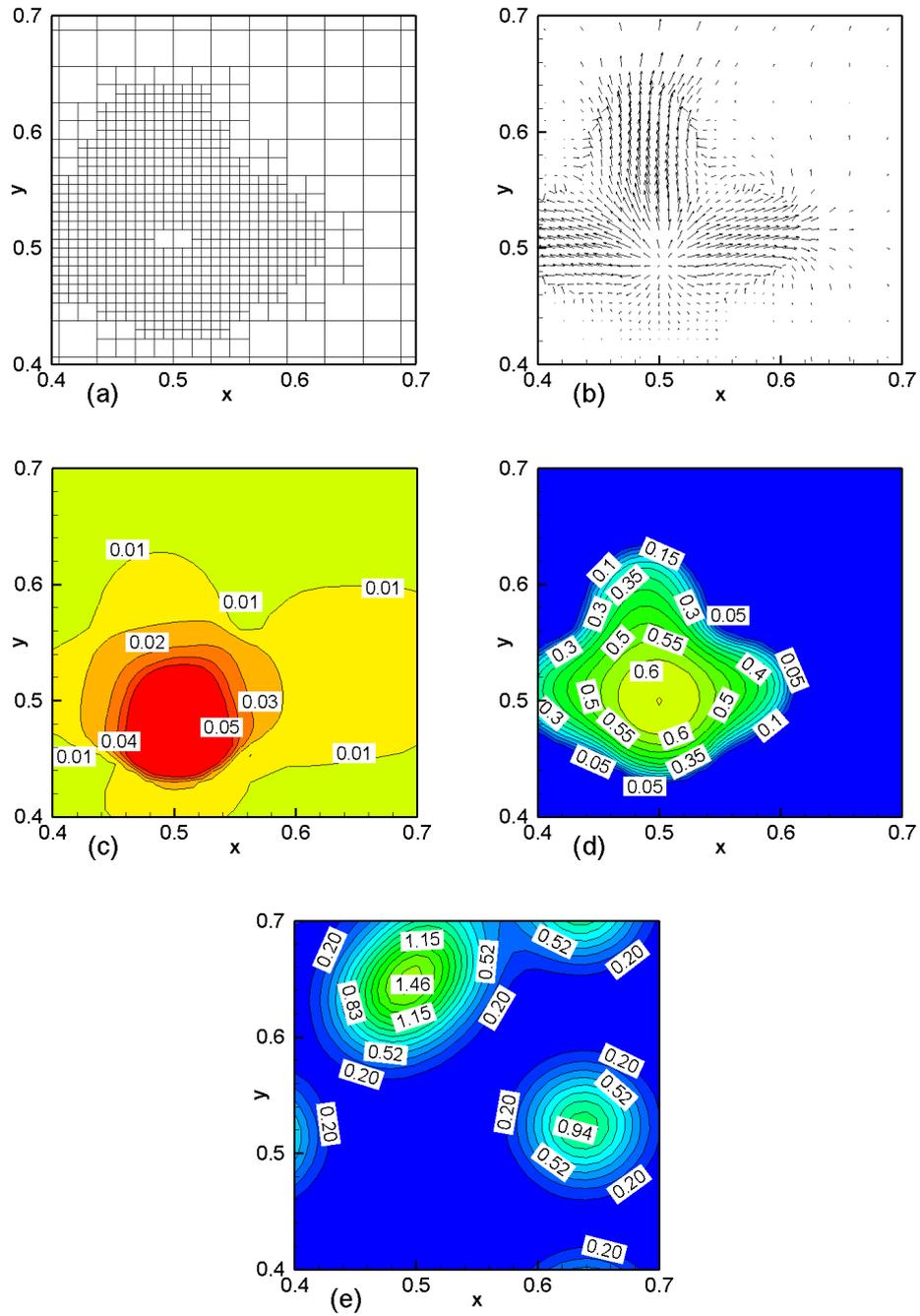


Fig. 5. Capillary pressure dependent on S only with $N_s = 1$, $\sigma_c = 10^{-2} \text{ g} \cdot \text{sec}^{-2}$ and $S_{\text{irr}} = 0$ at $t = 1.038 \text{ sec}$ on the sub-domain $[x, y] \in [0.4, 0.7] \times [0.4, 0.7]$ (a) adaptive mesh (b) total velocity vector field (c) wetting-phase pressure contour (d) wetting-phase saturation contour (e) permeability field

In addition, it is clearly seen in Figure 5b that two vortices appear around $(0.6, 0.55)$ and $(0.6, 0.45)$ on the sub-domain $[x, y] \in [0.4, 0.7] \times [0.4, 0.7]$. In fact, these vortex phenomena are caused by a combination of the heterogeneity of porous media and the complex interaction between the wetting-phase pressure and saturation gradients. The heterogeneity of porous media directs the two-phase fluid together to flow mainly through high permeability areas (Figure 5e). The wetting-phase pressure gradient effect pushes the two-phase fluid to move together forward (from high to low pressure) while, due to capillary effect, the wetting-phase saturation gradient effect pushes the two-phase fluid to move together backward (from low to high saturation). For instance, in the wetting-phase pressure contour (Figure 5c), the pressure gradient effect around $(0.6, 0.5)$ is stronger than those near its top and bottom and therefore forces caused by pressure gradient effect push the two-phase fluid to move to the right side. However, in the saturation contour (Figure 5d), capillary forces push the two-phase fluid near $(0.55, 0.55)$ from the northeast to southwest, and near $(0.55, 0.45)$ from the southeast to northwest. Thus, after a combination of pressure and capillary forces considered together in the porous media, the two vortices form in the porous domain.

5.3 Effect of the dependence of capillary pressure on S and k both with $S_{irr} = 0$

For this case, $N_s = 1$. The immobile saturation S_{irr} is set to zero, the surface tension σ_c is set to $10^{-2} g \cdot sec^{-2}$, and the permeability gradient effect appearing in both equations (13) and (15) is considered.

Consider the wetting-phase transport alone. In this case, capillary diffusive transport consists of two mechanisms: saturation and permeability gradients. As mentioned several times before, saturation and permeability gradients push the wetting-phase fluid to advance from high to low values of S and k . When considering the permeability gradient,

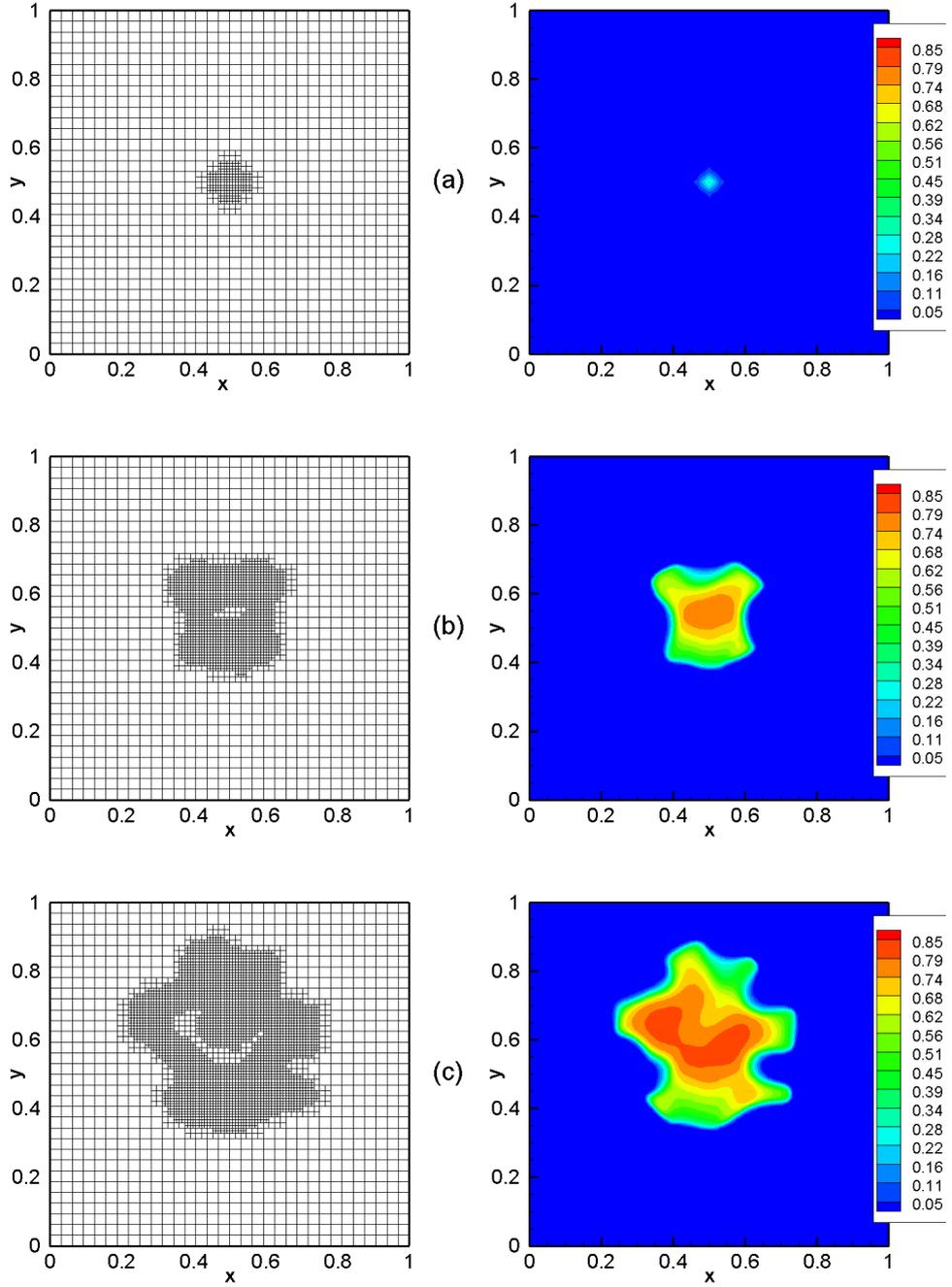


Fig. 6. Time evolution of wetting-phase saturation contours (right) with their corresponding grids (left) for capillary pressure ($N_s = 1$, $\sigma_c = 10^{-2} g \cdot sec^{-2}$ and $S_{irr} = 0$) dependent on both S and k at (a) $t = 0.016 sec$ with 1240 cells and 13068 DoFs (b) $t = 3.079 sec$ with 2437 cells and 25538 DoFs (c) $t = 5.882 sec$ with 4261 cells and 44518 DoFs

there is a special phenomenon which doesn't happen in any previous case: isolation of the wetting-phase fluid in heterogeneous porous media. In Figure 6, the wetting-phase fluid surrounded by three high-permeability areas on the top, left and right sides of $(0.5, 0.5)$ passes through a throat-like passageway around $(0.45, 0.55)$ at the beginning and then reaches a triangle-shaped area around $(0.39, 0.65)$ surrounded by few high-permeability regions. After that, the wetting-phase fluid accumulates inside this triangular area. In fact, during the process, the wetting-phase fluid doesn't flow through high-permeability regions but rather through low-permeability areas (Figure 2 and 6c), which appears counter-intuitive. The reason for this phenomenon is that the saturation gradient effect directs the fluid to flow in three directions (i.e. top, left and right directions) while the permeability gradient effect pushes it to flow in the opposite direction: as a result, those two effects counter-balance each other (i.e. the fluid is hard to move forward via capillary diffusion). Consequently, even though areas have lower permeability values, the wetting-phase fluid is forced by a combination of pressure, saturation and permeability gradient effects to flow through them.

Consider the two-phase fluid transport together. Its total velocity consists of three gradients: wetting-phase pressure, saturation and permeability gradients. Again, the wetting-phase pressure gradient directs the two-phase fluid to flow from high to low values while saturation and permeability gradients push it to advance from low to high values of S and k . Complex interactions among the three gradients is explained in Figure 7. In the areas of $\nabla S \approx 0$ (Figure 7d), the two-phase fluid is mainly governed by pressure and permeability gradient effects. For example, on the one hand, one high-permeability effect near $(0.64, 0.52)$ attracts the surrounding two-phase fluid to flow towards its highest permeability center $(0.64, 0.52)$. On the other hand, the pressure accumulates or gradually increases inside that circle-shaped region during the process caused by the permeability gradient effect. In other words, the accumulating pressure creates an opposing force to

keep the fluid from further moving towards the high-permeability area. According to the fact, each of the high-permeability areas surrounded by low-permeability regions has the high-pressure effect as well, as shown in Figure 7c. Furthermore, the velocity vector field is complicated by a combination of permeability and pressure gradients (Figure 7b) in those areas.

5.4 *Effect of the dependence of capillary pressure on S and k both with $S_{irr} = 0.18$*

The parameters for this case are the same as those in section 5.3 except for one parameter, S_{irr} , which is set to a value of 0.18 for comparison. Below the threshold of S_{irr} , the fractional flow F in equation (15) and capillary pressure p_c in equations (13) and (15) are both set to zero, representing no advective flow and capillary transport when the liquid fluid remains immobile.

Compare Figures 7 and 8 at the same time $t = 1.038$ (*sec*). Figure 8a demonstrates a maple leaf formation of adaptive grids while Figure 7a displays a circle formation. This implies that in the case of $S_{irr} = 0.0$ more wetting-phase fluid flows to the areas where saturation gradients are relatively large compared to the case of $S_{irr} = 0.18$. With the influence of the irreducible saturation threshold, Figure 8b shows the discontinuities in the total velocity vector field along the iso- S_{irr} line as well as vortex transport phenomena, as elucidated in sections. In Figure 8c, there is little wetting-phase pressure gradient around which $\nabla S \approx 0$, but there are three small spots representing the low pressure regions. These three small low-pressure areas are caused by a fact that there are interfaces between low and high permeability values. Finally, the saturation contour (Figure 8d) shows a football-like formation. Below the irreducible threshold (no capillary transport), the flow behaviour is the same as that in Figure 8d.

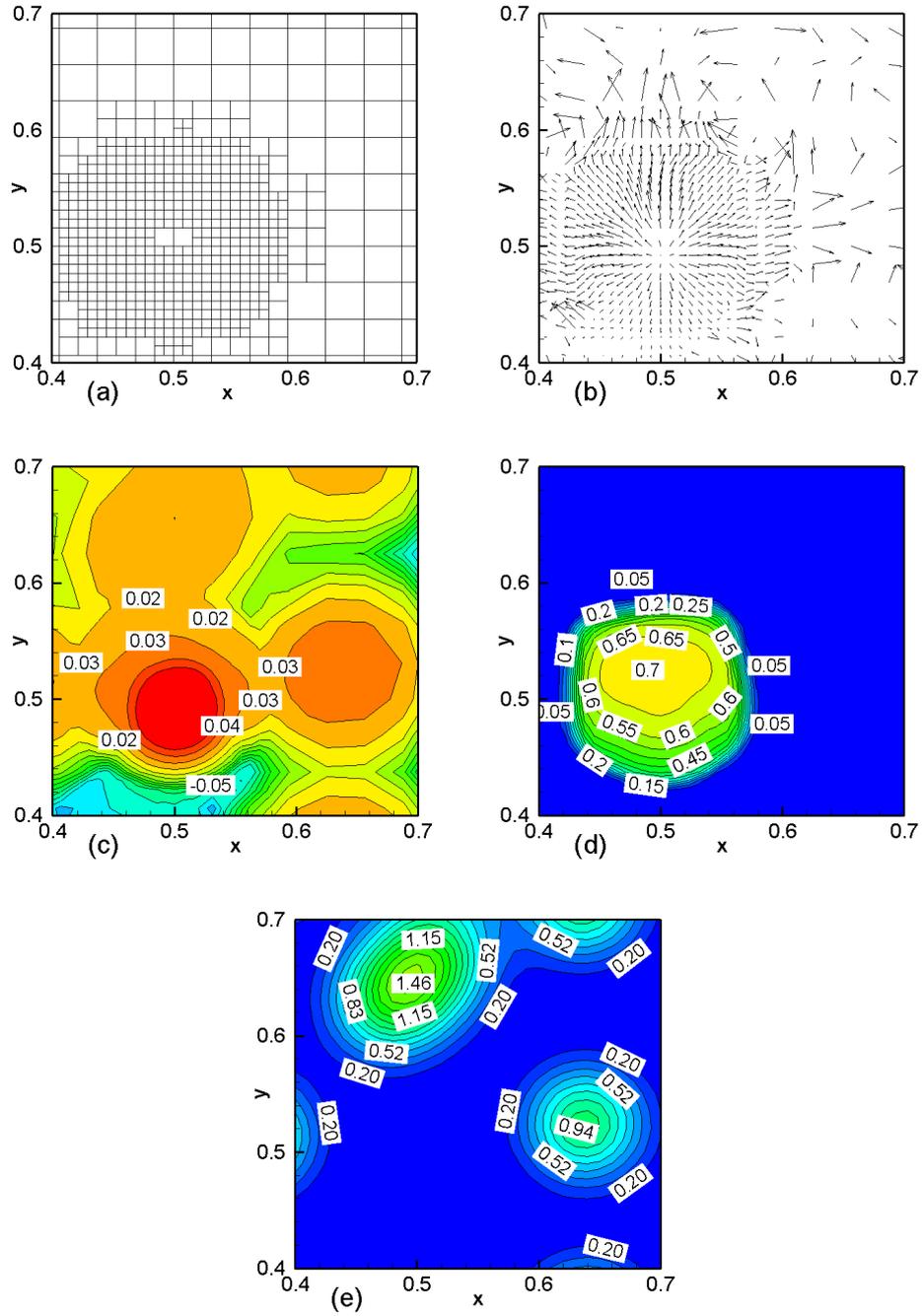


Fig. 7. Capillary pressure dependent on both S and k with $N_s = 1$, $\sigma_c = 10^{-2} g \cdot sec^{-2}$ and $S_{irr} = 0.0$ at $t = 1.038 sec$ on the sub-domain $[x, y] \in [0.4, 0.7] \times [0.4, 0.7]$ (a) adaptive mesh (b) total velocity vector field (c) wetting-phase pressure contour (d) wetting-phase saturation contour (e) permeability field

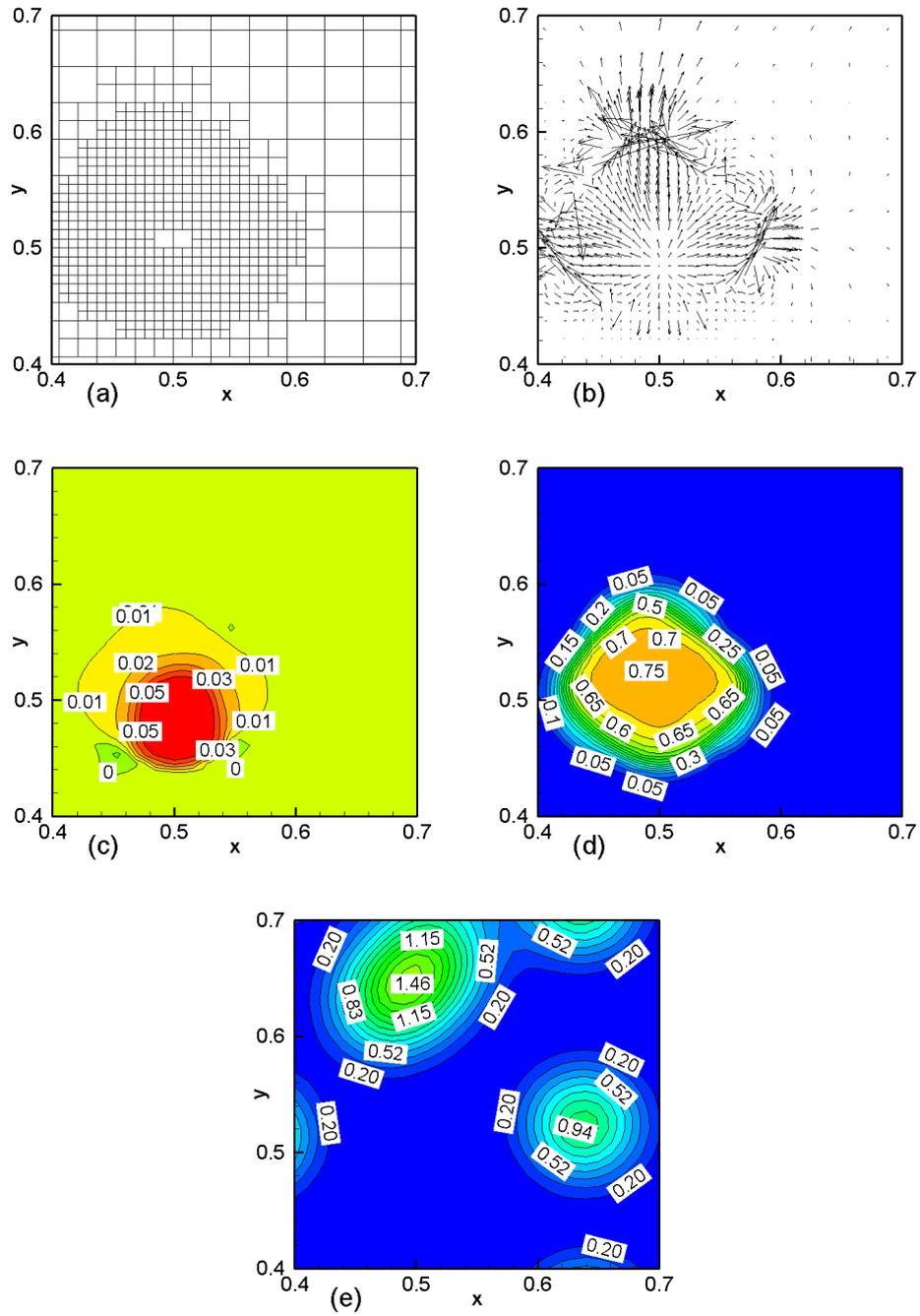


Fig. 8. Capillary pressure dependent on both S and k with $N_s = 1$, $\sigma_c = 0.2 \text{ g} \cdot \text{sec}^{-2}$ and $S_{\text{irr}} = 0.18$ at $t = 1.038 \text{ sec}$ on the sub-domain $[x, y] \in [0.4, 0.7] \times [0.4, 0.7]$ (a) adaptive mesh (b) total velocity vector field (c) wetting-phase pressure contour (d) wetting-phase saturation contour (e) permeability field

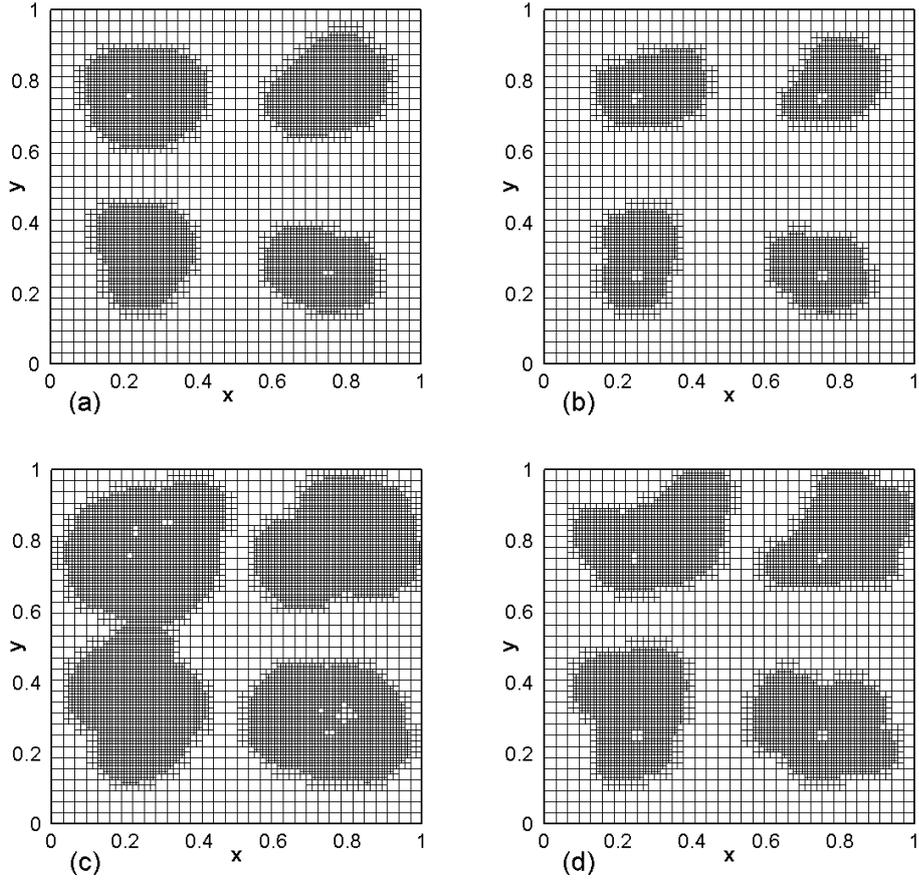


Fig. 9. Comparison of adaptive grids ($N_s = 4$ and $S_{\text{irr}} = 0.18$) dependent on both S and k (left column: $\sigma_c = 10^{-2} \text{ g} \cdot \text{sec}^{-2}$; right column: $\sigma_c = 10^{-4} \text{ g} \cdot \text{sec}^{-2}$) (a) at $t = 2.182 \text{ sec}$ with 5491 cells and 57718 DoFs (b) at $t = 2.182 \text{ sec}$ with 4264 cells and 45248 DoFs (c) at $t = 4.364 \text{ sec}$ with 9109 cells and 94778 DoFs (d) at $t = 4.364 \text{ sec}$ with 7015 cells and 73542 DoFs

5.5 Effect of different capillary intensities

For this case, $N_s = 4$. The immobile saturation S_{irr} is set to 0.18, the surface tension σ_c is set to more than one value (i.e. 10^{-2} and $10^{-4} \text{ g} \cdot \text{sec}^{-2}$) for comparison, and the permeability gradient effect appearing in both equations (13) and (15) is considered.

Figure 9 shows the time evolution of the adaptive refinement meshes. Apparently, since four wetting-phase sources are placed in the computational domain, more grids are needed

to accurately capture the complex spatiotemporal flow behavior as time passes. Again, our numerical implementation demonstrates its adaptive ability to accurately resolve the important flow features in each time step, while increasing the computational efficiency with the aid of adaptive mesh refinement/coarsening.

Consider the wetting-phase transport alone. The mechanism of the wetting-phase transport can be split into two parts: one governed by two-phase mixing convection (i.e. the second term in (15)) and the other by capillary diffusive transport (i.e. the third term in (15)). Those two mechanisms pushes the wetting-phase fluid to advance. But those two different ways produce different flow patterns. The saturation contours are shown in Figure 10. When capillary diffusion is stronger than convection (see the left column of Figure 10), capillary diffusion significantly smears sharp fronts produced by convection. This can be viewed as a diffusion-dominated case. In contrast, when convection acts stronger than capillary transport (see the right column of Figure 10), the wetting-phase fluid goes in a particular direction (convection) rather than going in all direction (diffusion). The different flow patterns need to be numerically treated in different ways to satisfy the individual physical meaning. This is a reason why people need to use an upwind scheme for the convection term instead of adopting a central difference scheme.

6 Conclusions and future work

In conclusion, there are two major improvements in the present work: one for a numerical issue and the other for the physical modelling. For numerical part, an integrated numerical method was presented to investigate the combined effects of two-phase advective transport and capillary transport with consideration of heterogeneities and, with the aid of grid adaptation, we can obtain the numerical results quickly. Most importantly, it incorporates a recently developed stabilization for continuous finite element discretization allowing both

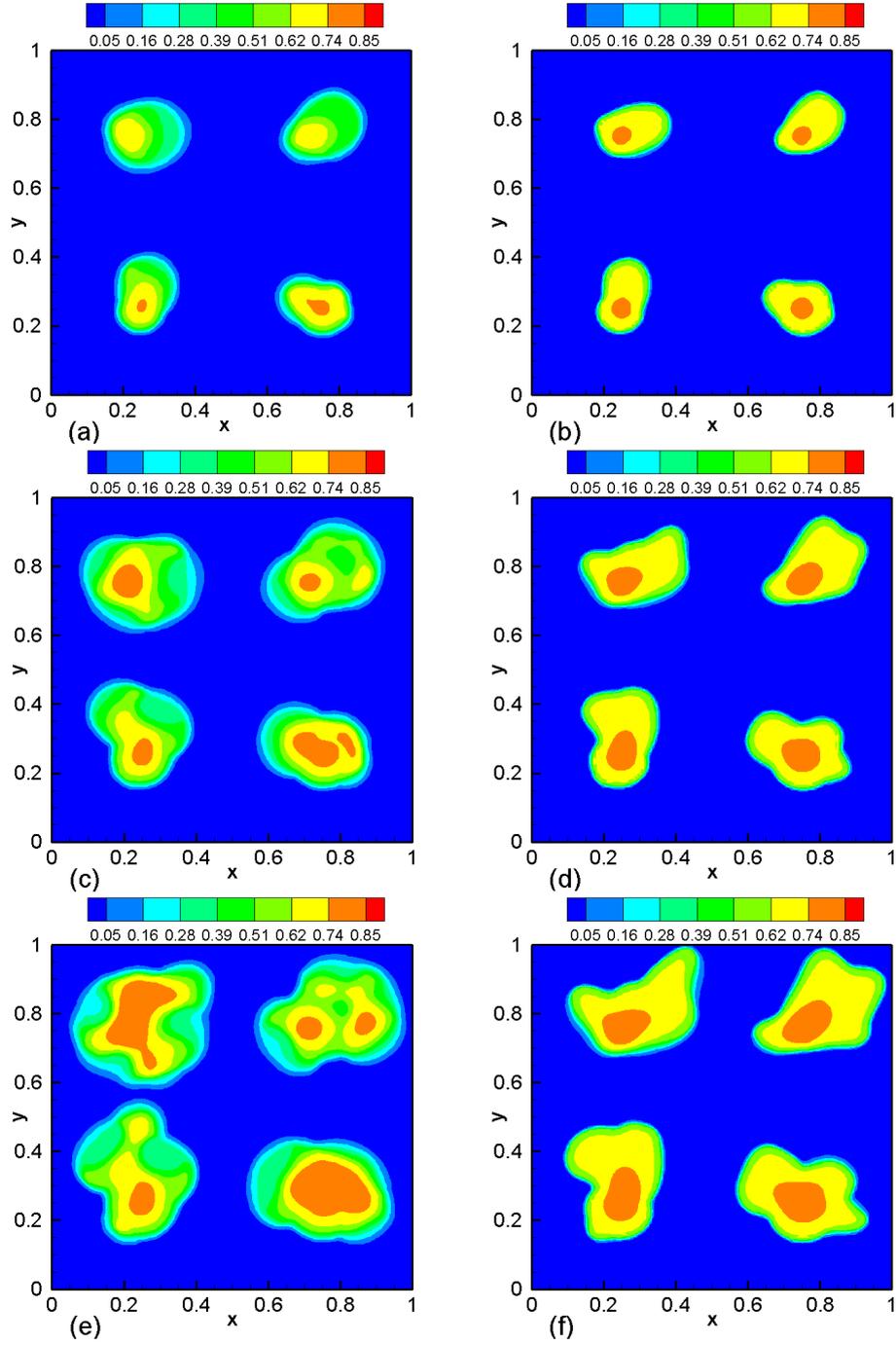


Fig. 10. Comparison of wetting-phase saturation fields in the case of capillary pressure dependent on both S and k with $N_s = 4$ and $S_{\text{irr}} = 0.18$ (left column: $\sigma_c = 10^{-2} g \cdot \text{sec}^{-2}$; right column: $\sigma_c = 10^{-4} g \cdot \text{sec}^{-2}$) (a) at $t = 1.452 \text{ sec}$ (b) at $t = 1.452 \text{ sec}$ (c) at $t = 2.904 \text{ sec}$ (d) at $t = 2.904 \text{ sec}$ (e) at $t = 4.357 \text{ sec}$ (f) at $t = 4.375 \text{ sec}$

higher order accuracy and stability both, which has been discussed and validated in [4,5]. For modelling part, the method coupled with the consideration of irreducible or immobile saturation has made our modelling more representative of realistic physical problems.

In addition, the permeability gradient effect illustrates the effect of porous heterogeneities on fluid behaviours. Here, we compare and discuss the different effects acting on capillary transport (i.e. $C_p(S)$ and $C_p(S, k)$) as well as the effects of the amount of immobile saturation on spatiotemporal two-phase fluid behaviours. This is done, even though the standard Leverett J function relationship to correlate capillary pressure with saturation and permeability is unable to accurately display transport phenomena in heterogeneous porous media primarily because it is an inherently volume-averaged function. It might be problematic when highly inhomogeneous media are considered or when it is used to account for flow behaviours through variable porous structures. Therefore, a better, accurate relationship will be considered in future work.

7 Acknowledgments

The work of the first and second authors was funded through the Canada Research Chairs Program and the MITACS Network of Centres of Excellence. Parts of the work by the third author were funded through Award No. KUS-C1-016-04, made by the King Abdullah University of Science and Technology, and through an Alfred P. Sloan Research Fellowship. Some computational resources used here were also funded by NSF award DMS-0922866.

Thanks are due to Dr. Zhangxin Chen (University of Calgary) for his invaluable insights on finite element theory, to Dr. Sergey Mashchenko (McMaster University) for installing the relevant libraries, and to the first author's wife, Mrs. Hsiao-Yun Janette Cheng (PhD candidate, University of Sheffield), for helping with the graphics.

References

- [1] MM Mench. *Fuel Cell Engines*. Wiley, 2008.
- [2] H Li, Y Tang, Z Wang, Z Shi, S Wu, D Song, J Zhang, K Fatih, J Zhang, H Wang, Z Liu, R Abouatallah, and A Mazza. A review of water flooding issues in the proton exchange membrane fuel cell. *Journal of Power Sources*, 178(1):103 – 117, 2008.
- [3] R Mosdale, G Gebel, and M Pineri. Water profile determination in a running proton exchange membrane fuel cell using small-angle neutron scattering. *Journal of Membrane Science*, 118(2):269 – 277, 1996.
- [4] CC Chueh, M Secanell, W Bangerth, and N Djilali. Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media. *Computers and Fluids*, 39:1585–1596, 2010.
- [5] CC Chueh, W Bangerth, and N Djilali. An h-adaptive operator splitting method for two-phase flow in 3D heterogeneous porous media. *Submitted to Communications in Computational Physics*.
- [6] Z Chen, G Huan, and B Li. An improved impes method for two-phase flow in porous media. *Transport in Porous Media*, 54:361 – 376, 2004.
- [7] MC Leverett. Capillary behaviour in porous solids. *AIME Trans.*, 142:152 – 169, 1941.
- [8] JH Nam and M Kaviany. Effective diffusivity and water-saturation distribution in single-and-two layer pemfc diffusion medium. *International Journal of Heat and Mass Transfer*, 46:4595 – 4611, 2003.
- [9] J Bear. *Dynamics of Fluids in Porous Media*. American Elsevier, 1972.
- [10] S Lister and N Djilali. *Two-phase transport in porous gas diffusion electrodes (Eds. M. Faghri & B. Sunden)*. WIT Press, Southampton, UK, 2005.

- [11] KS Udell. Heat transfer in porous media considering phase change and capillary - the heat pipe effect. *International Journal of Heat and Mass Transfer*, 28(2):485 – 495, 1985.
- [12] M Kaviany. *Principles of Heat Transfer in Porous Media*. Springer-Verlag, New York, 1991.
- [13] JF App and KK Mohanty. Gas and condensate relative permeability at near-critical conditions: capillary and reynolds number dependence. *Journal of Petroleum Science and Engineering*, 36(1-2):111 – 126, 2002.
- [14] A Papafotiou, H Sheta, and R Helmig. Numerical modeling of two-phase hysteresis combined with an interface condition for heterogeneous porous media. *Computational Geosciences*, 14:273–287, 2010.
- [15] F Brezzi and M Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, 1991.
- [16] Z Chen. *Finite Element Methods and Their Applications*. Springer, 2005.
- [17] JL Guermond and R Pasquetti. Entropy-based nonlinear viscosity for fourier approximations of conservation laws. *Comptes Rendus Mathematique*, 346(13-14):801 – 806, 2008.
- [18] W Bangerth, R Hartmann, and G Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [19] W Bangerth, R Hartmann, and G Kanschat. deal.II *Differential Equations Analysis Library, Technical Reference*, 2010. <http://www.dealii.org/>.
- [20] Y Li and W Bangerth. *The deal.II tutorial manual: step-21*. http://dealii.org/developer/doxygen/deal.II/step_21.html.
- [21] M Kronbichler and W Bangerth. *The deal.II tutorial manual: step-31*. http://dealii.org/developer/doxygen/deal.II/step_31.html.
- [22] D Neckels and W Bangerth. *The deal.II tutorial manual, step-33*. http://dealii.org/developer/doxygen/deal.II/step_33.html.
- [23] M Honarpour, L Koederitz, and AH Harvey. *Relative Permeability of Petroleum Reservoirs*. First edition, CRC Press, 1986.

- [24] JA Trangenstein. Multi-scale iterative techniques and adaptive mesh refinement for flow in porous media. *Advances in Water Resources*, 25(8-12):1175 – 1213, 2002.

Appendix D

Numerical Analysis of Transient Two-phase Flow in Hydrophobic Heterogeneous Porous Media

Numerical Analysis of Transient Two-phase Flow in Hydrophobic Heterogeneous Porous Media

C.C. Chueh^a, N. Djilali^{a,*}, W. Bangerth^b

^a*Institute for Integrated Energy Systems and Department of Mechanical Engineering,
University of Victoria, PO Box 3055 STN CSC, Victoria, BC, Canada, V8W 3P6*

^b*Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA*

Abstract

This paper presents numerical analysis of two-phase flow in heterogeneous porous media with hydrophobic capillary transport. The analysis is conducted using an efficient, accurate set of algorithms which has been recently developed and validated for an advection-dominated case with no capillary transport in our previously published papers [1,2]. We focus more on the solution of one numerical issue (saturation shocks or discontinuities) that is hard to numerically solve and causes faulty saturation fronts commonly happening in mixture PEM fuel cell two-phase flow modelling. Most importantly, we provide a solution approach using a recently developed artificial diffusive stabilization term which is added into the saturation transport equation to eliminate unphysical oscillations while ensuring the accuracy of solutions. In addition to this, we also demonstrate complex transport phenomena between two fluids in heterogeneous porous media with consideration of capillary effects.

Key words: Adaptive mesh refinement; Stabilized finite element method; Operator splitting; Preconditioning; Two-phase flow; Heterogeneous porous media; Fuel cells

1 Introduction

The simulation of two-phase flow in porous media plays a decisive part in polymer electrolyte membrane (PEM) fuel cells [3–5] but it faces many difficult problems that need to be surmounted. One of the problems is caused by excessive amounts of water, which is produced electrochemically at the catalyst sites of PEM fuel cells. Large amounts of water can hinder reactant gas transport, resulting a poor cell performance. Improvements in PEM fuel cells are therefore contingent upon advancements in water management, which can be facilitated by the simulation of realistic, complex two-phase transport phenomena in heterogeneous porous media.

However, getting correct solutions is not easy since discontinuities or shocks of any physical quantity caused by flow advection increase the difficulty of solving the transport equations without an appropriate numerical scheme. We have previously validated and demonstrated an artificial diffusion method that, when used with a general continuous finite element discretization for the saturation transport (advection) equation in two-phase flow in porous media, ensures stability and accuracy of the solution [1,2]. The method uses an entropy-based diffusion term, proposed by Guermond and Pasquetti [6], and is able to efficiently damp unphysical oscillations while providing the same resolution of the saturation field as a low-order DG method [7] without upwinding. The trick is to ensure that the artificial viscosity term acts only in the vicinity of strong saturation gradients and other discontinuities, while disabling it in regions where the solution is smooth. The scheme offers higher order accuracy in smooth regions while providing stability where necessary [6].

There is one unresolved problem in the field of PEM fuel cell modelling: when a mix-

* Corresponding author.

Email address: ndjilali@uvic.ca (N. Djilali).

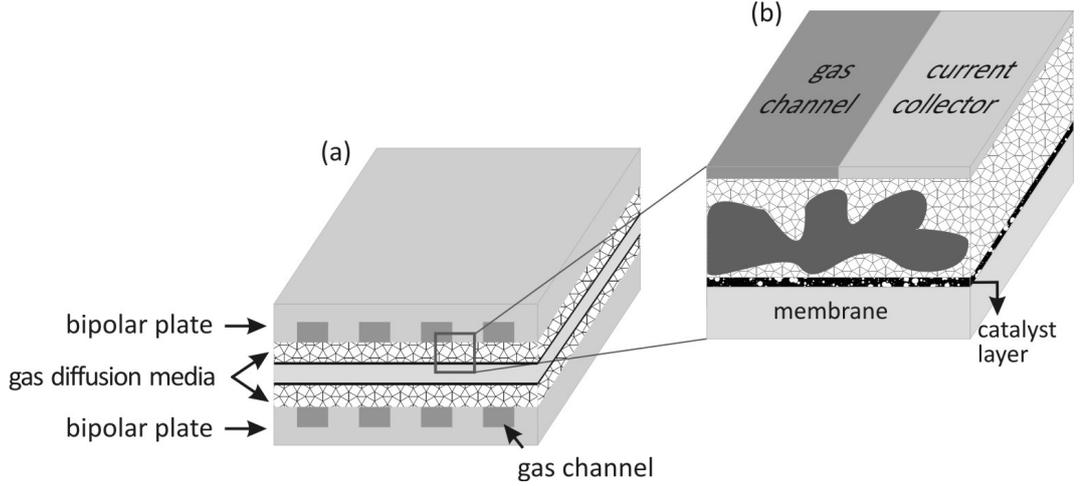


Fig. 1. Schematic showing simulation with respect to PEMFC geometry: (a) diagram of PEMFC (b) computational domain of a simulated diffusion medium

ture multi-phase model is employed to simulate the transport phenomena of PEM fuel cells, some important mass and momentum transport is not considered in the model as reported in [8]. The authors in [8] note that this problem originates from the mixture model's inability to correctly capture the complex mass and momentum transport compared to multi-fluid modelling. However, this problem is mostly caused by a numerical issue. When advective flow is considered in the formulation of the mixture multi-phase flow, solution discontinuities are created, leading to wrong or faulty solutions if an inappropriate numerical scheme is used.

The current study is motivated by the need to solve the numerical issue mentioned above and provide a solution approach to advection-dominated flow to the field of mixture multi-phase fuel cell modelling. The solution uses a recently developed artificial diffusive stabilization to effectively damp unphysical oscillations while retaining the accuracy of the simulation. To demonstrate the powerful capability of this diffusive scheme, we first show numerical validation of an advection-dominated two-phase flow (with no capillarity) in uniform porous media using real physical parameters (i.e. air and water viscosity). Next, to demonstrate numerical results which consider capillary transport, we use the two-

dimensional cross-section representation of the membrane electrolyte assembly including half of the gas channel and half of the current collector, as shown in Figure 1. We also adopt a recently developed relationship [9] between capillary pressure and saturation which is representative of material properties of PEMFC diffusion media.

2 Mixture Model of Two-phase Flow in Heterogeneous Porous Media

In this section, we derive the governing equations of a mixture model for a hydrophilic case, define relevant physical parameters and coefficients, boundary and initial conditions as well as introduce an absolute permeability model. All of the involved parameters are shown in Table 1.

2.1 Governing equations

Here a two-phase immiscible, incompressible, isothermal fluid is considered and gravity effects are neglected. The governing equations for the fluid flowing in hydrophobic porous media are expressed in terms of the nonwetting phase pressure and saturation.

General governing equations The mass balance equations for wetting and nonwetting phases are respectively

$$\epsilon \frac{\partial (\rho_w S_w)}{\partial t} + \nabla \cdot (\rho_w \mathbf{u}_w) = \rho_w q_w \quad (1)$$

$$\epsilon \frac{\partial (\rho_{nw} S_{nw})}{\partial t} + \nabla \cdot (\rho_{nw} \mathbf{u}_{nw}) = \rho_{nw} q_{nw} \quad (2)$$

where S is the saturation (volume fraction) indicating the composition of mixture of fluids and subscripts w and nw denote the wetting and non-wetting phase respectively, ρ is the density, ϵ is the porosity, q is the source term and \mathbf{u} is the velocity. The velocities \mathbf{u}_w and

\mathbf{u}_{nw} are modeled by Darcy's Law

$$\mathbf{u}_w = -\frac{k_{rw}}{\mu_w} \mathbf{K} \nabla p_w = -\lambda_w \mathbf{K} \nabla p_w \quad (3)$$

$$\mathbf{u}_{nw} = -\frac{k_{rnw}}{\mu_{nw}} \mathbf{K} \nabla p_{nw} = -\lambda_{nw} \mathbf{K} \nabla p_{nw} \quad (4)$$

where \mathbf{K} is the permeability tensor, λ is the fluid mobility, k_r is the relative permeability, μ is the viscosity and p is the pressure. In addition to the equations shown above, the constraint for the saturation is

$$S_w + S_{nw} = 1 \quad (5)$$

and the two pressures are related by a capillary pressure function

$$p_c = p_{nw} - p_w \quad (6)$$

The total mobility λ_t and fractional flow of the wetting phase F are respectively defined by

$$\lambda_t = \lambda_w + \lambda_{nw} = \frac{k_{rw}}{\mu_w} + \frac{k_{rnw}}{\mu_{nw}} \quad (7)$$

$$F = \frac{\lambda_w}{\lambda_t} = \frac{\lambda_w}{\lambda_w + \lambda_{nw}} = \frac{k_{rw}/\mu_w}{k_{rw}/\mu_w + k_{rnw}/\mu_{nw}} \quad (8)$$

Hydrophobic case In this case, \mathbf{u}_t , p_{nw} and S_{nw} are unknown variables and therefore all other variables can be expressed in terms of these three unknown variables. After a lengthy derivation process similar to that described for the hydrophilic case in [10], the transport of two fluids in *hydrophobic* porous media is governed by the following set of equations

$$\mathbf{u}_t = -\lambda_t \mathbf{K} \nabla p_{nw} + \lambda_w \mathbf{K} \nabla p_c \quad (9)$$

$$\nabla \cdot \mathbf{u}_t = q_w + q_{nw} \quad (10)$$

$$\epsilon \frac{\partial S_{nw}}{\partial t} + \nabla \cdot \left((1 - F) \mathbf{u}_t \right) - \nabla \cdot (\lambda_{nw} F \mathbf{K} \nabla p_c) = q_{nw} \quad (11)$$

The capillary pressure p_c in equations (9) and (11) is correlated as a function of the saturation. This relationship, proposed by Leverett [11,12], is the so-called Leverett J-function:

$$J(S_e) = \frac{p_c}{\sigma_c \cos(\theta_c)} \left(\frac{k}{\epsilon} \right)^{1/2} \quad (12)$$

where σ_c is the surface tension, k is the permeability, θ_c is the contact angle varying between 90° and 180° for the hydrophobic case, and S_e is the effective saturation [13,3,14], defined as

$$S_e = \frac{S_{nw} - S_{irr}}{1 - S_{irr}} \quad (13)$$

where S_{irr} is the so-called immobile or irreducible saturation [13,15], which stands for the amount of entrapped liquid fluid. It can be regarded as a threshold point below which liquid remains immobile.

Here we use a validated J capillary relationship developed by Kumbur et al. [9] for PEMFC diffusion media between capillary pressure and saturation as

$$K(S_{nw}) = \begin{cases} wt\% (0.0469 - 0.00152(wt\%) - 0.0406S_{nw}^2 + 0.143S_{nw}^3) + 0.0561 \ln S_{nw} \\ \quad \text{if } 0 < S_{nw} < 0.5 \\ \\ wt\% (1.534 - 0.0293(wt\%) - 12.68S_{nw}^2 + 18.824S_{nw}^3) + 3.416 \ln S_{nw} \\ \quad \text{if } 0.5 \leq S_{nw} \leq 0.65 \\ \\ wt\% (1.7 - 0.0324(wt\%) - 14.1S_{nw}^2 + 20.9S_{nw}^3) + 3.79 \ln S_{nw} \\ \quad \text{if } 0.65 < S_{nw} < 1.0 \end{cases}$$

where $wt\%$ represents PTFE (one hydrophobic material coating carbon fibers) weight percentage in PEMFC diffusion media. Moreover, the relationship represents physical properties of PEMFC diffusion media [9]. Here, since the validated J-function $K(S_{nw})$ already incorporates the wettability of air and water inside equation (14), we can cancel

the contact angle θ_c in the original Leverett-J relation (12) and then rearrange it as

$$p_c = \frac{\sigma_c}{(k/\epsilon)^{1/2}} K(S_{nw}) \quad (14)$$

Furthermore, in order to make the system well-posed physically and mathematically, $\partial p_c / \partial S_{nw}$ has to have a positive sign so that the capillary diffusion term in equation (11) is always negative. Its positive slope can be proved by checking the form of equation (14) in Figure 2.

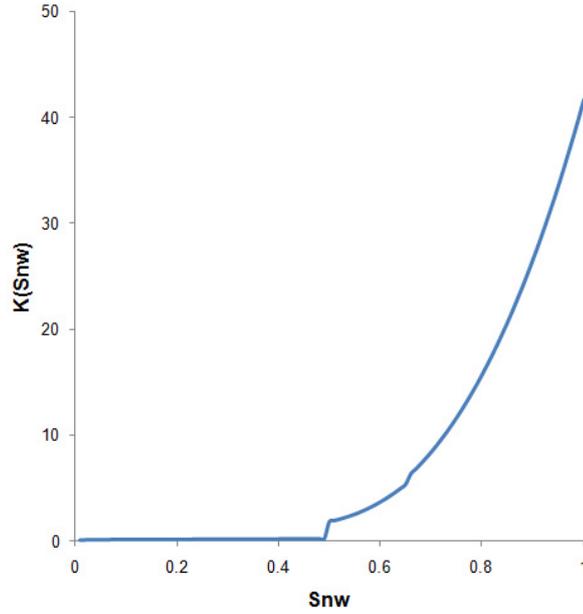


Fig. 2. A validated J function ($K(S_{nw})$ vs S_{nw}) with $wt\% = 5$ for PEMFC diffusion media developed by Kumbur et al. [9]

Then, ∇p_c in equations (9) and (11) is further obtained:

$$\nabla p_c = \frac{\sigma_c}{(k/\epsilon)^{1/2}} \frac{d(K(S_{nw}))}{dS_{nw}} \nabla S_{nw} + \sigma_c K(S_{nw}) \epsilon^{\frac{1}{2}} \left(-\frac{1}{2} k^{-\frac{3}{2}} \right) \nabla k \quad (15)$$

When the above equation is substituted into equations (9) and (11) respectively, we obtain the final governing equations considered herein.

The considered heterogeneous porous media are set to be isotropic. Thus, $\mathbf{K}(\mathbf{x}) = \gamma_m \cdot$

$k_{rm}(\mathbf{x}) \cdot \mathbf{I}$, where k_{rm} which is a scalar function of \mathbf{x} represents a randomly distributed medium, \mathbf{I} is the second-order unit tensor and γ_m is a magnitude scaling factor of permeability. In order to demonstrate the robustness of this proposed shock-type adaptive refinement technique, one permeability model already used in one tutorial program implemented by Li and Bangerth [7] is introduced into this work. The permeability model is defined to be $k_{rm}(\mathbf{x})$

$$k_{rm}(\mathbf{x}) = \min \left\{ \max \left\{ \sum_{l=1}^{N_p} \Psi_l(\mathbf{x}), 10^{-10} \right\}, 3.0 \times 10^{-8} \right\} \quad (16)$$

where

$$\Psi_l(\mathbf{x}) = \exp \left(- \left(\frac{|\mathbf{x} - \mathbf{x}_l|}{B_w} \right)^2 \right) \quad (17)$$

where the centres \mathbf{x}_l are N_p randomly chosen locations inside the domain and B_w is the band width of the exponential function. This function models a domain in which there are N_p centres of higher permeability in the cracked region of the rock embedded in a matrix of more pristine, unperturbed background rock. The permeability function is bounded both above and below to limit the size of the condition number [7]. Figure 3 shows the permeability field contour with $N_p = 15$ used in the present work. Here, the permeability that is confined to a range of $O(10^{-8})$ to $O(10^{-10}) \text{ cm}^2$ satisfies the permeability range for PEMFC diffusion media.

2.2 Computational Domain and Initial and Boundary Conditions

The simulations in the hydrophobic case are performed on the computational domain $\Omega = [0, L] \times [0, L]$ in two dimensions for $t \in [0, T]$ where L is set to 0.025. The initial condition is set to $S_{nw}(\mathbf{x}, 0) = 0$ in the non-wetting phase reservoir picture. The boundary

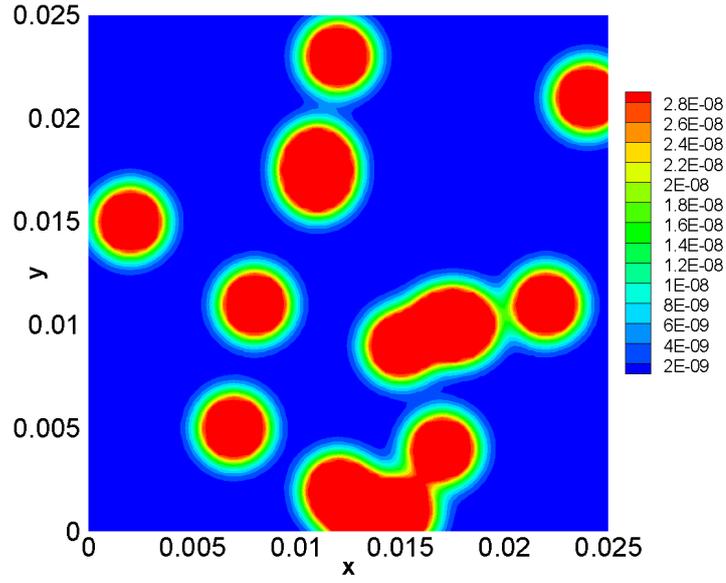


Fig. 3. Permeability contour used in this work

conditions for the velocity/pressure equations (9) and (10) in two dimensions are set to

$$p_{nw}(\mathbf{x}, t) = 0 \quad \text{for } (x, y) \in \{x = [0, L/2]\} \times \{y = L\}, \quad (18)$$

$$\mathbf{u}_t = \mathbf{0} \quad \text{for } (x, y) \in \{x = [L/2, L]\} \times \{y = L\}, \quad (19)$$

$$\mathbf{u}_t = \mathbf{0} \quad \text{for } (x, y) \in \{x\} \times \{y = 0\}, \quad (20)$$

$$\mathbf{u}_t \cdot \mathbf{n} = 0 \quad \text{for } (x, y) \in \{x = 0\} \times \{y\}, \quad (21)$$

$$\mathbf{u}_t \cdot \mathbf{n} = 0 \quad \text{for } (x, y) \in \{x = L\} \times \{y\} \quad (22)$$

The boundary conditions for the saturation transport equation (11) in two dimensions are set to

$$S_{nw}(\mathbf{x}, t) = 0 \quad \text{on} \quad \Gamma_{in}(t) \quad \text{for} \quad (x, y) \in \{x = [0, L/2]\} \times \{y = L\}, \quad (23)$$

$$\mathbf{n} \cdot \nabla S_{nw} = 0 \quad \text{for} \quad (x, y) \in \{x = [L/2, L]\} \times \{y = L\}, \quad (24)$$

$$\mathbf{n} \cdot \nabla S_{nw} = 0 \quad \text{for} \quad (x, y) \in \{x\} \times \{y = 0\}, \quad (25)$$

$$\mathbf{n} \cdot \nabla S_{nw} = 0 \quad \text{for} \quad (x, y) \in \{x = 0\} \times \{y\}, \quad (26)$$

$$\mathbf{n} \cdot \nabla S_{nw} = 0 \quad \text{for} \quad (x, y) \in \{x = L\} \times \{y\} \quad (27)$$

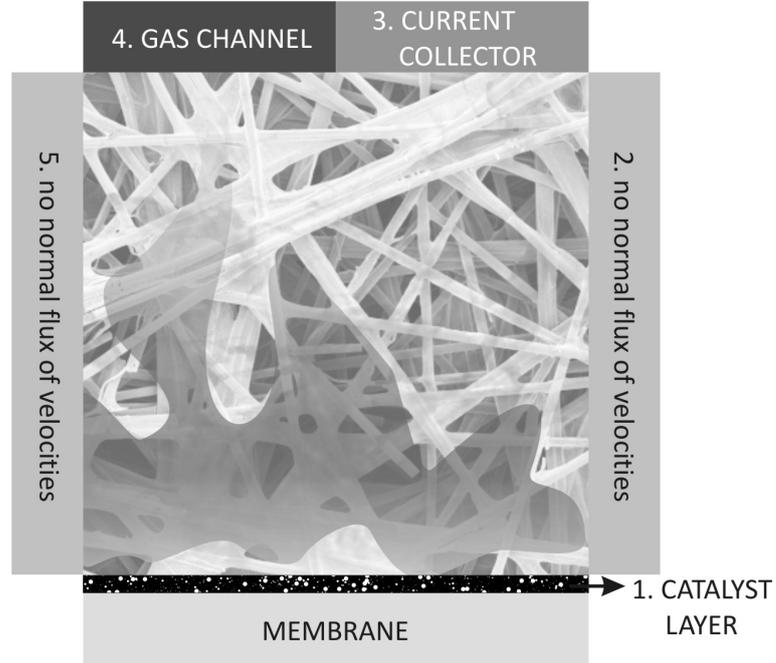


Fig. 4. Boundary conditions imposed in this work

The arrangement of these boundaries (see Figure 4) in the hydrophobic case is meant to ideally construct the configuration of the gas diffusion layer (GDL) of PEM fuel cells. The boundary at $(x, y) \in \{x = [0, L/2]\} \times \{y = L\}$ in two dimensions represents the gas channel-GDL interface and the boundary at $(x, y) \in \{x = [L/2, L]\} \times \{y = L\}$ in two dimensions denotes the current collector-GDL interface. The boundary at $(x, y) \in \{x\} \times \{y = 0\}$ in two dimensions stands for the GDL-catalyst layer interface.

2.3 Numerical Methods and Implementations

Adaptive operator splitting and time stepping The time stepping schemes most commonly used to solve equations (9)–(11) of the kind are IMPES (*implicit pressure, explicit saturation*) type where one first solves the implicit pressure/velocity system (9)–(10) with the current saturation values, and then uses an explicit time stepping scheme to advance the saturation by one time step using (11). The vast majority of computing time in such schemes is spent in the implicit solver for the pressure and velocity variables. Computing efforts may be significantly reduced by noting that the pressure and velocity fields depend only weakly on the saturation, and therefore do not change significantly between time steps, whereas saturation fronts typically move by one cell in each step. To deal with this point, we have developed an adaptive operator splitting with an a posteriori criterion for advection-dominated two-phase flow [2]. This criterion is still used in this study and is given as

$$\theta(n, n_p) = \max_{\kappa \in \mathbb{T}} \left(\left\| \frac{1}{\lambda_t(S_{nw}^{(n-1)})} - \frac{1}{\lambda_t(S_{nw}^{(n_p-1)})} \right\|_{L^\infty(\kappa)} \left\| \mathbf{K}^{-1} \right\|_1 \right)_{L^\infty(\kappa)}. \quad (28)$$

where superscripts in parentheses denote the number of the time step at which a quantity is defined and $n_p < n$ is the last step at which we actually computed the velocity and pressure variables. Readers are referred to our previous paper for a more detailed algorithm [2].

Space discretization for velocity/pressure part Continuous finite elements are used to discretize the velocity and pressure fields. In order to approximate both a vector variable (e.g. a fluid velocity) and one scalar variable (e.g. pressure) simultaneously and to give a high order approximation of these variables, mixed finite elements are used. For the mixed finite element spaces, it is well established that the so-called Babuska-Brezzi or Ladyzhenskaya-Babuska-Brezzi (LBB) conditions [16,17] need to be satisfied to ensure

the stability of the pressure-velocity system. These stability conditions are satisfied in the present work by using elements for velocity that are one order higher than for the pressure, i.e. $u_h \in Q_{p+1}^d$ and $p_h \in Q_p$, where $p = 1$, d is the space dimension, and Q_s denotes the space of tensor product Lagrange polynomials of degree s in each variable.

Space discretization of saturation transport equation Continuous finite elements are used to discretize the saturation field, i.e. $S_h \in Q_1$. This was chosen even though discontinuous Galerkin (DG) finite element methods have become more popular for advection problems [17] primarily because we have a capillary pressure (diffusion) term. The discretization of the Laplace operator (or diffusion term) using DG methods leads to a significant number of additional terms that need to be integrated on each face between cells. Discontinuous Galerkin finite elements also have the drawback that the use of numerical fluxes introduces an additional numerical diffusion that acts everywhere. In order to avoid the cumbersome problems associated with DG finite element space, an alternative method is to add a stabilization term in the form of a nonlinear viscosity into the saturation transport equation (11), i.e.

$$\epsilon \frac{\partial S_{nw}}{\partial t} + \nabla \cdot ((1 - F) \mathbf{u}_t) - \nabla \cdot (\lambda_{nw} F \mathbf{K} \nabla p_c) - q_w - \nabla \cdot (\nu(S_{nw}) \nabla S_{nw}) = 0 \quad (29)$$

where $\nu(S_w)$ is the amount of artificial viscosity. The stabilization factor ν is chosen in such a way that, if the discretized saturation S_{nw} satisfies the original equation (11), the artificial diffusion term is zero. In other words, this term acts primarily in the vicinity of discontinuities (or other dependent variables) where our numerical approximation is not very accurate, and does not affect the areas where the saturation varies smoothly and our numerical approximation is accurate. The literature contains a number of approaches to achieve this. Here we adopt a scheme developed by Guermond and Pasquetti [6] which builds on a suitably defined residual and a limiting procedure for the additional viscosity. The artificial viscosity is a piecewise constant function defined on each cell K with

diameter h_K as

$$\nu(\mathbf{u}_t, S_{nw})|_K = \beta \|\mathbf{u}_t\|_{L^\infty(K)} \min \left\{ h_K, h_K^\alpha \frac{\|\text{Res}(\mathbf{u}_t, S_{nw})\|_{L^\infty(K)}}{c(\mathbf{u}_t, S_{nw})} \right\} \quad (30)$$

where α is a stabilization exponent that we chose to be $\alpha = 1$, and β is a user-defined dimensionless stabilization constant that is chosen to be as small as possible but as large as necessary to avoid unphysical oscillations in the solution [1]. Following Guermond and Pasquetti [6] as well as Kronbichler and Bangerth [18], the velocity and saturation global normalization constant, $c(\mathbf{u}_t, S_{nw})$, and the residual $\text{Res}(\mathbf{u}_t, S_{nw})$ are respectively given by

$$c(\mathbf{u}_t, S_{nw}) = c_R \|\mathbf{u}_t\|_{L^\infty(\Omega)} \text{var}(S_{nw}) |\text{diam}(\Omega)|^{\alpha-2} \quad (31)$$

where c_R is the dimensionless user-defined constant, $\text{diam}(\Omega)$ is the diameter of the domain and $\text{var}(S_{nw})$ is the range of the average saturation values in the entire computational domain Ω , given mathematically by

$$\text{var}(S_{nw}) = \max_{\Omega} S_{nw} - \min_{\Omega} S_{nw} \quad (32)$$

and

$$\text{Res}(\mathbf{u}_t, S_{nw}) = D_R \cdot (S_{nw})^{\alpha-1} \quad (33)$$

where

$$D_R = \epsilon \frac{\partial S_{nw}}{\partial t} + \nabla \cdot ((1 - F) \mathbf{u}_t) - \nabla \cdot (\lambda_{nw} F \mathbf{K} \nabla p_c) - q_w \quad (34)$$

Guermond and Pasquetti [6] demonstrate excellent performance and computational results for this scheme and provide details on the derivation. The stabilization term is critical in order to obtain a saturation field that is oscillation free. Results already discussed and validated in [1] using the above method are free of unphysical oscillations.

Adaptive mesh refinement The same shock-type refinement approach used in [1] is adopted in an attempt to reduce the number of degrees of freedom, resulting in a significant increase in computational efficiency and fine-scale resolution. The refinement indicator for

each cell centre represents the scale of the saturation gradient and is computed by

$$\eta_K = |\nabla S_{nw}| \quad (35)$$

This is analogous to a compressible flow problem, where density gradients are used to indicate refinement. In this work, the refinement indicator is computed for each cell and the refinement operations are performed by the following criteria; if a cell has a indicator value greater than 1.7, the cell is refined; if a cell has an indicator less than 1.1, the cell is agglomerated. For more information regarding the implementation of mesh refinement readers are referred to [19,20].

Block matrix preconditioning Following the discretization of the governing equations (9)–(11) discussed above, we obtain a linear system of equations in time step (n) of the following form:

$$\begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T & \mathbf{R} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{H} & \mathbf{0} & \mathbf{M}^S + \mathbf{Y} \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n)} \\ \mathbf{P}^{(n)} \\ \mathbf{S}_{nw}^{(n)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \end{pmatrix} \quad (36)$$

where the individual matrices and vectors are defined as follows using shape functions \mathbf{v}_i for velocities and ϕ_i for both pressures and saturations:

$$\mathbf{M}_{ij}^u = \left(\left(\mathbf{K} \lambda_t (S_{nw}^{(n-1)}) \right)^{-1} \mathbf{v}_i, \mathbf{v}_j \right)_\Omega \quad (37)$$

$$\mathbf{B}_{ij} = - \left(\nabla \cdot \mathbf{v}_j, \phi_i \right)_\Omega \quad (38)$$

$$\mathbf{R}_{ij} = - \left(\left(\mathbf{I} \lambda_t (S_{nw}^{(n-1)}) \right)^{-1} \lambda_{nw} (S_{nw}^{(n-1)}) \frac{\partial p_c}{\partial S_{nw}} \nabla \phi_i, \mathbf{v}_j \right)_\Omega \quad (39)$$

$$(\mathbf{F}_1)_i = \left(\left(\mathbf{I} \lambda_t (S_{nw}^{(n-1)}) \right)^{-1} \lambda_{nw} (S_{nw}^{(n-1)}) \frac{\partial p_c}{\partial k} \nabla k, \mathbf{v}_i \right)_\Omega \quad (40)$$

$$(\mathbf{F}_2)_i = - \left((q_w + q_{nw}), \phi_i \right)_\Omega \quad (41)$$

and

$$\mathbf{M}_{ij}^S = (\epsilon \phi_i, \phi_j)_\Omega \quad (42)$$

$$\mathbf{Y}_{ij} = \Delta t \left(\left(\lambda_{nw} (S_{nw}^{(n-1)}) F (S_{nw}^{(n-1)}) \mathbf{K} \frac{\partial p_c}{\partial S_{nw}} \right) \nabla \phi_i, \nabla \phi_j \right)_\Omega \quad (43)$$

$$\mathbf{H}_{ij} = -\Delta t \left(\left(1 - F (S_{nw}^{(n-1)}) \right) \mathbf{v}_i, \nabla \phi_j \right)_\Omega \quad (44)$$

$$\begin{aligned} (\mathbf{F}_3)_i &= (\epsilon S_{nw}^{(n-1)}, \phi_i)_\Omega + \Delta t (q_{nw}, \phi_i)_\Omega - \Delta t \left(\nu_\alpha (S_{nw}^{(n-1)}) \nabla S_{nw}^{(n-1)}, \nabla \phi_i \right)_\Omega \\ &\quad + \Delta t \left(\nabla \cdot \left(\lambda_{nw} (S_{nw}^{(n-1)}) F (S_{nw}^{(n-1)}) \mathbf{K} \frac{\partial p_c}{\partial k} \nabla k \right), \phi_i \right)_\Omega \end{aligned} \quad (45)$$

To use a linear solver implemented in our previous paper [2], we have to move the block matrix \mathbf{R} to the right hand side of (36). The linear system above is in block triangular form if we consider the top left 2×2 panel of matrices as one block. We can therefore first solve for the velocity and pressure (unless we decide to use $\mathbf{U}^{(n_p)}$ in place of the velocity, as discussed in [2] followed by a solve for the saturation variable. The first of these steps requires us to solve

$$\begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}^{(n)} \\ \mathbf{P}^{(n)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{pmatrix} \quad (46)$$

We apply the Generalized Minimal Residual (GMRES) method (Saad and Schultz [21]) to this linear system. The ideal preconditioner for the system (46) is

$$\mathbf{P} = \begin{pmatrix} \mathbf{M}^u & \mathbf{0} \\ \mathbf{B} & -\mathbf{S} \end{pmatrix}, \quad \mathbf{P}^{-1} = \begin{pmatrix} (\mathbf{M}^u)^{-1} & \mathbf{0} \\ \mathbf{S}^{-1} \mathbf{B} (\mathbf{M}^u)^{-1} & -\mathbf{S}^{-1} \end{pmatrix} \quad (47)$$

where $\mathbf{S} = \mathbf{B} (\mathbf{M}^u)^{-1} \mathbf{B}^T$ is the Schur complement [22] of the system. This preconditioner

is optimal since

$$\mathbf{P}^{-1} \begin{pmatrix} \mathbf{M}^u & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & (\mathbf{M}^u)^{-1} \mathbf{B}^T \\ \mathbf{0} & \mathbf{I} \end{pmatrix}, \quad (48)$$

and consequently all eigenvalues are equal to one. GMRES with this preconditioner would then converge in one iteration.

However, we cannot of course expect to use exact inverses of the velocity mass matrix and the Schur complement. We therefore follow the approach by Silvester and Wathen [23] originally proposed for the Stokes system. Adapting it to the current set of equations yield the preconditioner

$$\tilde{\mathbf{P}}^{-1} = \begin{pmatrix} \widetilde{(\mathbf{M}^u)^{-1}} & \mathbf{0} \\ \widetilde{\mathbf{S}^{-1}} \mathbf{B} \widetilde{(\mathbf{M}^u)^{-1}} & -\widetilde{\mathbf{S}^{-1}} \end{pmatrix} \quad (49)$$

where a tilde indicates an approximation of the exact inverse matrix. In particular, since $(\mathbf{M}^u)^{-1} = ((\mathbf{K}\lambda_t)^{-1} \mathbf{v}_i, \mathbf{v}_j)_\Omega$ is a sparse symmetric and positive definite matrix, we choose for $\widetilde{(\mathbf{M}^u)^{-1}}$ a single application of a sparse incomplete Cholesky decomposition of this matrix [24]. We note that the Schur complement that corresponds to the porous media flow operator (without capillary transport) in non-mixed form, $-\nabla \cdot [\mathbf{K}\lambda_t(S_{nw})]\nabla$ and $\tilde{\mathbf{S}} = ((\mathbf{K}\lambda_t) \nabla \phi_i, \nabla \phi_j)_\Omega$ should be a good approximation of the actual Schur complement matrix \mathbf{S} . Since both of these matrices are again symmetric and positive definite, we use an incomplete Cholesky decomposition of $\tilde{\mathbf{S}}$ for $\widetilde{\mathbf{S}^{-1}}$. It is important to note that $\tilde{\mathbf{S}}$ needs to be built with Dirichlet boundary conditions to ensure its invertibility.

Once the velocity is available, we can assemble \mathbf{Y} , \mathbf{H} and \mathbf{F}_3 and solve for the saturations

using

$$\mathbf{M}^S \mathbf{S}_{nw}^{(n)} = \mathbf{F}_3 - \mathbf{H}\mathbf{U}^{(n)} - \mathbf{Y}\mathbf{S}_{nw}^{(n)}. \quad (50)$$

where the mass matrix \mathbf{M}^S is solved by the conjugate gradient method, using an incomplete Cholesky decomposition as preconditioner once more.

2.4 Coefficients and Parameters

In the present work, there are four quantities that need to be defined. The two relative permeability curves are estimated by the following equations [25]:

$$k_{rw}(S_e) = S_e^3 \quad (51)$$

$$k_{rnw}(S_e) = (1 - S_e)^3 \quad (52)$$

And the source/sink terms q_w and q_{nw} in equations (10) and (11) are respectively defined as

$$q_w = \sum_{i=1}^{N_s} q_w^{(i)} H_{A_s^{(i)}}(\mathbf{x}) \quad (53)$$

$$q_{nw} = 0 \quad (54)$$

where $q_w^{(i)}$ indicates the volume of the fluid produced or injected per unit time at the i -th source/sink, N_s is the number of the sources/sinks and $H_{A_s^{(i)}}(\mathbf{x})$ is an indicator function at the i -th sub-domain $A_s^{(i)}$, defined as $H_{A_s^{(i)}}(\mathbf{x}) = 1$ if \mathbf{x} lies inside the i -th sub-domain and $H_{A_s^{(i)}}(\mathbf{x}) = 0$ if \mathbf{x} lies outside the i -th sub-domain. In the present work, the source $q_w^{(i)}$ for each individual sub-domain has equal magnitude, which is set to a constant value of 100 (1/sec), every sub-domain is a circle with a constant radius of 10^{-3} cm and N_s is set to a value of four. Their four locations are at $(x_c^{(1)}, y_c^{(1)}) = (0.005, 0.002)$, $(x_c^{(1)}, y_c^{(1)}) = (0.010, 0.002)$, $(x_c^{(1)}, y_c^{(1)}) = (0.015, 0.002)$, and $(x_c^{(1)}, y_c^{(1)}) = (0.020, 0.002)$.

All the sub-domains considered do not overlap each other according to their specifications described above. All the other parameters are shown in Table 1. Among them, we choose two parameters – $\mu_w = 2.075 \times 10^{-4} \text{ g} \cdot \text{cm}^{-1} \cdot \text{sec}^{-1}$ for air (wetting phase) and $\mu_{nw} = 3.55 \times 10^{-3} \text{ g} \cdot \text{cm}^{-1} \cdot \text{sec}^{-1}$ for water (non-wetting phase) – as actual physical quantities.

3 Numerical Results and Discussion

At the beginning, we first show the numerical validation of an advection-dominated two-phase flow in *uniform* porous media. Then, numerical results which consider capillary transport are shown in a computational domain.

3.1 Validation of the Buckley-Leverett problem in uniform porous media

Here, for numerical validation, we compare the present results obtained with the adaptive grid method to a previous work using uniform grids in which Raviart-Thomas finite elements are adopted for velocity and discontinuous Galerkin (DG) elements for pressure and saturation [7]. The used parameters in this validation are shown in Table 2. Initial conditions are only required for the saturation variable, and we choose $S_{nw}(\mathbf{x}, 0) = 0$, i.e. the porous medium is initially filled by the non-wetting phase. We prescribe a linear pressure on the boundaries:

$$p(\mathbf{x}, t) = 1 - x \quad \text{on} \quad \partial\Omega \times [0, T]. \quad (55)$$

The pressure and saturation uniquely determine a velocity, and the velocity determines whether a boundary segment is an inflow or outflow boundary. On the inflow part of the

boundary, $\Gamma_{in}(t)$, we impose

$$S_{nw}(\mathbf{x}, t) = 1 \quad \text{on} \quad \Gamma_{in}(t) \cap \{x = 0\}, \quad (56)$$

$$S_{nw}(\mathbf{x}, t) = 0 \quad \text{on} \quad \Gamma_{in}(t) \setminus \{x = 0\}. \quad (57)$$

In other words, the domain is flooded by the wetting phase from the left. No boundary conditions for the saturation are required for the outflow parts of the boundary.

Following a commonly used prescription for the dependence of the relative permeabilities k_{rw} and k_{rnw} on saturation, we use

$$k_{rw} = S_e^2, \quad k_{rnw} = (1 - S_e)^2. \quad (58)$$

Figure 5 shows the present numerical results in comparison with the work of Li and Bangerth [7]. In this numerical simulation where the ratio of μ_{nw} to μ_w is up to 17.11, in order to efficiently damp unphysical oscillations, we need to increase the amount of stabilization by increasing the value of β to 0.38 since numerical instabilities arise due to physical flow instabilities ($\mu_{nw}/\mu_w \gg 1$) (see, for example, Wooding and Morel-Seytoux [26], and Bear [13]). Figure 5c shows a good agreement of both the cases; however, Figure 5d shows noticeable differences in the saturation around the top and bottom corners. The differences are due to the lower accuracy and higher diffusivity of the low-order DG approximation. This approximation makes the solutions rough around complicated flow areas, resulting in the inaccuracy of the solutions.

3.2 Numerical experiments with considering capillary transport

In this section, the four circle-shaped source terms for q_{nw} are placed around the bottom of the computational domain Ω .

Figure 6 shows time-evolution saturation contours with their corresponding adaptive grids.

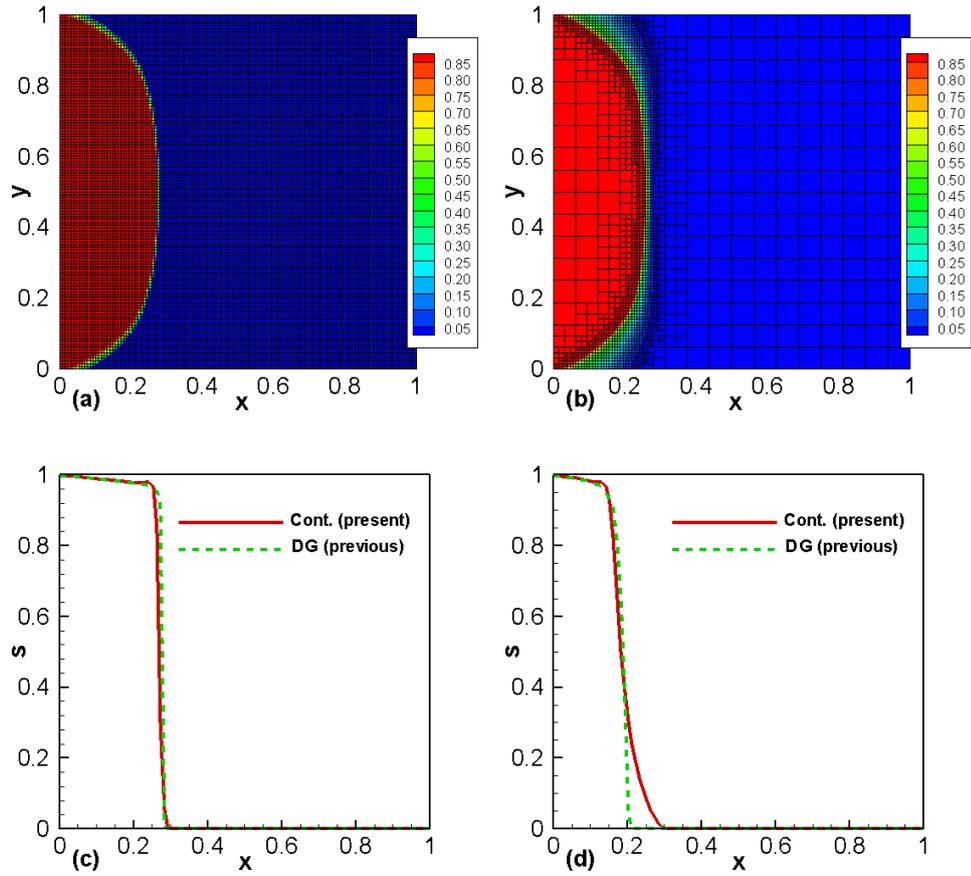


Fig. 5. Numerical results for uniform media in comparison with previous work of Li and Bangerth [7] at $t = 14.27 \text{ sec}$ when the pressure-velocity system is solved adaptively saturation time steps (adaptive operator splitting: $\theta^* = 1.0$). (a) Saturation field using discontinuous Galerkin space (previous work). (b) Saturation field using continuous space with the stabilizing term and operator splitting (present work). (c) Saturation profile along $y = 0.5$. (d) Saturation profile along $y = 0.9$.

The non-wetting phase fluid is injected into a hydrophobic porous medium initially filled with the wetting-phase fluid. The simulation refines and coarsens the meshes, depending on where discontinuities propagate. Therefore, the simulation reaches clear levels of resolution of flow field while keeping the numbers of degrees of freedom as low as possible.

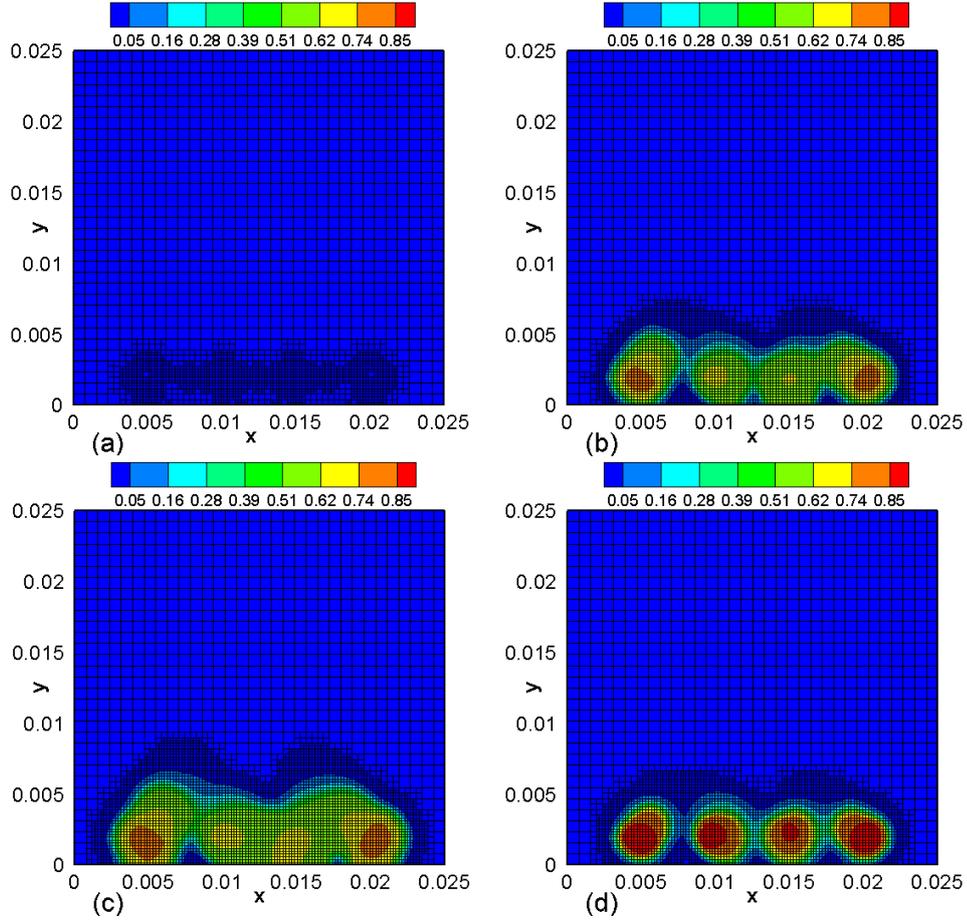


Fig. 6. Time evolution saturation contours with their corresponding adaptive grids ($S_{irr} = 0.18$) (a) at $t = 3.71 \times 10^{-5}$ sec with 2452 cells and 26126 DoFs (b) at $t = 2.4 \times 10^{-2}$ sec with 4105 cells and 42442 DoFs (c) at $t = 4.7 \times 10^{-2}$ sec with 5594 cells and 64539 DoFs (d) at $t = 7.1 \times 10^{-2}$ sec with 6758 cells and 90074 DoFs.

4 Conclusions

Motivated by a problem that fuel cell two-phase mixture models are facing, we demonstrate that our validated diffusive scheme has a powerful ability to handle highly sharp concentration gradients while maintaining accuracy with no upwinding. The diffusive scheme that we adapt here is believed to solve the problem in the field of two-phase mixture model of PEMFC. In addition to this, this numerical combination [1,2] we present includes an adaptive operator splitting with an a posteriori criterion that avoids solving a transport

equation at every time step, block matrix preconditioning that greatly makes an improvement in the solution of linear systems, and adaptive mesh refinement that can refine and coarsen areas that are eventful. Overall, to improve the numerical efficiency in all aspects, the numerical algorithms we use in the present work can be extended to a broad range of complex transport problems.

5 Acknowledgments

The work of the first and second authors was funded through the Canada Research Chairs Program and the MITACS Network of Centres of Excellence. Parts of the work by the third author were funded through Award No. KUS-C1-016-04, made by the King Abdullah University of Science and Technology, and through an Alfred P. Sloan Research Fellowship. Some computational resources used here were also funded by NSF award DMS-0922866.

Thanks are due to Dr. Zhangxin Chen (University of Calgary) for his invaluable insights on finite element theory, to Dr. Sergey Mashchenko (McMaster University) for installing the relevant libraries, and to the first author's wife, Mrs. Hsiao-Yun Janette Cheng (PhD candidate, University of Sheffield), for helping with the graphics.

References

- [1] CC Chueh, M Secanell, W Bangerth, and N Djilali. Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media. *Computers and Fluids*, 39:1585–1596, 2010.
- [2] CC Chueh, W Bangerth, and N Djilali. An h-adaptive operator splitting method for two-phase flow in 3D heterogeneous porous media. *Submitted to Communications in Computational Physics*.

- [3] S Lister and N Djilali. *Two-phase transport in porous gas diffusion electrodes* (Eds. M. Faghri & B. Sunden). WIT Press, Southampton, UK, 2005.
- [4] N Djilali. Computational modelling of polymer electrolyte membrane (pem) fuel cells: challenges and opportunities. *Energy*, 32(4):269 – 280, 2007.
- [5] N Djilali and PC Sui. Transport phenomena in fuel cells: from microscale to macroscale. *International Journal of Computational Fluid Dynamics*, 22(1-2):115 – 133, 2008.
- [6] JL Guermond and R Pasquetti. Entropy-based nonlinear viscosity for fourier approximations of conservation laws. *Comptes Rendus Mathematique*, 346(13-14):801 – 806, 2008.
- [7] Y Li and W Bangerth. *The deal.II tutorial manual: step-21*. http://dealii.org/developer/doxygen/deal.II/step_21.html.
- [8] V Gurau and JA Mann Jr. A critical overview of computational fluid dynamics multiphase models for proton exchange membrane fuel cells. *SIAM Journal on Applied Mathematics*, 70(2):410 – 454, 2009.
- [9] EC Kumbur, KV Sharp, and MM Mench. Validated leverett approach for multiphase flow in pefc diffusion media I. Hydrophobicity effect. *Journal of The Electrochemical Society*, 154(12):B1295–B1304, 2007.
- [10] CC Chueh and N Djilali. Numerical analysis of two-phase flow in heterogeneous porous media using a mixture model. *Submitted to Physical Review E*.
- [11] MC Leverett. Capillary behaviour in porous solids. *AIME Trans.*, 142:152 – 169, 1941.
- [12] JH Nam and M Kaviany. Effective diffusivity and water-saturation distribution in single-and-two layer pemfc diffusion medium. *International Journal of Heat and Mass Transfer*, 46:4595 – 4611, 2003.
- [13] J Bear. *Dynamics of Fluids in Porous Media*. American Elsevier, 1972.
- [14] KS Udell. Heat transfer in porous media considering phase change and capillary - the heat pipe effect. *International Journal of Heat and Mass Transfer*, 28(2):485 – 495, 1985.

- [15] M Kaviani. *Principles of Heat Transfer in Porous Media*. Springer-Verlag, New York, 1991.
- [16] F Brezzi and M Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, 1991.
- [17] Z Chen. *Finite Element Methods and Their Applications*. Springer, 2005.
- [18] D Neckels and W Bangerth. *The deal.II tutorial manual, step-33*.
http://dealii.org/developer/doxygen/deal.II/step_33.html.
- [19] GF Carey. *Computational Grids: Generation, Adaptation and Solution Strategies*. Taylor & Francis, 1997.
- [20] W Bangerth, R Hartmann, and G Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [21] Y Saad and MH Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [22] F Zhang. *The Schur Complement and its Applications*. Springer, 2005.
- [23] D Silvester and A Wathen. Fast iterative solution of stabilised Stokes systems part II: Using general block preconditioners. *SIAM Journal on Numerical Analysis*, 31(5):1352–1367, 1994.
- [24] GH Golub and CF van Loan. *Matrix Computations*. 3rd Edition, Johns Hopkins, 1996.
- [25] M Honarpour, L Koederitz, and AH Harvey. *Relative Permeability of Petroleum Reservoirs*. First edition, CRC Press, 1986.
- [26] RA Wooding and HJ Morel-Seytoux. Multi-phase fluid flow through porous media. *Annual Review of Fluid Mechanics*, 8:233 – 274, 1976.

Table 1

Parameters used in the model

PARAMETER	SYMBOL	VALUE	UNIT
Porosity	ϵ	0.6	-
Air viscosity (wetting)	μ_w	2.075×10^{-4}	$g \cdot cm^{-1} \cdot sec^{-1}$
Water viscosity (nonwetting)	μ_{nw}	3.55×10^{-3}	$g \cdot cm^{-1} \cdot sec^{-1}$
Immobile or irreducible saturation	S_{irr}	0.18	-
Surface tension	σ_c	10^{-4} or 10^{-2}	$g \cdot sec^{-2}$
PTFE weight percentage	$wt\%$	5	-
Number of high-permeability centres	N_p	15	-
Permeability magnitude scaling factor	γ_m	10^{-7}	-
Band width	B_w	0.0013	-
Number of source/sink terms	N_s	4	-
The i -th wetting-phase source/sink intensity	$q_w^{(i)}$	100	sec^{-1}
Stabilization exponent	α	1.0	-
Stabilization constant	β	0.03	-
Normalization constant	c_R	1.0	-
Operator splitting threshold	θ^*	2.0	-

Table 2

Parameters used in the Buckley-Leverett problem

PARAMETER	SYMBOL	VALUE	UNIT
Porosity	ϵ	1.0	-
Air viscosity (wetting)	μ_w	2.075×10^{-4}	$g \cdot cm^{-1} \cdot sec^{-1}$
Water viscosity (nonwetting)	μ_{nw}	3.55×10^{-3}	$g \cdot cm^{-1} \cdot sec^{-1}$
permeability (constant)	k	0.1	cm^2
Immobile or irreducible saturation	S_{irr}	0.0	-
Stabilization exponent	α	1.0	-
Stabilization constant	β	0.38	-
Normalization constant	c_R	1.0	-
Operator splitting threshold	θ^*	1.0	-

Bibliography

- [1] SE Buckley and MC Leverett. Mechanism of fluid displacements in sands. *AIME Trans.*, 146:107 – 116, 1942.
- [2] RA Wooding and HJ Morel-Seytoux. Multi-phase fluid flow through porous media. *Annual Review of Fluid Mechanics*, 8:233 – 274, 1976.
- [3] CC Chueh, M Secanell, W Bangerth, and N Djilali. Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media. *Computers and Fluids*, 39:1585 – 1596, 2010.
- [4] J Larminie and A Dicks. *Fuel Cell Systems Explained*. Wiley, 2006.
- [5] MM Mench. *Fuel Cell Engines*. Wiley, 2008.
- [6] J Bear. *Dynamics of Fluids in Porous Media*. American Elsevier, 1972.
- [7] A Scheidegger. *The Physics of Flow Through Porous Media*. University of Toronto Press, Toronto and Buffalo, 3rd Edition, 1974.
- [8] R Helmig. *Multiphase Flow and Transport Processes in the Subsurface*. Springer-Verlag, Heidelberg, 1997.
- [9] S Lister and N Djilali. *Two-phase transport in porous gas diffusion electrodes (Eds. M. Faghri & B. Sunden)*. WIT Press, Southampton, UK, 2005.
- [10] M von Paul. *Simulation of Two-phase Flow Processes in Heterogeneous Porous Media with Adaptive Methods*. Dissertation, Universität Stuttgart, Institut für Wasserbau, 2003.
- [11] MC Leverett. Capillary behaviour in porous solids. *AIME Trans.*, 142:152 – 169, 1941.

- [12] JH Nam and M Kaviany. Effective diffusivity and water-saturation distribution in single-and-two layer pemfc diffusion medium. *International Journal of Heat and Mass Transfer*, 46:4595 – 4611, 2003.
- [13] V Gurau and JA Mann Jr. A critical overview of computational fluid dynamics multiphase models for proton exchange membrane fuel cells. *SIAM Journal on Applied Mathematics*, 70(2):410 – 454, 2009.
- [14] M Honarpour, L Koederitz, and AH Harvey. *Relative Permeability of Petroleum Reservoirs*. First edition, CRC Press, 1986.
- [15] L Lake. *Enhanced Oil Recovery*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1989.
- [16] RJ LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser-Verlag, Basel, 1992.
- [17] R Helmig. *Theorie und Numerik der Mehrphasenströmungen in geklüftet-porösen Medien*. Dissertation, Universität Hannover, Institut für Strömungsmechanik und Elektron. Rechnen im Bauwesen, 1993.
- [18] C Hirsch. *Numerical Computation of Internal and External Flows – Volume 1: Fundamentals of Numerical Discretization*. John Wiley and Sons, 1988.
- [19] OC Zienkiewicz, RL Taylor, and P Nithiarasu. *The Finite Element Method for Fluid Dynamics*. Kindle Edition, Elsevier, 2005.
- [20] A Ern and JL Guermond. *Theory and Practice of Finite Elements*. Springer, 2004.
- [21] Z Chen. *Finite Element Methods and Their Applications*. Springer, 2005.
- [22] GH Golub and JM Ortega. *Scientific Computing and Differential Equations – An Introduction to Numerical Methods*. Academic Press, 1992.
- [23] V Komkov. *Variational principles of continuum mechanics with engineering applications: Critical points theory*, volume 1. D. Reidel Publishing Co., Dordrecht, 1986.

- [24] JL Guermond and R Pasquetti. Entropy-based nonlinear viscosity for fourier approximations of conservation laws. *Comptes Rendus Mathematique*, 346(13-14):801 – 806, 2008.
- [25] JH Ferziger and M Perić. *Computational Methods for Fluid Dynamics*. Springer, Berlin, Heidelberg, New York, 2002.
- [26] M Ainsworth and JT Oden. *A Posteriori Error Estimation in Finite Element Analysis*. John Wiley and Sons, 2000.
- [27] R Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques*. Wiley/Teubner, New York, Stuttgart, 1996.
- [28] GF Carey. *Computational Grids: Generation, Adaptation and Solution Strategies*. Taylor & Francis, 1997.
- [29] W Bangerth and R Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser Verlag, 2003.
- [30] P Ellsiepen. *Zeit- und ortsadaptive Verfahren angewandt auf Mehrphasenprobleme poröser Medien*. Dissertation, Universität Stuttgart, Institut für Mechanik, Lehrstuhl II, 1999.
- [31] R Hinkelmann. *Efficient Numerical Methods and Information-Processing Techniques in Environment Water*. Habilitation, Universität Stuttgart, 2002.
- [32] H Tang and T Tang. Adaptive mesh methods fo one-and two-dimensional hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 41(2):487, 2003.
- [33] Z Tan, Z Zhang, Y Huang, and T Tang. Moving mesh methods with locally varying time steps. *Journal of Computational Physics*, 200(1):347 – 367, 2004.
- [34] Z Tan, T Tang, and Z Zhang. A simple moving mesh method for one- and two-dimensional phase-field equations. *Journal of Computational and Applied Mathematics*, 190(1-2):252 – 269, 2006.
- [35] Y Di, R Li, T Tang, and P Zhang. Moving mesh finite element methods for the incompressible navier–stokes equations. *SIAM Journal on Scientific Computing*, 26(3):1036–1056, 2005.

- [36] Y Di, R Li, T Tang, and P Zhang. Moving mesh methods for singular problems on a sphere using perturbed harmonic mappings. *SIAM Journal on Scientific Computing*, 28(4):1490 – 1508, 2006.
- [37] Y Di, R Li, and T Tang. A general moving mesh framework in 3d and its application for simulating the mixture of multi-phase flows. *Communications in Computational Physics*, 3:582 – 602, 2008.
- [38] R Li and T Tang. Moving mesh discontinuous galerkin method for hyperbolic conservation laws. *Journal of Scientific Computing*, 27:347 – 363, 2006.
- [39] YB Zhang, H Wang, and T Tang. Simulating two-phase viscoelastic flows using moving finite element methods. *Communications in Computational Physics*, 7:333 – 349, 2010.
- [40] MJ Berger and P Colella. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, 82(1):64 – 84, 1989.
- [41] W Bangerth, R Hartmann, and G Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [42] I Babuška and WC Rheinboldt. A-posteriori error estimates for the finite element method. *International Journal of Numerical Methods in Engineering*, 12(4):1597–1615, 1978.
- [43] I Babuška and WC Rheinboldt. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis*, 15(4):736–754.
- [44] C Johnson and P Hansbo. Adaptive finite element methods in computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 101(1-3):143 – 181, 1992.
- [45] OC Zienkiewicz and JZ Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357, 1987.
- [46] C Johnson. Adaptive finite element methods for diffusion and convection problems. *Computer Methods in Applied Mechanics and Engineering*, 82(1-3):301 –

- 322, 1990. Proceedings of the Workshop on Reliability in Computational Mechanics.
- [47] A Papastavrou. *Adaptive Finite Element Methoden für Konvektions-Diffusionsprobleme*. Dissertation, Ruhr-Universität Bochum, 1998.
- [48] C Barlag. *Adaptive Methoden zur Modellierung von Stofftransport im Kluftgestein*. Dissertation, 1997.
- [49] E Godlewski and PA Raviart. *Numerical Approximation of Hyperbolic Systems of Conservation Laws*. Springer-Verlag, New York, 1996.
- [50] H Holden and NH Risebro. *Front Tracking for Conservation Laws*. Lecture Notes, Department of Mathematics, Norwegian University of Science and Technology, 1997.
- [51] D Kröner. *Numerical Schemes for Conservation Laws*. Wiley/Teubner, 1997.
- [52] KW Morton. *Numerical Solution of Convection-Diffusion Problems*. Chapman & Hall, London, 1996.
- [53] EF Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer-Verlag, Berlin, 1997.
- [54] JT Beale, C Greengard, and E Thomann. *Operator splitting for Navier-Stokes and Chorin-Marsden product formula, In Vortex Flows and Related Numerical Methods (Greenoble)*. Kluwer Acad. Publ., Dordrecht, 1993.
- [55] R Glowinski and O Pironneau. Finite element methods for navier-stokes equations. *Annual Review of Fluid Mechanics*, 24(1):167–204, 1992.
- [56] GE Karniadakis and RD Henderson. *Spectral element methods for incompressible flows, In RW Johson, Editor, The Handbook of Fluid Dynamics*, volume 29. Kluwer Acad. Publ., Dordrecht, 1998.
- [57] O Pironneau. On the transport-diffusion algorithm and its applications to the navier-stokes equations. *Numerische Mathematik*, 38(3):309–332, 1982.
- [58] XC Tai and P Neittaanmäki. Parallel finite element splitting-up method for parabolic problems. *Numerical Methods for Partial Differential Equations*, 7(3):209–225, 1991.

- [59] MS Espedal and RE Ewing. Characteristic petrov-galerkin subdomain methods for two-phase immiscible flow. *Computer Methods in Applied Mechanics and Engineering*, 64(1-3):113 – 135, 1987.
- [60] HK Dahle. *Operator Splitting Techniques for Convection-Dominated Diffusion Problems in One and Two Space Dimensions*. PhD Thesis, Department of Mathematics, University of Bergen, Norway, 1988.
- [61] MS Espedal and KH Karlsen. *Numerical Simulation of Reservoir Flow Models Based on Large Time Step Operator Splitting Algorithms*, volume 1734. Lecture Notes in Mathematics (Springer-Verlag, New York), 2000.
- [62] KH Karlsen, KA Lie, JR Natvig, HF Nordhaug, and HK Dahle. Operator splitting methods for systems of convection-diffusion equations: Nonlinear error mechanisms and correction strategies. *Journal of Computational Physics*, 173(2):636 – 663, 2001.
- [63] GA Cederberg, RL Street, and JO Leckie. A groundwater mass transport and equilibrium chemistry model for multicomponent systems. *Water Resources Research*, 21(8):1095–1104, 1985.
- [64] AL Walter, EO Frind, DW Blowes, CJ Ptacek, and JW Molson. Modeling of multicomponent reactive transport in groundwater 1. model development and evaluation. *Water Resources Research*, 30(11):3137–3148, 1994.
- [65] A Zysset, F Stauffer, and T Dracos. Modeling of chemically reactive groundwater transport. *Water Resources Research*, 30(7):2217–2228, 1994.
- [66] LSJ Bell and PJ Binning. A split operator approach to reactive transport with the forward particle tracking eulerian lagrangian localized adjoint method. *Advances in Water Resources*, 27(4):323 – 334, 2004.
- [67] AI James and JW Jawitz. Modeling two-dimensional reactive transport using a godunov-mixed finite element method. *Journal of Hydrology*, 338(1-2):28 – 41, 2007.
- [68] Z Chen. *Computational Methods for Multi-phase Flows in Porous Media*. SIAM, 2006.

- [69] Z Chen, G Huan, and B Li. An improved impes method for two-phase flow in porous media. *Transport in Porous Media*, 54:361 – 376, 2004.
- [70] E Abreu, J Douglas, F Furtado, and F Pereira. Operating splitting for three-phase flow in heterogeneous porous media. *Communications in Computational Physics*, 6:72 – 84, 2009.
- [71] GH Golub and CF van Loan. *Matrix Computations*. 3rd Edition, Johns Hopkins, 1996.
- [72] Y Saad and MH Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [73] JL Lions, Y Maday, and G Turinici. Réolution d’edp par un schéma en temps pararéel. *Comptes Rendus de l’Académie des Sciences - Series I - Mathematics*, 332(7):661 – 668, 2001.
- [74] C Farhat and M Chandesris. Time-decomposed parallel time-integrators: theory and feasibility studies for fluid, structure, and fluid-structure applications. *International Journal for Numerical Methods in Engineering*, 58(9):1397–1434, 2003.
- [75] B Guillaume and Y Maday. *A Parareal time discretization for Non-linear PDEs with Application to the Pricing of an American Put*, volume 23. Recent Developments in Domain Decomposition Methods of Lecture Notes in Computational Science and Engineering, Springer, 2002.
- [76] I Garrido, MS Espedal, and GE Fladmark. *A Convergent Algorithm for Time Parallelization Applied to Reservoir Simulation*, volume 40. Domain Decomposition Methods in Science and Engineering of Lecture Notes in Computational Science and Engineering, Springer Berlin Heidelberg, 2006.