

Numerical aspects of transport modelling in Enhanced Oil Recovery

L. Wiegman

Numerical aspects of transport modelling in Enhanced Oil Recovery

by

L. Wiegman

in partial fulfilment of the requirements for the degree of

Master of Science
in
Applied Mathematics

at the Delft University of Technology,
to be defended publicly on Friday June 7, 2017 at 16:00 PM.

Student number: 1527126
Project duration: September 1, 2016 – July 7, 2017
Thesis committee: Dr. ir. J. E. Romate, TU Delft & Shell Global Solutions International (supervisor)
Prof. dr. ir. C. Vuik, TU Delft
Dr. P. Wilders, TU Delft

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Abstract

Since the 1950s, numerical models are widely used in reservoir simulators to predict and optimize oil recovery from petroleum reservoirs. Commercial simulators typically combine multiphase porous media flow models with a separate module for the transport of tracers. By decoupling flow and transport, the transport equation can be solved using a more accurate and efficient numerical scheme than the fully implicit first order scheme that is commonly used to solve the flow equations. The accuracy of the numerical transport scheme is of high importance in Enhanced Oil Recovery modelling in order to accurately predict the influence of EOR techniques on the oil production. Of all numerical methods considered in this research, the explicit high-resolution flux-limiter method with the van Leer limiter performed best in terms of accuracy and efficiency. The accuracy and monotonicity of the numerical transport scheme strongly depend on the underlying flow solution. To ensure monotonicity of the scheme in all model situations considered, a partially implicit method is introduced that switches to the monotone first order implicit scheme where necessary.

In addition, care must be taken when modelling influences of the polymer and surfactant concentration on the flow. Bad choices can lead to instabilities or cause deviations in the model outcome. The conventional approach for modelling the hydrodynamic acceleration of polymer via a constant velocity enhancement factor results in an ill-posed system. To obtain a well-posed system, some form of a saturation dependent velocity enhancement factor could be used instead. However, which physical and numerical model is best suited for modelling the hydrodynamic acceleration remains an open question.

When modelling surfactant flooding, the implementation of a discontinuous transition in relative permeabilities around the surfactant front results in oscillations in the numerical solution over time. These oscillations can be resolved or at least diminished by applying some form of interpolation between the two sets of relative permeabilities. However, there is no guarantee that the oscillations disappear completely and the interpolation function has to be carefully selected as it can have a large impact on the solution profile.

Acknowledgements

This thesis has been submitted for the degree Master of Science in Applied Mathematics at Delft University of Technology. Research for this thesis took place during an internship at the Enhanced Oil Recovery team of Shell Global Solutions International. I would like to thank my supervisor Johan Romate, for giving me the opportunity to conduct my research at Shell and for all his advice and patience throughout this thesis. Also I would like to thank all my colleagues and fellow interns at Shell for their contribution to my project and the great experience I had during my internship.

*L. Wiegman
Delft, June 2017*

Contents

List of Figures	ix
List of Tables	xi
1 Introduction	1
1.1 Enhanced oil recovery	1
1.2 Reservoir simulation	1
1.2.1 Matlab Reservoir Simulation Toolbox	2
1.3 Structure of report	2
2 Two-phase flow in porous media	5
2.1 Fractional flow equation	6
2.2 Analytical solution in 1D	7
2.3 Numerical solution	8
2.3.1 Fully implicit discretization	9
2.3.2 Fractional flow discretization	11
2.3.3 Accuracy, stability and efficiency.	11
2.3.4 Extension to two dimensions	14
3 Transport of passive tracers	15
3.1 Analytical solution	15
3.2 Numerical solution	16
3.2.1 First order methods	17
3.2.2 High-resolution methods	19
3.2.3 Conclusions	22
3.3 Extension to two dimensions	23
3.4 Nonconservative methods	24
4 Monotonicity	25
4.1 Monotonicity analysis.	25
4.1.1 Explicit scheme	26
4.1.2 Semi-implicit scheme	28
4.1.3 Implicit scheme	29
4.1.4 High-resolution schemes	30
4.2 Monotonicity preserving scheme	30
4.2.1 Accuracy, stability and efficiency.	31
4.2.2 Extension to two dimensions	32
5 Transport of polymer	35
5.1 Model equations	36
5.2 Analytical solution	37
5.3 Numerical solution	39
5.3.1 First order methods	39
5.3.2 High-resolution methods	40
5.3.3 Accuracy, stability and efficiency.	43
5.4 Hydrodynamic acceleration.	44
5.4.1 Constant factor	44
5.4.2 Saturation dependent factor	46

6	Transport of surfactant	51
6.1	Model equations	52
6.2	Analytical solution	53
6.3	Numerical solution	57
6.4	Analysis oscillations.	59
6.4.1	Interpolation.	62
6.4.2	Illustration on typical SP flooding sequence	64
6.4.3	Conclusion.	65
7	Conclusions and recommendations	67
7.1	Conclusions.	67
7.2	Further remarks and recommendations.	68
A	Deriving analytical solutions	69
A.1	Passive tracer	69
A.2	Polymer	70
A.3	Surfactant.	71
B	Monotonicity analysis in 2D	73
	Bibliography	75

List of Figures

1.1	Schematic reservoir setup in 1D and 2D with one injection well (I) and one production well (P).	2
2.1	Typical two-phase relative permeabilities and the corresponding fractional flow function.	7
2.2	Analytical solution of two-phase porous media flow with Riemann initial data for certain time $t > 0$.	8
2.3	Schematic illustration of the construction of a physically admissible solution. S^* denotes the shock saturation.	8
2.4	Buckley-Leverett solution at a certain time $t > 0$ with $S_o(x, t)$ shown in red and $S_w(x, t)$ in blue.	8
2.5	Grid.	9
2.6	Log-log plot of the error versus the amount of grid cells N showing that the flow solver has a convergence rate smaller than one.	13
2.7	Solution of the fully implicit flow solver at a certain time t for different values of Δx and Δt .	13
3.1	Analytical solution of a passive tracer in two-phase porous media flow with Riemann initial data for certain time $t > 0$.	17
3.2	Numerical solution of $S_w(x, t)$ and $c(x, t)/\bar{c}$ for different first order methods at a certain time $t > 0$.	18
3.3	Most widely used TVD limiters. The superbee limiter, $\Phi(\theta) = \max(0, \min(1, 2\theta), \min(2, \theta))$, is shown in red and the van Leer limiter, $\Phi(\theta) = \frac{\theta + \theta }{1 + \theta }$, is shown in blue.	20
3.4	Numerical solution of $S_w(x, t)$ and $c(x, t)/\bar{c}$ for three different high-resolution methods at a certain time $t > 0$.	21
3.5	Log-log plot of the 1-norm error versus the amount of grid cells N for the first order semi-implicit method (circles) and the high-resolution semi-implicit method (triangles).	22
4.1	Numerical solution of $c(x, t)/\bar{c}$ for three different high-resolution methods at a certain time $t > 0$ for a low initial water saturation, $S_w^0 = S_{wc} = 0.015$.	26
4.2	Numerical solution of the water saturation $S_w(x, t)$ using an explicit method (blue) and an implicit method (red) together with the analytical solution (black).	27
4.3	Ratio between the explicit approximation $S_{w,i}^n$ and the implicit approximation $S_{w,i}^n$.	27
4.4	Partially-implicit approximation of $c(x, t)/\bar{c}$ at a certain time $t > 0$ for a low initial water saturation, $S_w^0 = S_{wc} = 0.015$.	32
4.5	Partially-implicit approximation of $c(x, t)/\bar{c}$ at a certain time $t > 0$ for a high initial water saturation, $S_w^0 = S_{wc} = 0.15$.	32
4.6	Log-log plot of the 1-norm error versus the amount of grid cells N for the partially implicit method with $S_w^0 = 0.15$ (solid) and $S_w^0 = 0.015$ (dashed).	32
5.1	Production profiles for a water flood and a polymer flood showing the benefit of the polymer flood at the economical limit [21].	35
5.2	Fractional flow curves for water without polymer ($c_p = 0$) and water with a fixed polymer concentration \bar{c}_p .	36
5.3	Typical viscosity multiplier function $\mu_{\text{mult}}(c_p)$.	37
5.4	Analytical solution of polymer flooding in secondary mode for certain time $t > 0$.	38
5.5	Analytical solution of polymer flooding in tertiary mode for certain time $t > T$.	38
5.6	Production profiles for a polymer flood using an explicit first order method and an explicit high-resolution method compared to the production profile of a waterflood.	40
5.7	First order numerical solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 50 cells and time steps.	41
5.8	First order numerical solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 500 cells and time steps.	41

5.9	High-resolution solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 50 cells and time steps.	42
5.10	High-resolution solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 500 cells and time steps.	42
5.11	Log-log plot of the 1-norm error versus the amount of grid cells N for the water saturation (left) and the polymer concentration (right).	44
5.12	Numerical solutions of polymer flooding in secondary mode for different constant velocity enhancement factors α using 500 grid cells. Solutions are shown in x -space (left) and in phase-space (right), where the shaded area denotes the elliptic region.	46
5.13	Numerical solutions of polymer flooding in secondary mode with a saturation dependent velocity enhancement factor using 500 grid cells and different amount of time steps.	49
6.1	Typical set of relative permeability functions for an oil-water and oil-water-surfactant system.	51
6.2	Interfacial tension (IFT) as a function of the surfactant concentration c_s , showing that the lowest IFT value is reached for a very low surfactant concentration.	51
6.3	Fractional flow functions corresponding to the relative permeability curves shown in Figure 6.1 for an oil-water system ($f_w(S_w, 0)$), an oil-water-polymer system ($f_w(S_w, \bar{c}_p)$), an oil-water-surfactant system ($f_w^{Sft}(S_w, 0)$) and an oil-water-polymer-surfactant system ($f_w^{Sft}(S_w, \bar{c}_p)$).	53
6.4	Analytical solution of SP flooding in secondary mode with $K = 0$ for certain time $t > 0$	55
6.5	Analytical solution of SP flooding in tertiary mode with $K = 0$ for certain time $t > T$	55
6.6	Analytical solution of SP flooding case 3 with $K = 0$ at a certain time $t > T$	56
6.7	Analytical solution of SP flooding case 3 with $K = 20$ for certain time $t > T$	56
6.8	Numerical solutions of $S_w(x, t)$ and $c_s(x, t)/\bar{c}_s$ at a certain time $t > 0$ for different SP flooding cases. The analytical solution of $S_w(x, t)$ is given by the solid black line and the analytical solution of $c_s(x, t)/\bar{c}_s$ is given by the dashed black line.	58
6.9	Numerical solutions of $S_w(x, t)$, $S_o(x, t)$ and $c_s(x, t)/\bar{c}_s$ over time at the end of the reservoir for different SP flooding cases showing small oscillations in the oil and water saturations over time.	58
6.10	Numerical solutions of S_w , $f_w(S_w, c_s, c_p)$ and $c_s(x, t)/\bar{c}_s$ over time at the end of the reservoir for SP flooding case 3.	59
6.11	Fractional flow functions for an oil-water-polymer system ($f_w(S_w, \bar{c}_p)$) and an oil-water-polymer-surfactant system ($f_w^{Sft}(S_w, \bar{c}_p)$).	60
6.12	Solution profile at time t^n with a discontinuous switch in fractional flow function from cell I to cell $I + 1$	60
6.13	Contour plot of $F(S_w, I, S_w, I+1)$	61
6.14	Contour plot of F with solution profiles for a coarse grid (left) and a fine grid (right) showing how the oscillations arise in the water saturation over time and that the period of the oscillations depends on Δx	62
6.15	Extended relative permeability functions for the oil-water and oil-water-surfactant system to be used for interpolation.	63
6.16	Three popular interpolation functions $m(c_{sft})$: a linear function (blue), a sine shaped function (red) and a tangent hyperbolic shaped function (yellow).	63
6.17	Effect of three different interpolation functions $m(c_{sft})$ on the fractional flow function.	63
6.18	Numerical solutions of S_w , $f_w(S_w, c_s, c_p)$ and $c_s(x, t)/\bar{c}_s$ obtained through linear interpolation between the relative permeabilities.	64
6.19	Numerical solutions at $x = L$ of the oil and water fluxes ($f_w(S_w, c_s, c_p)$ and $1 - f_w(S_w, c_s, c_p)$), the normalized polymer concentration ($c_p(x, t)/\bar{c}_p$) and the normalized surfactant concentration in water ($c_s(x, t)/\bar{c}_s$) for different transitions in relative permeabilities.	65

List of Tables

2.1	CPU-times of the flow solver for for increasing amount of grid cells with constant ratio $\Delta t/\Delta x$. . .	13
3.1	CPU-times of the semi-implicit first order, semi-implicit high-resolution and fully implicit high-resolution method for increasing amount of grid cells.	22
4.1	CPU-times of the semi-implicit high-resolution, partially implicit and implicit high-resolution method for increasing amount of grid cells.	31
5.1	CPU-times of all numerical methods applied to polymer flooding for increasing amount of grid cells.	43
6.1	Injection scheme of a typical SP flooding sequence in terms of injected pore volumes (PV). . . .	64

Nomenclature

Abbreviations

AD	Automatic differentiation
ADI	Alternating direction implicit
ASP	Alkaline surfactant polymer
BL	Buckley-Leverett
DCU	Donor cell upwind
EOR	Enhanced oil recovery
FV	Finite volume
IFT	Interfacial tension
MRST	Matlab Reservoir Simulation Toolbox
PDE	Partial differential equation
PV	Pore volume
SF	Surfactant foam
SP	Surfactant polymer
TV	Total variation

Symbols

α	Velocity enhancement factor
β	Degree of implicitness
ϵ	Tolerance of nonlinear solver
λ	Mobility
μ	Viscosity
μ_{mult}	Viscosity multiplier function
Φ	Flux limiter function
ϕ	Porosity
ρ	Density
σ	Speed of shock wave or discontinuity
A	Area
C	Courant number
c	Tracer concentration
c_{sft}^*	Surfactant concentration at which the lowest IFT value is reached

c_{sft}	Average surfactant concentration
D	Discriminant
E	Error
F	Flux function
f	Fractional flow function
f^{sft}	Fractional flow function for an oil-water-surfactant system
G	Flux function
J	Jacobian matrix
K	Absolute permeability
K	Partitioning coefficient
k_r^0	Endpoint of the relative permeability for a Corey relative permeability
k_r	Relative permeability
L	Length of the reservoir
M	Mobility ratio
m	Interpolation function
N	Total amount of grid blocks
n	Corey coefficient
p	Pressure
P_c	Capillary pressure
q	Source term
R	Residual
S	Saturation
S^*	Threshold saturation
S_{or}	Residual oil saturation
S_{or}^{sft}	Residual oil saturation for an oil-water-surfactant system
S_{wc}	Connate water saturation
S_{wc}^{sft}	Connate water saturation for an oil-water-surfactant system
T	Period of oscillations
u	Darcy velocity
V	Volume
v	Actual velocity

Subscripts

α	Phase α , either o for oil or w for water
c	Grid cell

H	Higher order
i	Index of grid block
L	Lower order
o	Oil phase
p	Polymer
s	Surfactant
T	Total, i.e. the sum over all phases
t	Partial differentiation with respect to time
w	Water phase
x	Partial differentiation with respect to space

Superscripts

inj	Injection value
p	Order of accuracy



Introduction

Mathematical models have been used since the late 19th century to describe the flow of fluids in petroleum reservoirs. Since real fields are too complicated to be described exactly, many simplifications have to be made to develop such a model. Solving the resulting model analytically is usually not possible, so numerical methods are used to find and approximate solutions. Since the 1950s, these numerical models are widely used in reservoir simulators to predict and optimise oil recovery from petroleum reservoirs. The more the numerical models resemble the actual behaviour in the reservoir, the more reliable these predictions become, so the need for accurate models and sophisticated computational tools is eminent.

1.1. Enhanced oil recovery

In the very early stage of oil recovery, the pressure in the reservoir is so high that oil is produced through the created wells by natural decompression. This stage is referred to as primary recovery. It ends when the pressure in the field is too low for flow against the pressure in the atmosphere. At this time usually 70-85% of the hydrocarbons are still left in the reservoir. To maintain reservoir pressure and recover part of the remaining oil, water or gas is injected into some wells (injection wells) while oil is produced through other wells (production wells). This stage is called secondary recovery, or water flooding in case the injected fluid is water. The efficiency of a water flood depends on many factors such as the oil viscosity and rock characteristics. In the case of very viscous oil, the water is extremely mobile compared to the oil and production wells primarily produce water instead of oil. Also when a large amount of the oil is trapped in small pores, water is not able to wash it out. Often 50% or more of the hydrocarbons remain in the reservoir after water flooding. Any technique applied after secondary recovery is referred to as tertiary recovery.

Throughout all three phases enhanced oil recovery techniques can be used to recover more hydrocarbons from the reservoir. Enhanced oil recovery (EOR) is oil recovery by injection of materials not normally present in petroleum reservoirs [12]. Several categories of EOR techniques exist of which chemical flooding is an important one. Here chemicals such as alkaline, surfactant, polymer and/or foam are added to reduce the fluid mobility and hereby improve the efficiency of the flood. The introduction of these techniques was accompanied by the need for tools capable of modelling the complex physical phenomena as well as solving the resulting sharply changing fluid interfaces associated with these techniques [7].

1.2. Reservoir simulation

Reservoir simulators typically use multiphase porous media flow models to describe the flow of a hydrocarbon gaseous phase, an aqueous phase and a hydrocarbon liquid phase through porous rock. These models consist of coupled, nonlinear, time-dependent PDE's. An important problem in reservoir simulation is how to develop a robust, stable, efficient and accurate solution scheme. The most common scheme applied in commercial simulators is the fully implicit scheme, which solves all flow equations simultaneously and implicitly with the advantage of producing a very stable scheme. Compared to explicit or partially implicit methods it is however computationally more expensive and therefore not suited for problems that involve many components [8]. In this light several commercial simulators contain a separate module for describing the transport of components, an idea that originated in tracer flow modelling [19]. By decoupling flow and

transport, one not only gains efficiency but also the possibility to solve the transport using a more accurate numerical scheme.

To perform the transport calculations, the module requires parameters from the flow simulator at each time step. This communication, or coupling, between the separate transport module and the flow simulator is one of the research subjects in this thesis. Especially in situations where a component influences the flow, the coupling becomes important and bad choices can lead to instabilities or cause deviations in the model outcome. These outcomes are used to predict the influence of components on the oil production, so errors in the model or its numerical solution could lead to false predictions which in turn can lead to economic losses. Having a correct description and accurate solution of the transport is therefore of high importance in EOR simulation. This leads to the next two research subjects of this thesis: to have a closer look at the terms in the transport equation used to model the various physical phenomena and to investigate the order of accuracy of several numerical methods used to solve these transport equations.

The overall goal of this research is to determine the best way to discretise and couple flow and transport equations in terms of robustness, stability, efficiency and accuracy.

1.2.1. Matlab Reservoir Simulation Toolbox

Since the focus is on solving the transport of components and not necessarily on solving the flow model it is coupled to, tools from the free open-source Matlab Reservoir Simulation Toolbox (MRST) are used to set up the flow solver. This toolbox contains a module with a set of fully implicit solvers based on automatic differentiation (AD) [11], a technique that enables quick prototyping of new flow models. The framework includes complex three-phase solvers with several components as well as simpler two-phase models with only an aqueous and hydrocarbon liquid phase.

MRST also contains an EOR module capable of simulating two chemical flooding techniques, polymer and surfactant injection [2], [10]. This module solves the transport of the chemicals simultaneously with the flow using a fully implicit first order numerical method. Since the focus of this research is on decoupling the flow and transport, a separate module is developed in Matlab as well and coupled to the MRST flow solver.

1.3. Structure of report

In this thesis the focus is only on very simple two-phase flow models. After primary recovery water is injected with one or more special components on one end of the reservoir (injection well), while oil is produced at the other end (production well). The reservoir is assumed to be homogeneous. In one and two dimensions this simple reservoir setup is schematically given in Figure 1.1.

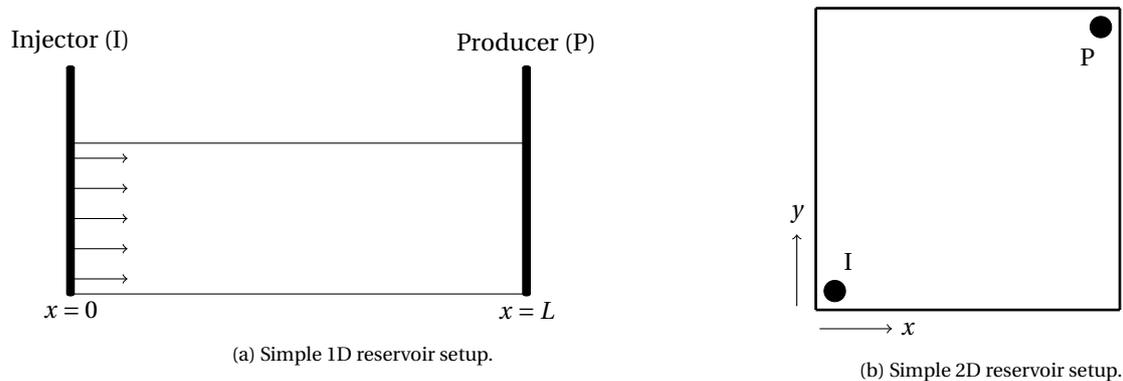


Figure 1.1: Schematic reservoir setup in 1D and 2D with one injection well (I) and one production well (P).

To be able to investigate the transport of the components, the underlying two-phase flow has to be well understood first. Hence in Chapter 2 the mathematical model describing two-phase porous media flow is derived together with its analytical and numerical solution. In Chapter 3 the model to describe the transport of a component that has no influence on the flow, the passive tracer, is developed. An analytical solution is derived and several numerical methods to solve the transport equation are discussed. Some of these numerical methods show non-monotonic behaviour which is further investigated in Chapter 4 and a new numerical method to overcome this problem is proposed. In Chapter 5 and 6 two EOR components are introduced that do influence the flow, polymer and surfactant. Their physical influence on the flow is discussed together with

ways to model these physical phenomena. Analytical solutions using fractional flow theory are derived for simplified cases. Next, the numerical methods developed in Chapter 3 and 4 are applied and evaluated in terms of accuracy, stability and efficiency. Special attention is paid to the dependency and coupling between flow and transport. Finally, the conclusions are given and discussed in Chapter 7.

2

Two-phase flow in porous media

The two-phase flow model of oil and water through a porous medium is derived, where it is assumed that no mass transfer occurs between the phases. For immiscible porous media flow the mass conservation equation for phase α is given by:

$$\frac{\partial}{\partial t}(\phi \rho_\alpha S_\alpha) + \nabla \cdot (\rho_\alpha \mathbf{u}_\alpha) = 0 \quad \text{for } \alpha \in \{o, w\}. \quad (2.1)$$

It is assumed that both fluids are incompressible as well as the rock, which results in the simplified equations

$$\phi \frac{\partial}{\partial t}(S_\alpha) + \nabla \cdot (\mathbf{u}_\alpha) = 0 \quad \text{for } \alpha \in \{o, w\}.$$

These equations are written in terms of the the phase saturations S_α , the volume fraction occupied by phase α , and the Darcy velocities \mathbf{u}_α . The porosity, which represents the ratio of accessible void volume in the rock, is denoted by ϕ . The Darcy (or superficial) velocities are related to the actual velocities \mathbf{v}_α by $\mathbf{u}_\alpha = \phi \mathbf{v}_\alpha$.

In addition to the mass conservation equations, the momentum conservation equations are defined, which are given in the form of Darcy's law. This empirically derived law describes a linear relation between the fluid velocity and the pressure gradient:

$$\mathbf{u}_\alpha = -\frac{K k_{r\alpha}}{\mu_\alpha} (\nabla p_\alpha - \rho_\alpha \mathbf{g} \nabla z).$$

The reservoir is taken horizontal, which means that the effect of gravity can be neglected. This gives the more simplified expression

$$\mathbf{u}_\alpha = -\frac{K k_{r\alpha}}{\mu_\alpha} (\nabla p_\alpha) = -\lambda_\alpha \nabla p_\alpha.$$

Here p_α denotes the phase pressure and λ_α the phase mobility, which will be addressed later on.

Substituting the Darcy velocities in the mass balance equations results in a system of two equations with four unknowns (S_w, S_o, p_w, p_o), so two additional relations are necessary to complete the model. The first relation requires that all available pore space is filled, i.e. the saturations add to one:

$$S_o + S_w = 1.$$

Secondly, the pressure difference between the phases due to the curvature and surface tension of the interface is described by the capillary pressure P_c :

$$P_c = p_o - p_w.$$

Combining all this results in the well posed system

$$\begin{cases} \phi \frac{\partial}{\partial t}(S_o) - \nabla \cdot (\lambda_o \nabla p_o) = 0 \\ \phi \frac{\partial}{\partial t}(S_w) - \nabla \cdot (\lambda_w \nabla p_w) = 0 \\ S_o + S_w = 1, \quad P_c = p_o - p_w. \end{cases}$$

In this research, the pressure difference between the phases is neglected, i.e. $P_c = 0$, which simplifies the system of equations even further to three equations with three unknowns:

$$\begin{cases} \phi \frac{\partial}{\partial t}(S_o) - \nabla \cdot (\lambda_o \nabla p) = 0 \\ \phi \frac{\partial}{\partial t}(S_w) - \nabla \cdot (\lambda_w \nabla p) = 0 \\ S_o + S_w = 1, \end{cases} \quad (2.2)$$

where $p = p_o = p_w$. To complete the two-phase flow model, boundary conditions must be imposed. On the injector side of the reservoir a constant injection rate is assumed, while at the production side the pressure is assumed constant.

2.1. Fractional flow equation

Adding the two mass conservation equations and using the fact that $S_o + S_w = 1$ results in $\nabla \cdot (\mathbf{u}_o + \mathbf{u}_w) = \nabla \cdot \mathbf{u}_T = 0$, meaning that the velocity field is divergence free. In terms of the pressure p this can be written as:

$$\nabla \cdot \mathbf{u}_T = \nabla \cdot (\mathbf{u}_o + \mathbf{u}_w) = -\nabla \cdot [(\lambda_o + \lambda_w) \nabla p] = 0. \quad (2.3)$$

By substituting this into (2.2), the pressure p can be eliminated and the so-called fractional flow equation is obtained

$$\begin{aligned} \phi \frac{\partial}{\partial t}(S_w) + \nabla \cdot \left(\frac{\lambda_w}{\lambda_w + \lambda_o} \mathbf{u}_T \right) &= 0 \\ \phi \frac{\partial}{\partial t}(S_w) + \mathbf{u}_T \cdot \nabla \left(\frac{\lambda_w}{\lambda_w + \lambda_o} \right) &= 0 \\ \phi \frac{\partial}{\partial t}(S_w) + \mathbf{u}_T \cdot \nabla (f_w(S_w)) &= 0, \end{aligned} \quad (2.4)$$

where f_w is the fractional flow function of the water phase.

In one dimension the solution to equation (2.3) is very simple, i.e. $u_T(x) = q_T$ where q_T is the injection velocity which is known from the imposed boundary condition. For the one-dimensional problem only the fractional flow equation must be solved to obtain the water saturation

$$\phi \frac{\partial}{\partial t}(S_w) + u_T \frac{\partial}{\partial x}(f_w(S_w)) = 0, \quad (2.5)$$

and the oil saturation follows from $S_o + S_w = 1$. Equation (2.5) is known as the Buckley-Leverett (BL) equation. The fractional flow $f_w(S_w)$ is an S-shaped, nonlinear function of the water saturation S_w , involving both the oil and water mobility:

$$f_w(S_w) = \frac{\lambda_w}{\lambda_w + \lambda_o} = \frac{\frac{K k_{rw}}{\mu_w}}{\frac{K k_{rw}}{\mu_w} + \frac{K k_{ro}}{\mu_o}} = \frac{k_{rw}(S_w)}{k_{rw}(S_w) + \frac{\mu_w}{\mu_o} k_{ro}(S_w)},$$

where μ_w and μ_o represent the phase viscosities. K is the absolute permeability, a rock property that describes how easily fluids can travel through the porous medium, and the relative permeability functions $k_{r\alpha}(S_w)$ correspondingly describe how easily phase α flows through the medium depending on the saturation of that phase. Relative permeability values lie between zero and one such that the phase permeability $k_\alpha = K k_{r\alpha}$ lies between zero and the absolute permeability. A popular analytic expression for two-phase relative permeability functions is proposed by Corey [12]:

$$\begin{aligned} k_{rw}(S_w) &= k_{rw}^o \left(\frac{S_w - S_{wc}}{1 - S_{wc} - S_{or}} \right)^{n_w} \\ k_{ro}(S_w) &= k_{ro}^o \left(\frac{1 - S_w - S_{or}}{1 - S_{wc} - S_{or}} \right)^{n_o} \end{aligned}$$

where S_{wc}, S_{or} are the connate water and residual oil saturations, k_{rw}^o, k_{ro}^o are the endpoints of the relative permeabilities and n_o, n_w are the Corey coefficients. The values of these parameters determine the shape of the functions and can be chosen to capture the correct pore-level displacement physics, fluid-fluid properties and rock-fluid properties. Typical relative permeability curves and the corresponding fractional flow function are shown in Figure 2.1.

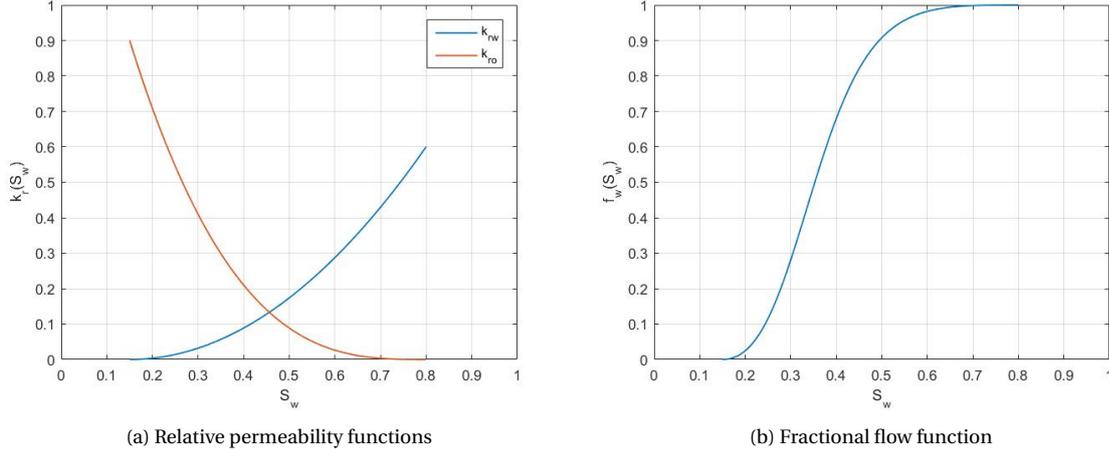


Figure 2.1: Typical two-phase relative permeabilities and the corresponding fractional flow function.

2.2. Analytical solution in 1D

Consider the nonlinear BL equation (2.5), which can be written as

$$(S_w)_t + (f(S_w))_x = 0 \quad (2.6)$$

with $f(S_w) = \frac{u_T}{\phi} f_w(S_w)$, and impose the following Riemann initial data

$$S_w(x, 0) = \begin{cases} S_w^{inj} & x < 0 \\ S_w^0 & x \geq 0. \end{cases} \quad (2.7)$$

For $u_T > 0$ the solution of this problem represents the solution in a reservoir on $x > 0$, where at $t = 0$ the water saturation is constant throughout the reservoir and from that time water is injected at $x = 0$ at a constant rate. Equation (2.6) can be written in the quasilinear form

$$\frac{\partial S_w}{\partial t} + \frac{df}{dS_w} \frac{\partial S_w}{\partial x} = \frac{\partial S_w}{\partial t} + \frac{u_T}{\phi} \frac{df_w}{dS_w} \frac{\partial S_w}{\partial x} = 0. \quad (2.8)$$

The flux function f is S-shaped, so neither convex nor concave, which means that the characteristic speed $\left(\frac{dX}{dt}\right)_{S_w} = \frac{u_T}{\phi} \frac{df_w}{dS_w}$ changes sign at the inflection point of f . Consequence is that following the characteristics fails and results in a nonphysical triple valued solution. For the relative permeabilities and fractional flow function of Figure 2.1 this nonphysical solution is shown in Figure 2.2a. As a constant initial reservoir state $S_w^0 = S_{wc} = 0.15$ is taken and pure water ($S_w = 1$) is injected, which means that $S_w^{inj} = 1 - S_{or} = 0.8$. The equal area rule is then used to replace the triple valued solution by a correct shock, see Figure 2.2b. The resulting weak solution involves both a shock and a rarefaction wave and is called a compound wave [13].

To determine if a weak solution of equation (2.5) is physically admissible, i.e. to determine the correct position of the shock, we apply the Oleinik entropy condition, which states that for all S between S_w^{inj} and S_w^0 we require:

$$\frac{f(S) - f(S_w^{inj})}{S - S_w^{inj}} \geq \sigma \geq \frac{f(S) - f(S_w^0)}{S - S_w^0}.$$

Here σ is the speed of the discontinuity, which is given by the Rankine–Hugoniot jump condition [13]:

$$\sigma = \frac{f(S_w^0) - f(S_w^{inj})}{S_w^0 - S_w^{inj}}.$$

With reference to Figure 2.2b we see that the characteristic velocity at saturation S^* is equal to the shock velocity. From the characteristics the speed left of the shock is known to be $\frac{df}{dS_w}(S^*) = \frac{u_T}{\phi} \frac{df_w}{dS_w}(S^*)$, which

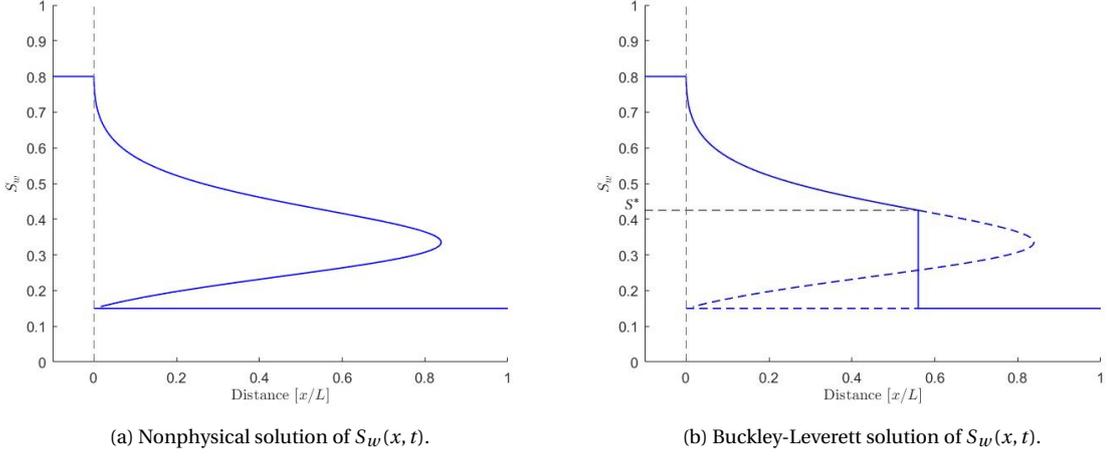


Figure 2.2: Analytical solution of two-phase porous media flow with Riemann initial data for certain time $t > 0$.

means that the saturation S^* can be found by solving

$$\frac{df_w}{dS_w}(S^*) = \sigma = \frac{f_w(S^*) - f_w(S_w^0)}{S^* - S_w^0}. \quad (2.9)$$

The solution of this equation is a straight line through $(S_w^0, f_w(S_w^0))$ with slope $\frac{df_w}{dS_w}(S^*)$, which means that in the point S^* it is tangent to the fractional flow function $f_w(S_w)$. This straight line represents the shock wave and the part of the fractional flow function from S^* to S_w^{inj} represents the rarefaction wave of the solution. A schematic illustration of this construction is shown in Figure 2.3. The resulting BL solution at a certain time t is shown in Figure 2.4.

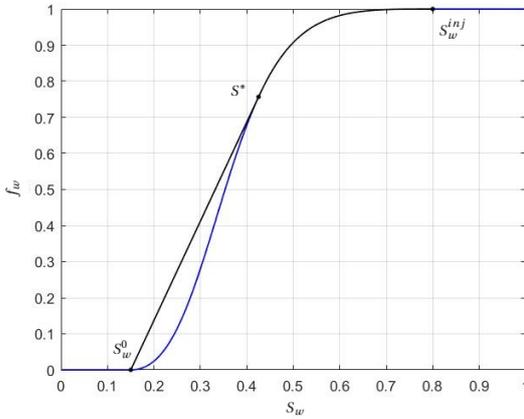


Figure 2.3: Schematic illustration of the construction of a physically admissible solution. S^* denotes the shock saturation.

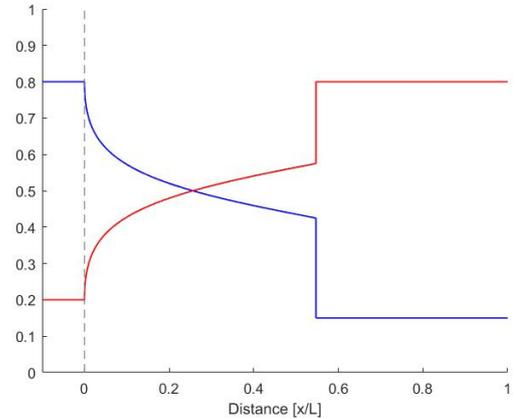


Figure 2.4: Buckley-Leverett solution at a certain time $t > 0$ with $S_o(x, t)$ shown in red and $S_w(x, t)$ in blue.

2.3. Numerical solution

The analytical solution derived in the previous section only holds for a very simplified one dimensional case. For more advanced problems a numerical method is needed to solve the system of equations. Since most contemporary commercial reservoir simulators use the compressible formulation (2.1) and solve the coupled flow equations using a fully implicit discretization, that is also the approach taken in this research. The way this approach is implemented in MRST is discussed in the next section. For analysis purposes, methods to solve the fractional flow equation (2.4) numerically are treated as well.

2.3.1. Fully implicit discretization

The fully implicit method combines an implicit method for the time discretization with a finite volume (FV) method for the spatial discretization. In 1D the reservoir is taken finite at $x \in [0, L]$ and is divided into N equally sized grid blocks c_i , with center x_i and boundaries $x_{i-1/2}$ and $x_{i+1/2}$, see Figure 2.5. A cell centered discretization is used, which means that the unknowns are calculated at the cell centers. In order to apply

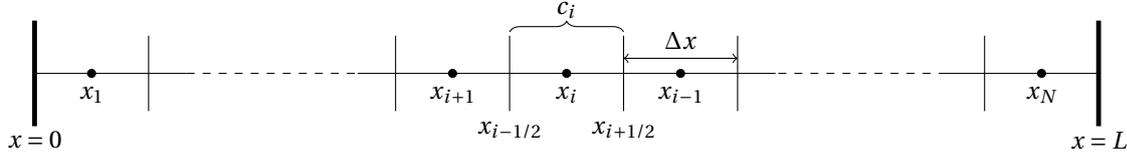


Figure 2.5: Grid.

the finite volume method a constant cross-section $A_c = \Delta y \Delta z$ is introduced such that each grid block has a volume of $V_c = \Delta x \Delta y \Delta z$.

In one dimension the compressible mass-conservation equations are given by

$$\frac{\partial}{\partial t} (\phi \rho_\alpha S_\alpha) + \frac{\partial}{\partial x} (\rho_\alpha u_\alpha) = 0 \quad \text{for } \alpha \in \{o, w\},$$

with $u_\alpha = -\lambda_\alpha \frac{\partial p_\alpha}{\partial x}$. In MRST, the phase densities are not entered into the model directly but are calculated from the surface densities $\rho_{\alpha,sc}$ using so-called formation volume factors (FVF) B_α :

$$\rho_\alpha = \frac{\rho_{\alpha,sc}}{B_\alpha}.$$

Substituting this expression and cancelling the surface density results in the simplified model equations

$$\frac{\partial}{\partial t} \left(\frac{\phi S_\alpha}{B_\alpha} \right) + \frac{\partial}{\partial x} \left(\frac{u_\alpha}{B_\alpha} \right) = 0 \quad \text{for } \alpha \in \{o, w\}. \quad (2.10)$$

Since this thesis only covers incompressible flow, the FVF will be left out of the notation when discussing the discretization.

Integrating equation (2.10) over one grid cell results in

$$\begin{aligned} \int_{V_c} \frac{\partial}{\partial t} (\phi S_\alpha) dx &= - \int_{V_c} \frac{\partial}{\partial x} (u_\alpha) dx = \int_{V_c} \frac{\partial}{\partial x} \left(\lambda_\alpha \frac{\partial p_\alpha}{\partial x} \right) dx \\ \Delta y \Delta z \int_{c_i} \frac{\partial}{\partial t} (\phi S_\alpha) dx &= \Delta y \Delta z \int_{c_i} \frac{\partial}{\partial x} \left(\lambda_\alpha \frac{\partial p_\alpha}{\partial x} \right) dx = \Delta y \Delta z \left[\lambda_\alpha \frac{\partial p_\alpha}{\partial x} \right]_{x_{i-1/2}}^{x_{i+1/2}} := F_{\alpha,i+1/2} - F_{\alpha,i-1/2}, \end{aligned}$$

where $F_{\alpha,i+1/2}$ and $F_{\alpha,i-1/2}$ represent the fluxes through the left and right cell boundaries. The pressure gradient in the flux terms is approximated by a finite difference scheme, i.e.

$$F_{\alpha,i+1/2} = \Delta y \Delta z \lambda_{\alpha,i+1/2} \frac{p_{\alpha,i+1} - p_{\alpha,i}}{\Delta x}.$$

Furthermore, the mobility at the boundary

$$\lambda_{\alpha,i+\frac{1}{2}} = \lambda_\alpha(S_{\alpha,i+1/2}) = \frac{k k_{r\alpha}(S_{\alpha,i+1/2})}{\mu_\alpha},$$

is evaluated using a single-point upstream scheme:

$$\lambda_{\alpha,i+\frac{1}{2}} = \begin{cases} \lambda_\alpha(S_{\alpha,i}) = \frac{k k_{r\alpha}(S_{\alpha,i})}{\mu_\alpha} & \text{if flow from } i \rightarrow i+1 \\ \lambda_\alpha(S_{\alpha,i+1}) = \frac{k k_{r\alpha}(S_{\alpha,i+1})}{\mu_\alpha} & \text{if flow from } i+1 \rightarrow i \end{cases}$$

Since the injector is placed at the left end of the reservoir (see Figure 1.1a), the flow is from left to right and the fluxes can thus be expressed as

$$F_{\alpha,i+1/2} = \Delta y \Delta z \lambda_{\alpha,i} \frac{p_{\alpha,i+1} - p_{\alpha,i}}{\Delta x} = \Delta y \Delta z \frac{k k_{r\alpha}(S_{\alpha,i})}{\mu_\alpha} \frac{p_{\alpha,i+1} - p_{\alpha,i}}{\Delta x}.$$

An implicit-time integration method is used to evaluate these fluxes

$$\Delta y \Delta z \int_{c_i} \frac{\partial}{\partial t} (\phi S_\alpha) dx = F_{\alpha,i+1/2}^{n+1} - F_{\alpha,i-1/2}^{n+1},$$

and the time derivative is approximated by a first order backward scheme, i.e.

$$\frac{\partial}{\partial t} (\phi S_\alpha) = \phi \frac{S_\alpha^{n+1} - S_\alpha^n}{\Delta t}.$$

Applying the midpoint rule gives

$$\Delta y \Delta z \int_{c_i} \frac{\partial}{\partial t} (\phi S_\alpha) dx \approx \Delta y \Delta z \int_{c_i} \phi \frac{S_{\alpha,i}^{n+1} - S_{\alpha,i}^n}{\Delta t} dx = \Delta x \Delta y \Delta z \phi \frac{S_{\alpha,i}^{n+1} - S_{\alpha,i}^n}{\Delta t}.$$

Combining all this, the fully implicit discretization scheme for fluid phase α is given by

$$\phi V_c \frac{S_{\alpha,i}^{n+1} - S_{\alpha,i}^n}{\Delta t} = A_c \left(\lambda_{\alpha,i}^{n+1} \frac{p_{\alpha,i+1}^{n+1} - p_{\alpha,i}^{n+1}}{\Delta x} - \lambda_{\alpha,i-1}^{n+1} \frac{p_{\alpha,i}^{n+1} - p_{\alpha,i-1}^{n+1}}{\Delta x} \right), \quad (2.11)$$

for all internal grid cells. The boundary conditions are incorporated into the expression by assuming a constant injection rate at the left boundary and a constant pressure at the right boundary.

In MRST, the primary unknowns are chosen to be S_w and p_o and the other two unknowns follow from the capillary pressure relationship and the fact that $\sum_\alpha S_\alpha = 1$. Since the mobility is a nonlinear function of the unknown saturation S_α^{n+1} , the resulting discrete system is nonlinear. To solve this system of nonlinear equations the most common method, which is the Newton method (also called Newton-Raphson method), is used in MRST.

Newton method

The discretized equations (2.11) can be written in residual form as

$$R_{\alpha,i} = \phi V_c \frac{S_{\alpha,i}^{n+1} - S_{\alpha,i}^n}{\Delta t} - A_c \left(\lambda_{\alpha,i}^{n+1} \frac{p_{\alpha,i+1}^{n+1} - p_{\alpha,i}^{n+1}}{\Delta x} - \lambda_{\alpha,i-1}^{n+1} \frac{p_{\alpha,i}^{n+1} - p_{\alpha,i-1}^{n+1}}{\Delta x} \right) = 0.$$

Rewriting this into the two unknowns S_w , for simplicity denoted as S , and p (since $p_w = p_o = p$) a system of two equations for every grid cell is obtained:

$$R_{w,i} = \phi V_c \frac{S_i^{n+1} - S_i^n}{\Delta t} - A_c \left(\lambda_{w,i}^{n+1} \frac{p_{i+1}^{n+1} - p_i^{n+1}}{\Delta x} - \lambda_{w,i-1}^{n+1} \frac{p_i^{n+1} - p_{i-1}^{n+1}}{\Delta x} \right)$$

$$R_{o,i} = \phi V_c \frac{(1 - S_i)^{n+1} - (1 - S_i)^n}{\Delta t} - A_c \left(\lambda_{o,i}^{n+1} \frac{p_{i+1}^{n+1} - p_i^{n+1}}{\Delta x} - \lambda_{o,i-1}^{n+1} \frac{p_i^{n+1} - p_{i-1}^{n+1}}{\Delta x} \right).$$

Collecting all the discrete equations, the resulting system of nonlinear equations can be written in short vector form as

$$\mathbf{R}(\mathbf{x}^{n+1}; \mathbf{x}^n) = 0,$$

where \mathbf{x}^{n+1} is the vector of unknowns, i.e.

$$\mathbf{x}^{n+1} = [S_1^{n+1}, p_1^{n+1}, S_2^{n+1}, p_2^{n+1}, \dots, S_N^{n+1}, p_N^{n+1}].$$

This system of equations is solved every time step using the following iterative scheme

$$\frac{d\mathbf{R}}{d\mathbf{x}} \delta \mathbf{x}^{k+1} = -\mathbf{R}(\mathbf{x}^k), \quad k = 0, 1, \dots$$

where $\mathbf{J} = d\mathbf{R}/d\mathbf{x}$ is the Jacobian matrix of the residual $\mathbf{R}(\mathbf{x})$ and $\delta \mathbf{x}^{k+1}$ the Newton update at iteration $k+1$. The new iteration \mathbf{x}^{k+1} is obtained from the old one by $\mathbf{x}^{k+1} = \mathbf{x}^k + \delta \mathbf{x}^{k+1}$. This process continues until $\|\mathbf{R}(\mathbf{x})\|_\infty < \epsilon$ for some given tolerance level ϵ . The final iteration is then used as the solution at the new time level, \mathbf{x}^{n+1} .

Convergence of the Newton process is highly dependent on the accuracy of the Jacobian. For complex flow models, analytical derivation of the Jacobian can be very time consuming and prone to errors. MRST circumvents this problem by using *automatic differentiation* (AD), which allows for the Jacobian to be computed for any function that is implemented as a sequence of algebraic operations [11]. The idea of this technique is that you keep track of variables and their derivatives simultaneously; all operations applied to the variable are applied in differential form to its derivative.

2.3.2. Fractional flow discretization

To solve the fractional flow function numerically, an implicit and explicit upwind method will be considered. Remember that in 1D the fractional flow function is given by

$$\phi \frac{\partial S_w}{\partial t} + u_T \frac{\partial f_w(S_w)}{\partial x} = 0.$$

An implicit-time discretization combined with a single-point upstream method then results in the following scheme

$$\begin{aligned} \phi \frac{S_i^{n+1} - S_i^n}{\Delta t} &= -u_T \frac{f_w(S_i^{n+1}) - f_w(S_{i-1}^{n+1})}{\Delta x} \\ S_i^{n+1} &= S_i^n - \frac{u_T \Delta t}{\phi \Delta x} [f_w(S_i^{n+1}) - f_w(S_{i-1}^{n+1})]. \end{aligned}$$

Similar to the fully implicit case, this method is nonlinear in the unknown S_i^{n+1} and needs to be solved using the iterative Newton method.

For an explicit-time discretization the scheme is

$$S_i^{n+1} = S_i^n - \frac{u_T \Delta t}{\phi \Delta x} [f_w(S_i^n) - f_w(S_{i-1}^n)] \quad (2.12)$$

which is linear in S_i^{n+1} and can therefore be solved directly.

2.3.3. Accuracy, stability and efficiency

For linear equations the implicit discretization scheme is unconditionally stable. On the other hand, the explicit scheme given by (2.12) is conditionally stable. A necessary condition for stability of this explicit upwind method is that [13]:

$$C = \frac{\Delta t}{\Delta x} |v_{\max}| \leq 1 \quad (2.13)$$

where v_{\max} represents the largest wave speed that is encountered. This condition is called the *CFL condition*, named after Courant, Friedrichs, and Lewy, and C is called the Courant number. The characteristic speed of the flow is given by $\frac{u_T}{\phi} f'_w(S)$, where $f'_w(S) = df_w/dS$, which results in the necessary stability condition:

$$C = \max_S \left| \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w(S) \right| \leq 1. \quad (2.14)$$

For the explicit scheme applied to the quasilinear equation (2.8) with constant f'_w (thus turning it into a linear equation), i.e.

$$S_i^{n+1} = S_i^n - \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w [S_i^n - S_{i-1}^n], \quad (2.15)$$

this CFL criterion is also sufficient to prove stability. For linear methods stability analysis is particularly easy in the 2-norm by applying von Neumann stability analysis. Substituting $S_I^n = e^{i\xi I \Delta x}$ into equation (2.15) we obtain:

$$\begin{aligned} S_I^{n+1} &= \left(1 - \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w\right) S_I^n + \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w S_{I-1}^n \\ &= \left(1 - \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w\right) e^{i\xi I \Delta x} + \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w e^{i\xi (I-1) \Delta x} \\ &= \left[\left(1 - \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w\right) + \frac{\Delta t}{\Delta x} \left(\frac{u_T}{\phi} f'_w\right) e^{-i\xi \Delta x} \right] e^{i\xi I \Delta x} \\ &= g(\xi, \Delta x, \Delta t) S_I^n. \end{aligned}$$

For the scheme to be stable it must hold that $|g(\xi, \Delta x, \Delta t)| \leq 1$ for all ξ . As ξ varies, g lies on a circle of radius $\frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w$ in the complex plane, centered on the real axis at $\left(1 - \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w\right)$. For $0 \leq \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} f'_w \leq 1$, this circle lies entirely inside the unit circle, which indeed proves stability in the 2-norm if (2.14) is satisfied.

For the nonlinear scheme (2.12) proving stability is much more difficult. For nonlinear schemes TV -stability can be used for considering stability, which requires the total variation of the method to be bounded for all n and Δt , i.e.

$$TV(S^n) = \sum_i |S_i^n - S_{i-1}^n| \leq r \quad (2.16)$$

for some constant $r > 0$. Proving that a nonlinear scheme satisfies this condition is tedious and often not even possible [13].

Although the implicit scheme is unconditionally stable for the linear case, there is no guarantee that the Newton solver described in section 2.3.1 converges to the correct solution. For too large time steps, the method may compute updates that do not point towards the correct solution. In MRST this problem is solved by a mechanism that reduces the time step if the method has not converged after a certain number of iterations. This solves the convergence problem, but also increases the amount of iterations needed per time step and thereby the computational costs per time step. To maintain efficiency it is therefore important that the time step is not taken too large. To get a sense of what too large is: Courant numbers of below five can be chosen without encountering any convergence problems. Compared to the explicit scheme, five times larger time steps can thus be taken without having stability or convergence problems.

The increase in stability obtained by the implicit method is gained at the cost of the accuracy of the method. Applying truncation error analysis to the quasi-linear equation (2.8) results in the following expression for the truncation error for the fully implicit discretization [20]

$$\tau^{n+1} = \left[\frac{\Delta t}{2} + U_w \frac{\Delta x}{2} \right] S_{tx} + \mathcal{O}(\Delta^3),$$

where $U_w = \frac{\phi}{u_T f_w}$. For the explicit approximation this expression is slightly different:

$$\tau^{n+1} = \left[-\frac{\Delta t}{2} + U_w \frac{\Delta x}{2} \right] S_{tx} + \mathcal{O}(\Delta^3).$$

Both methods are first order accurate, but the truncation error of the implicit method grows monotonically with $\Delta t, \Delta x$ and exceeds the error of the explicit method. Note that when $\Delta t = U_w \Delta x$, the first order error term in the explicit method disappears completely and a higher accuracy is obtained.

The local truncation errors are derived using Taylor series expansion, which assume smoothness of the underlying solution. Around discontinuities, like the shock in Figure 2.4, these approximations do not hold and the accuracy is even less than first order [13]. The formal order of accuracy of the method can be determined by looking at the behaviour of the error at some fixed time t_n :

$$\|E\|_1 = \Delta x \sum_{i=1}^N |E_i| = \Delta x \sum_{i=1}^N |S_i^n - S(x_i, t_n)|.$$

For a method with order of accuracy p this error is expected to behave like

$$\|E\|_1 = c(\Delta x)^p$$

as $\Delta x \rightarrow 0$ for some constant c . On a log-log scale the error then behaves linearly with a slope equal to the order of accuracy:

$$\log|E| \approx \log|c| + p \log|\Delta x|.$$

Looking at Figure 2.6, it can thus be concluded that the formal accuracy of the flow solver is indeed less than first order. While refining the grid, the ratio $\Delta t/\Delta x$ was kept constant and the Courant number was around one. By allowing the ratio $\Delta t/\Delta x$ to change, the convergence in Δt and Δx can be viewed separately as well. As shown in Figure 2.7, the convergence in Δx goes faster than in Δt .

Besides stability and accuracy, computational efficiency of the solution scheme also has to be taken into account. For a fully implicit solver the computation time depends mostly on the amount of Newton iterations needed per time step and the time it takes to solve the linearized system each iteration. A nonlinear tolerance of $\epsilon = 10^{-6}$ was used, resulting in an average of one to six iterations per time step for all situations shown in Figure 2.7. Most iterations were used during the first time step. When keeping the ratio $\Delta t/\Delta x$ fixed at a Courant number of approximately one, the average amount of iterations comes down to two, which indicates a rapid convergence of the Newton method. The resulting CPU-times are shown in Table 2.1 and will be used as a base case throughout this research to compare efficiency of the developed transport solvers.

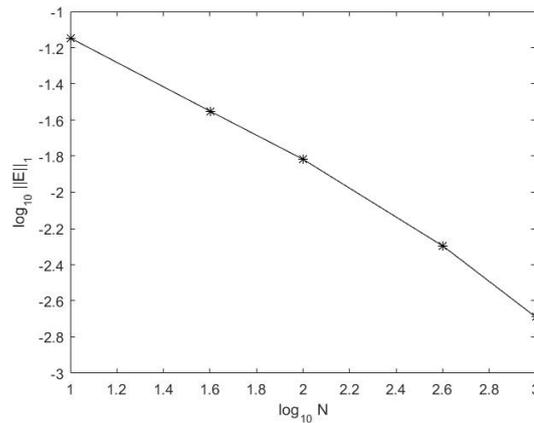


Figure 2.6: Log-log plot of the error versus the amount of grid cells N showing that the flow solver has a convergence rate smaller than one.

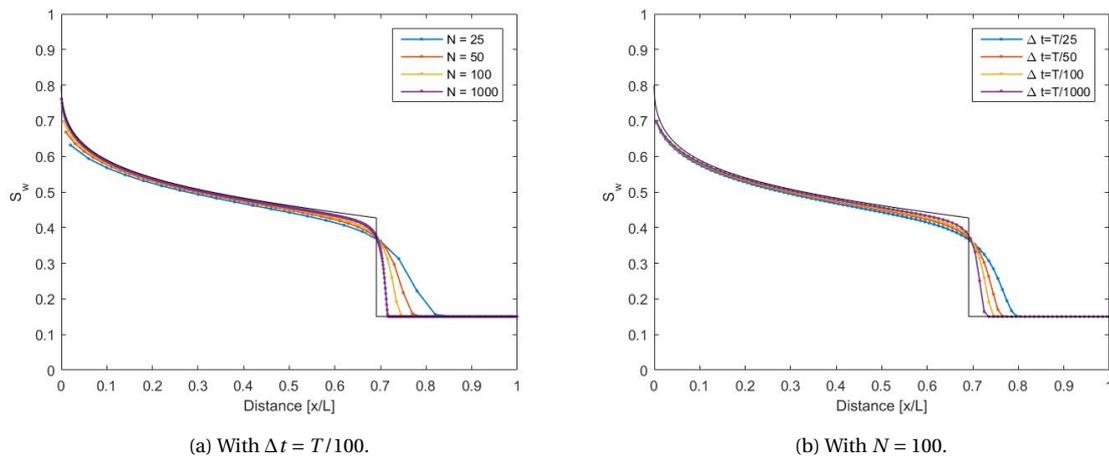


Figure 2.7: Solution of the fully implicit flow solver at a certain time t for different values of Δx and Δt .

N	CPU-time [s]
10	1.605
40	2.716
100	6.628
400	28.189
1000	79.970

Table 2.1: CPU-times of the flow solver for for increasing amount of grid cells with constant ratio $\Delta t/\Delta x$.

2.3.4. Extension to two dimensions

Extending the flow solver to two dimensions happens automatically in MRST if a 2D grid is supplied. However, for analysis purposes the discretization in two dimensions is explained shortly.

In two dimensions an identical process is applied as in the one dimensional case, only now a finite reservoir of $(x, y) \in [0, L] \times [0, L]$ is taken and divided it into $N \times N$ equally sized grid blocks c_{ij} given by

$$c_{ij} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}],$$

with cell centered grid points $\{(x_i, y_j)\}_{i,j=1}^{N,N}$. In two dimensions the flow equations are given by

$$\phi \frac{\partial}{\partial t} (S_\alpha) + \nabla \cdot (\mathbf{u}_\alpha) = 0 \quad \text{for } \alpha \in \{o, w\}, \quad (2.17)$$

with

$$\mathbf{u}_\alpha = -\frac{\mathbf{K}k_{r\alpha}}{\mu_\alpha} (\nabla p) = -\lambda_\alpha \nabla p.$$

In multiple dimensions the permeability \mathbf{K} is generally full tensor instead of a scalar, i.e.

$$\mathbf{K} = \begin{bmatrix} K_{xx} & K_{xy} \\ K_{yx} & K_{yy} \end{bmatrix}.$$

However, for simplicity we assume that it is a diagonal tensor $\mathbf{K} = \text{diag}(K_{xx}, K_{yy})$, which means that we assume the porous medium is *isotropic*.

Similar to in 1D a FV method is applied. Integrating equation (2.17) over the control volume V_c results in

$$\begin{aligned} \int_{V_c} \frac{\partial}{\partial t} (\phi S_\alpha) dV + \int_{V_c} \nabla \cdot (\mathbf{u}_\alpha) dV &= 0 \\ \Delta z \int_{c_{ij}} \frac{\partial}{\partial t} (\phi S_\alpha) + \Delta z \int_{c_{ij}} \nabla \cdot (\mathbf{u}_\alpha) dV &= 0. \end{aligned}$$

Applying Gauss' divergence theorem gives

$$\Delta z \int_{c_{ij}} \frac{\partial}{\partial t} (\phi S_\alpha) + \Delta z \int_{\delta c_{ij}} \mathbf{u}_\alpha \cdot \mathbf{n} dV = 0.$$

Using the fact that \mathbf{K} is diagonal, the boundary integral can be evaluated as

$$\int_{\delta c_{ij}} (\mathbf{u}_\alpha) \cdot \mathbf{n} dV = \Delta y u_{\alpha,x} \Big|_{(x_{i-1/2}, y_j)}^{(x_{i+1/2}, y_j)} + \Delta x u_{\alpha,y} \Big|_{(x_i, y_{j-1/2})}^{(x_i, y_{j+1/2})}.$$

where

$$u_{\alpha,x} \Big|_{(x_{i-1/2}, y_j)} = -K_{xx} \frac{(k_{r\alpha})_{i-1/2, j}}{\mu_\alpha} \frac{\partial p}{\partial x} \Big|_{(x_{i-1/2}, y_j)} = -K_{xx} \frac{k_{r\alpha}(S_{\alpha, i-1/2, j})}{\mu_\alpha} \frac{p_i - p_{i-1}}{\Delta x}.$$

As in 1D, a single-point upstream scheme is used to evaluate $k_{r\alpha}(S_{\alpha, i-1/2, j})$:

$$u_{\alpha,x} \Big|_{(x_{i-1/2}, y_j)} = -K_{xx} \frac{k_{r\alpha}(S_{\alpha, ij})}{\mu_\alpha} \frac{p_i - p_{i-1}}{\Delta x}.$$

Using the same time discretization as in the one dimensional case, we arrive at the fully implicit scheme for the two-dimensional case:

$$\phi V_c \frac{S_{\alpha, ij}^{n+1} - S_{\alpha, ij}^n}{\Delta t} = -\Delta y \Delta z u_{\alpha,x} \Big|_{(x_{i-1/2}, y_j)}^{(x_{i+1/2}, y_j)} - \Delta x \Delta z u_{\alpha,y} \Big|_{(x_i, y_{j-1/2})}^{(x_i, y_{j+1/2})}. \quad (2.18)$$

At the boundaries no-flow conditions are assumed and at the injector and producer a constant injection rate and constant pressure are imposed, similar to in 1D.

3

Transport of passive tracers

Tracers are chemical substances that can be added to fluids to track their movements. Various types of tracers have been used in hydrocarbon reservoirs over the years with the main purpose of acquiring information about fluid flow in the reservoir. This information is used to improve the description of the reservoir and thereby reduce uncertainties in the reservoir model. In all reservoir flooding cases, but especially in enhanced oil recovery applications, this information can be crucial in predicting and maximising the oil recovery [1].

Roughly speaking, tracers can be divided into two types, passive and active tracers. Active tracers influence the flow by changing the physical properties of the host fluid (such as density, viscosity, etc.), while passive tracers follow the fluids without affecting their physical properties. Tracers can remain in the fluid phase in which they are injected (non-partitioning tracer) or partition between the available phases (partitioning tracer). In this section the transport model for a non-partitioning passive water tracer is discussed.

A typical tracer transport equation includes convection, diffusion, dispersion, adsorption and source terms. Since tracer transport is mostly dominated by convection [22] rather than dispersion and diffusion, the latter terms are omitted from the transport equation. Adsorption only delays the tracer flow without drastically changing its solution so this term is omitted as well, see [17]. Furthermore, the fundamental assumption is made that the tracer has no mass and therefore also does not occupy any volume. The conservation equation for a passive water tracer is thus given by the simple advection equation

$$\frac{\partial}{\partial t}(\phi \rho_w S_w c) + \nabla \cdot (\rho_w c \mathbf{u}_w) = q,$$

where c is the tracer concentration (or mass-fraction) in water. Using the incompressibility of the water and rock this equation can be simplified to

$$\phi \frac{\partial}{\partial t}(S_w c) + \nabla \cdot (c \mathbf{u}_w) = q_w,$$

where $q_w = q/\rho_w$. This source term is only nonzero at the injection side of the reservoir and is therefore incorporated into the boundary condition. So the final conservative form of the tracer transport equation becomes:

$$\phi \frac{\partial}{\partial t}(S_w c) + \nabla \cdot (c \mathbf{u}_w) = 0. \quad (3.1)$$

Since S_w and u_w are known and independent of c this is simply a variable-coefficient linear advection equation.

3.1. Analytical solution

In 1D equation (3.1) turns into

$$\phi \frac{\partial}{\partial t}(S_w c) + \frac{\partial}{\partial x}(c u_w) = 0. \quad (3.2)$$

To solve this equation for the unknown concentration c on $x \in \mathbb{R}$, Riemann initial data are imposed:

$$c(x, 0) = \begin{cases} \bar{c} & x < 0 \\ 0 & x \geq 0. \end{cases} \quad (3.3)$$

As in Section 2.2, we take $u_T > 0$ and thus $u_w > 0$, such that the solution for $x > 0$ represents the solution in a reservoir on $x > 0$ where from $t = 0$ water with a constant tracer concentration \bar{c} is injected at $x = 0$. The water saturation and velocity are assumed to be known from the flow equation. Since the tracer does not influence the flow, the initial concentration profile (3.3) simply travels through the reservoir. The solution of the conserved quantity $S_w c$ thus contains a discontinuity at the point where there is a jump in c . This discontinuity marks the point at which the water with tracer ($S_w c$) is in contact with the water without tracer (S_w) and is therefore called a *contact discontinuity* [12]. The location of the discontinuity can be found by applying the Rankine–Hugoniot jump condition for systems, which in this situation states that at the discontinuity, the tracer velocity must equal the water velocity (see Appendix A).

The tracer velocity can be found by evaluating the derivatives in equation (3.1) and using the mass-conservation equation for the water phase

$$\phi S_w \frac{\partial c}{\partial t} + u_w \frac{\partial c}{\partial x} = 0,$$

which after substituting of $u_w = u_T f_w(S_w)$ can be written as

$$\frac{\partial c}{\partial t} + \frac{u_T f_w(S_w)}{\phi S_w} \frac{\partial c}{\partial x} = 0. \quad (3.4)$$

From the method of characteristics it directly follows that the characteristic velocity of the tracer is given by

$$\frac{dX}{dt} = \frac{u_T f_w(S_w)}{\phi S_w}. \quad (3.5)$$

In section 2.2 the characteristic water velocity was shown to be $\frac{dX}{dt} = \frac{u_T}{\phi} \frac{df_w}{dS_w}(S_w)$ which means that the contact discontinuity saturation S^* can be found by solving

$$\frac{f_w(S^*)}{S^*} = \frac{df_w}{dS_w}(S^*).$$

The solution of this equation is similar to that of equation (2.9), only in this case it is a straight line through $(0, 0)$ with slope $\frac{df_w}{dS_w}(S^*)$. This straight line represents the shock wave separating the water with tracer from the water without tracer. From S^* the tracer front location can be determined by

$$x(t) = \frac{u_T f_w(S^*)}{\phi S^*} t.$$

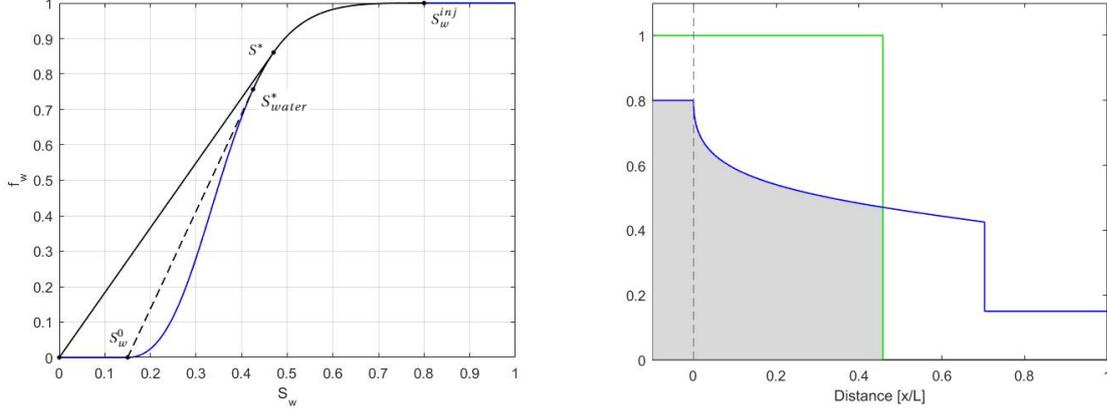
The analytical solution of the tracer concentration $c(x, t)$ with initial data (3.3) is thus given by

$$c(x, t) = \begin{cases} \bar{c} & \frac{x}{t} < \frac{u_T f_w(S^*)}{\phi S^*} \\ 0 & \frac{x}{t} \geq \frac{u_T f_w(S^*)}{\phi S^*}. \end{cases} \quad (3.6)$$

This solution together with the schematic illustration on how to find S^* are shown in Figure 3.1.

3.2. Numerical solution

For more complicated problems than the ones considered in this research, the tracer transport equation must be solved numerically. Since the passive tracer has no influence on the flow, the equations can be solved completely separately from the flow equation. After solving the flow, the obtained discrete values of S_w and u_w are used to solve the transport equation for the tracer concentration c . The grid spacing and time steps are adopted from the flow solver as well, meaning that Δt and Δx are fixed. Several numerical methods are available to solve equation (3.1). Three first order methods are discussed, followed by three high-resolution methods. This is all done for the one dimensional case first, whereafter an extension to two dimensions is given.



(a) Schematic illustration of the construction of the tracer solution. S^* denotes the tracer shock saturation while S_{water}^* denotes the water shock saturation.

(b) Solution of $S_w(x, t)$, $c(x, t)$ at a certain time $t > 0$. The water saturation $S_w(x, t)$ is shown in blue, the grey coloured area represents water with tracer ($S_w c$) and the normalized tracer concentration $c(x, t)/\bar{c}$ is shown in green.

Figure 3.1: Analytical solution of a passive tracer in two-phase porous media flow with Riemann initial data for certain time $t > 0$.

3.2.1. First order methods

Similar to the two-phase flow, the transport equation is discretized using a finite volume method combined with a first order temporal discretization:

$$\phi V_c \frac{(S_w c)_i^{n+1} - (S_w c)_i^n}{\Delta t} = -A_c (c_{i+1/2} u_{w,i+1/2} - c_{i-1/2} u_{w,i-1/2}).$$

For the advective fluxes an upwind approximation is used, which results in the following scheme:

$$\begin{aligned} (S_w c)_i^{n+1} &= (S_w c)_i^n - \frac{\Delta t}{\phi \Delta x} (c_i u_{w,i} - c_{i-1} u_{w,i-1}) \\ (S_w c)_i^{n+1} &= (S_w c)_i^n - \frac{\Delta t}{\phi \Delta x} (F_{i+1/2} - F_{i-1/2}). \end{aligned} \quad (3.7)$$

At the left boundary water with a constant tracer concentration \bar{c} is injected, i.e. $c_0^n = \bar{c}$. The water velocity at this boundary is equal to the injection velocity $u_{w,0}^n = u^{inj}$.

Dependent on how the fluxes $F_{i\pm 1/2}$ are evaluated, the following first order schemes are obtained

- Explicit:

$$F_{i+1/2} = F_{i+1/2}^n = c_i^n u_{w,i}^n \quad (3.8)$$

- Semi-implicit:

$$F_{i+1/2} = c_i^n u_{w,i}^{n+1} \quad (3.9)$$

- Implicit:

$$F_{i+1/2} = F_{i+1/2}^{n+1} = c_i^{n+1} u_{w,i}^{n+1} \quad (3.10)$$

Each of this schemes is used to solve equation (3.7) for the unknown value $(S_w c)_i^{n+1}$. The tracer concentration at time level t^{n+1} then follows from $c_i^{n+1} = \frac{(S_w c)_i^{n+1}}{S_{w,i}^{n+1}}$. The division by $S_{w,i}^{n+1}$ can lead to nonphysical values for c_i^{n+1} and numerical difficulties in case $S_{w,i}^{n+1}$ is close to zero. To prevent this from happening all saturation values smaller than some chosen tolerance level ϵ are replaced by ϵ . This tolerance level should be in line with the accuracy of the nonlinear solver used to compute S_w .

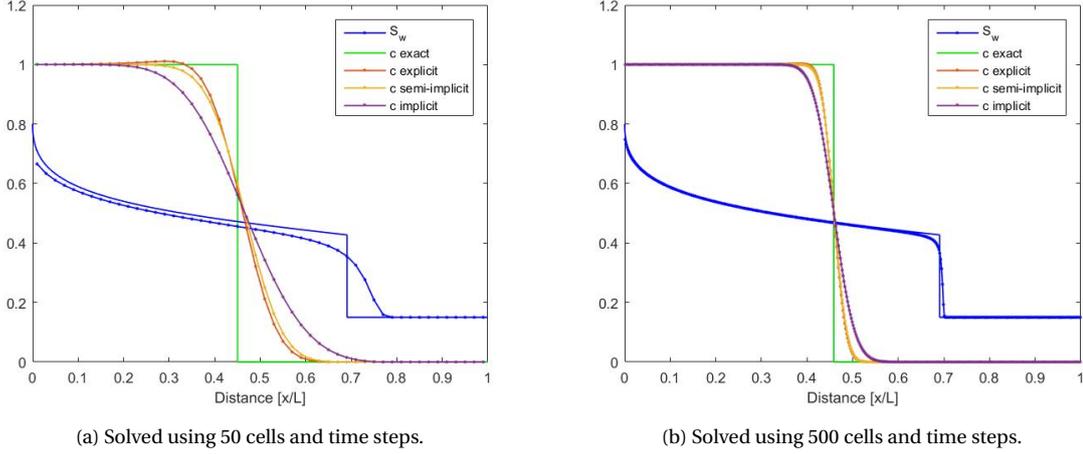


Figure 3.2: Numerical solution of $S_w(x, t)$ and $c(x, t)/\bar{c}$ for different first order methods at a certain time $t > 0$.

Accuracy

Numerical solutions for the situation of Figure 3.1b are shown in Figure 3.2a. All first order upwind methods introduce a great deal of numerical diffusion, yielding poor accuracy and smearing of the tracer front. From Figure 3.2a it can be seen that this numerical diffusion is most present in the implicit method. Refining the grid ten times (and scaling the time steps appropriately) severely reduces this numerical diffusion, but the discontinuity in c is still smeared, see Figure 3.2b.

Another observation from Figure 3.2a is that the explicit method shows concentration values above the injection concentration. This non-monotone behaviour of the solution is nonphysical and will be further addressed in Chapter 4.

Stability

Although the implicit method is the least accurate of all first order methods, it does have the advantage of producing a much more stable scheme. On the other hand, both the explicit and semi-implicit method are only conditionally stable. All methods are linear in c with variable coefficients $S_w(x, t)$ and $u_w(x, t)$. For linear methods, stability is obtained if

$$\|c^{n+1}\| \leq (1 + \alpha \Delta t) \|c^n\|$$

for some constant α [13]. This form of stability is generally referred to as *Lax-Richtmeyer stability*. From the *Lax equivalence theorem* it follows that if the method is stable, then it is also convergent. Since the numerical methods are all consistent and conservative, we can conclude from the *Lax-Wendroff theorem* that if the methods converge, then they converge to a weak solution $c(x, t)$ of the conservation law (3.1). This theorem was also proven to be true for linear equations with variable coefficients by Richtmeyer and Morton (1967).

However, proving stability for linear methods with variable coefficients is not trivial. If the coefficients are frozen at a certain point, then von Neumann analysis can be applied to find a local stability criterion. This condition leads to the necessary condition for stability in the form of the CFL condition:

$$C_T = \max_{S_w} \left| \frac{\Delta t}{\phi \Delta x} \frac{u_w(S_w)}{S_w} \right| = \max_{S_w} \left| \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} \frac{f_w(S_w)}{S_w} \right| \leq 1, \quad (3.11)$$

where C_T denotes the tracer Courant number. Since $\max_S (f'_w(S)) \geq \max_S \left(\frac{f_w(S)}{S} \right)$ for a general flux function like the one in Figure 2.1b, it holds that $C \geq C_T$, where C is the Courant number of the flow given by equation 2.14. This means that if the numerical scheme of the flow satisfies $C \leq 1$, then the necessary stability criterion for the transport equation is automatically satisfied.

The CFL condition is only a necessary condition for stability. In order to get a sufficient condition for stability of linear methods with variable coefficients, the von Neumann method has to be generalised. One way to do is to prove stability in the energy norm, i.e.

$$\|c^{n+1}\|_2^{\Delta x} = \sqrt{\Delta x \sum_i (c_i^{n+1})^2} \leq (1 + \Delta x) \|c^n\|_2^{\Delta x},$$

by using some properties of the numerical solution operator. An example of how this is done for the advection equation with variable coefficients is given in [16].

Computational cost

In matrix-vector notation, the explicit and semi-implicit schemes can be written as

$$(\mathbf{S}_w \mathbf{c})^{n+1} = (\mathbf{S}_w \mathbf{c})^n - A \mathbf{c}^n$$

with A a band matrix with nonzero entries on the diagonal and lower diagonal only. Per row this computation requires four operations (two multiplications and two deductions), so per time step the computational cost is $4N$ operations, where N is the amount of grid cells. Retrieving \mathbf{c}^{n+1} from $(\mathbf{S}_w \mathbf{c})^{n+1}$ costs another N operations, which brings the total to $5N$ operations per iteration.

For the implicit scheme the linear system

$$B \mathbf{c}^{n+1} = (\mathbf{S}_w \mathbf{c})^n$$

is solved by backward substitution, i.e.

$$\begin{aligned} c_1^{n+1} &= \frac{(S_w c)_1^n}{B_{1,1}} \\ c_i^{n+1} &= \frac{(S_w c)_i^n - B_{i,i-1} c_{i-1}^{n+1}}{B_{i,i}} \quad \text{for } i = 2 \dots N. \end{aligned}$$

This computation requires four operations per row as well. Combined with the operation to construct $B_{i,i-1}$ this results again a total of $5N$ operations per iteration.

So in terms of computational costs all three first order methods are equal.

3.2.2. High-resolution methods

In the previous section it was shown that first order methods severely smear the solution. To reduce the numerical diffusion introduced by the first order methods, high-resolution methods are implemented. These methods combine first order schemes with higher order schemes. When the underlying solution is smooth, the higher order (in our case second order) scheme is applied, while around discontinuities the method switches to the first order scheme. This approach is highly effective in advection problems, where the first order upwind scheme is very diffusive and the second order upwind scheme results in oscillations around the discontinuity [13]. Several classes of high-resolution methods exist, but the focus in this research is solely on total variation diminishing (TVD) flux-limiter methods, which guarantee that no nonphysical oscillations will arise in the solution. Multiple examples exist in literature where this class of methods has been successfully implemented in reservoir simulators to solve both flow and transport equations [1,5,6,14, 15,19, 22].

The flux limiter method combines a lower-order flux function with a higher-order flux function, i.e.

$$F_{i+1/2} = F_{L,i+1/2} + \Phi_{i+1/2} [F_{H,i+1/2} - F_{L,i+1/2}], \quad (3.12)$$

where F_L is the lower-order flux function, F_H the higher-order flux function and Φ the flux limiter function. The value of the flux limiter depends on θ , i.e. $\Phi_{i+1/2} = \Phi(\theta_{i+1/2})$, with

$$\theta_{i+1/2} = \frac{\Delta c_{i-1/2}}{\Delta c_{i+1/2}} = \frac{c_i - c_{i-1}}{c_{i+1} - c_i}.$$

When the data is smooth, $\theta_{i+1/2} \approx 1$, the higher-order flux is used, $\Phi(1) = 1$, while around discontinuities where $\theta_{i+1/2} \approx 0$ the method switches to the first order flux, $\Phi(0) = 0$. If a so-called TVD limiter is used, then it is possible to prove that the resulting flux-limiting scheme will be TVD [13]. This means that the total variation of the solution defined by

$$TV(c) = \sum_{i=1}^N |c_i - c_{i-1}|,$$

does not grow over time, i.e. $TV(c^{n+1}) \leq TV(c^n)$. Several TVD limiters exist, of which the *superbee* and the *van Leer* limiter, shown in Figure 3.3, are the most widely used.

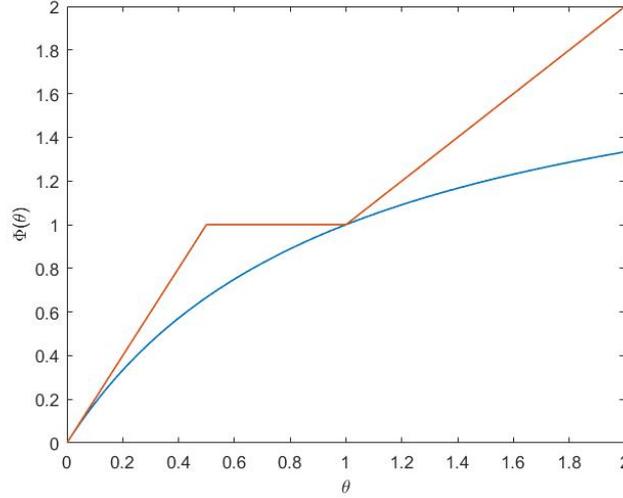


Figure 3.3: Most widely used TVD limiters. The superbee limiter, $\Phi(\theta) = \max(0, \min(1, 2\theta), \min(2, \theta))$, is shown in red and the van Leer limiter, $\Phi(\theta) = \frac{\theta + |\theta|}{1 + |\theta|}$, is shown in blue.

For the tracer transport problem (3.1), F_L is simply the first order upwind flux from section 3.2.1 and F_H is the corresponding second order upwind flux:

$$F_{L,i+1/2} = c_i u_{w,i}$$

$$F_{H,i+1/2} = \frac{1}{2} u_{w,i} (c_i + c_{i+1}) - \frac{1}{2} (u_{w,i})^2 \frac{\Delta t}{\Delta x} (c_{i+1} - c_i).$$

Substituting this into equation (3.12) results in the flux function

$$F_{i+1/2} = c_i u_{w,i} + \Phi(\theta_{i+1/2}) \frac{1}{2} u_{w,i} \left(1 - u_{w,i} \frac{\Delta t}{\Delta x} \right) [c_{i+1} - c_i]. \quad (3.13)$$

Combined with (3.7) this gives the high-resolution flux limiter method. Similar to the different fluxes used for the first order method, explicit, semi-implicit and fully implicit high-resolution fluxes are obtained by changing the time levels at which c and u_w are evaluated:

- Explicit High-Resolution:

$$F_{i+1/2} = F_{i+1/2}^n = c_i^n u_{w,i}^n + \Phi(\theta_{i+1/2}^n) \frac{1}{2} u_{w,i}^n \left(1 - u_{w,i}^n \frac{\Delta t}{\Delta x} \right) [c_{i+1}^n - c_i^n] \quad (3.14)$$

- Semi-implicit High-Resolution:

$$F_{i+1/2} = c_i^n u_{w,i}^{n+1} + \Phi(\theta_{i+1/2}^n) \frac{1}{2} u_{w,i}^{n+1} \left(1 - u_{w,i}^{n+1} \frac{\Delta t}{\Delta x} \right) [c_{i+1}^n - c_i^n] \quad (3.15)$$

- Implicit High-Resolution:

$$F_{i+1/2} = F_{i+1/2}^{n+1} = c_i^{n+1} u_{w,i}^{n+1} + \Phi(\theta_{i+1/2}^{n+1}) \frac{1}{2} u_{w,i}^{n+1} \left(1 - u_{w,i}^{n+1} \frac{\Delta t}{\Delta x} \right) [c_{i+1}^{n+1} - c_i^{n+1}] \quad (3.16)$$

Both the explicit and semi-implicit fluxes result in a linear expression for c_i^{n+1} . On the other hand, since $\Phi(\theta_{i+1/2}^{n+1})$ is a nonlinear function of c_i^{n+1} , the implicit high-resolution flux results in a nonlinear expression for c_i^{n+1} which must be solved using Newton's method.

Newton method for implicit high-resolution scheme

In residual form the implicit high-resolution scheme can be written as

$$R(c_i^{n+1}) = (S_w c)_i^{n+1} - (S_w c)_i^n + \frac{\Delta t}{\phi \Delta x} (F_{i+1/2}^{n+1} - F_{i-1/2}^{n+1}) = 0$$

As for the flow, the solution c_i^{n+1} is obtained at every time step using the iterative process

$$\left(\frac{\partial R_i}{\partial c_j} \right)^k \delta c_j^{k+1} = -R_i^k, \quad k = 0, 1, \dots; \quad j = 1, \dots, N$$

where $c_j^{k+1} = c_j^k + \delta c_j^{k+1}$ and $c_i^{k=0} = c_i^n$. For a first order scheme, R_i is a function of concentrations in cells i and $i-1$ only and the Jacobian $\frac{\partial R_i}{\partial c_j}$ has only two nonzero entries per row. However, for the more sophisticated high-resolution scheme considered here the Jacobian contains more non-zero elements, which makes solving the resulting linear equation more difficult and time-consuming. For this reason only first order terms are considered in calculation of the Jacobian:

$$\left(\frac{\partial R_i}{\partial c_i} \right)^k = S_{w,i}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{w,i}^{n+1} \quad \left(\frac{\partial R_i}{\partial c_{i-1}} \right)^k = -\frac{\Delta t}{\phi \Delta x} u_{w,i-1}^{n+1}.$$

To ensure convergence to the correct high-resolution solution, the right hand side $-R_i^k$ is still calculated to second order. Although the derivatives of the limiter are disregarded in calculating the Jacobian, the convergence of the Newton method is still sensitive to discontinuities in Φ . Therefore the continuous *van Leer* limiter is selected for all high-resolution methods.

Accuracy and computational costs

Numerical solutions using the high-resolution schemes with the van Leer limiter are shown in Figure 3.4. Based on these results the following conclusions can be drawn.

The semi-implicit high-resolution method performs best and severely reduces the numerical diffusion compared to the first order solutions shown in Figure 3.2.

The implicit high-resolution method shows improvement with respect to the implicit first order method, but does not outperform the first order explicit or first order semi-implicit method. Due to the nonlinearity of the scheme this method is computationally the most expensive and should therefore only be considered in case the time step restriction is such that all explicit and semi-implicit methods become unstable.

The explicit high-resolution method shows oscillations at the tracer front. Just like the behaviour shown by the explicit first order method, this behaviour is nonphysical and will be further addressed in Chapter 4.

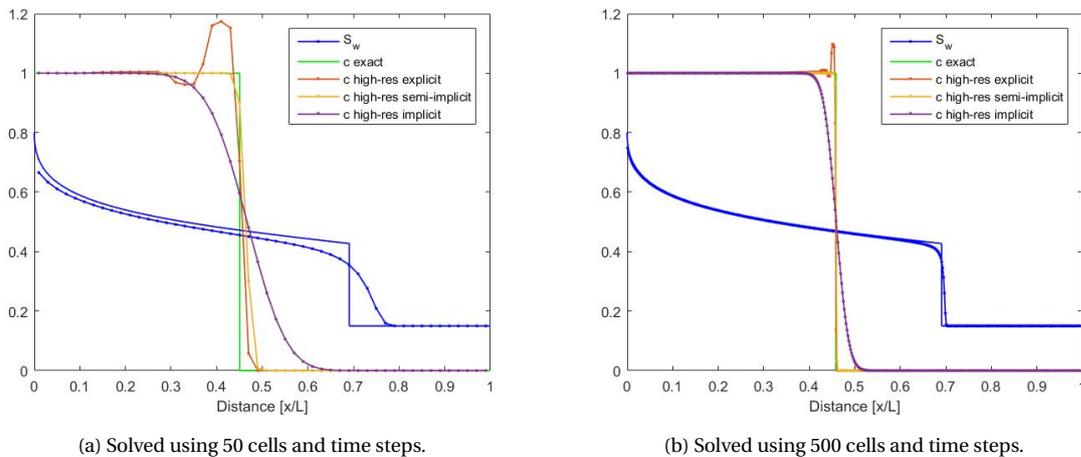


Figure 3.4: Numerical solution of $S_w(x, t)$ and $c(x, t)/\bar{c}$ for three different high-resolution methods at a certain time $t > 0$.

The increase in accuracy obtained when switching from a semi-implicit first order method to a semi-implicit high-resolution method can be quantified by comparing the 1-norm errors for both methods:

$$\|E\|_1 = \Delta x \sum_{i=1}^N |E_i| = \Delta x \sum_{i=1}^N |c_i^n - c(x_i, t_n)|.$$

A log-log plot of the error versus the amount of grid cells N is shown in Figure 3.5. As the grid is refined, the ratio $\Delta t/\Delta x$ is kept constant. The results not only show an increase in accuracy when using the high-resolution method, but also a faster convergence rate up to $N = 400$. Furthermore it can be concluded that the order of accuracy of the first order method is approximately $\mathcal{O}(\sqrt{\Delta x})$, which is lower than the order of accuracy of the flow solver.

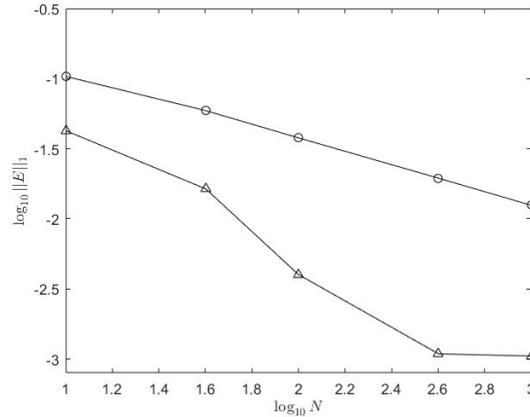


Figure 3.5: Log-log plot of the 1-norm error versus the amount of grid cells N for the first order semi-implicit method (circles) and the high-resolution semi-implicit method (triangles).

An increase in accuracy is at the expense of an increase in computational costs as can be seen from the CPU-times in Table 3.1. However, compared to the CPU-times of the flow solver (Table 2.1) the computational costs of the high-resolution method are still very low. Table 2.1 also confirms that the implicit high-resolution method is significantly more expensive than the other methods.

N	CPU-time [s]		
	First order methods	Semi-implicit high-res method	Implicit high-res method
10	1.3404e-04	9.5823e-04	4.0240e-02
40	4.4782e-04	3.7404e-03	2.0566e-01
100	1.0373e-03	1.1345e-02	5.1546e-01
400	8.5824e-03	9.0978e-02	1.7706e-00
1000	4.9727e-02	4.6072e-01	5.8027e-00

Table 3.1: CPU-times of the semi-implicit first order, semi-implicit high-resolution and fully implicit high-resolution method for increasing amount of grid cells.

Stability

Since the limiter is a nonlinear function of the concentration c , the resulting high-resolution methods are all nonlinear. As mentioned in section 2.3.3, nonlinear methods must be total variation bounded (TVB) in order to be stable. All our methods are TVD, i.e. $TV(c^{n+1}) \leq TV(c^n)$, and therefore they are certainly TVB, with $r = TV(c^0)$. Thus all high-resolution methods are stable provided that the CFL condition is satisfied, which is a necessary condition for a method to be TVD [13].

3.2.3. Conclusions

For the transport of a passive water tracer described by the simple advection equation (3.1), the semi-implicit high-resolution method given by equation (3.15) performs best in terms of accuracy and efficiency. If the

time step is too large and this method becomes unstable, implicit methods have to be applied. The implicit high-resolution method is more accurate than the implicit first order method, but it is computationally much more expensive so the accuracy has to be weighed against the efficiency. Furthermore, the explicit first order and high-resolution method show nonphysical solutions and should therefore not be used to solve the tracer transport equation. The origin of this nonphysical behaviour is further addressed in Chapter 4.

3.3. Extension to two dimensions

As for the flow solver, the methods for the passive tracer transport are extended to two dimensions. In two dimensions the tracer transport equation is given by

$$\phi \frac{\partial}{\partial t} (S_w c) + \nabla \cdot (c \mathbf{u}_w) = 0. \quad (3.17)$$

For this equation, a simple first-order finite volume method in two-dimensions is derived in a similar way as in section 2.3.4:

$$\phi V_c \frac{S_{w,ij}^{n+1} c_{ij}^{n+1} - S_{w,ij}^n c_{ij}^n}{\Delta t} = -\Delta y \Delta z (c u_{w,x}) \Big|_{(x_{i-1/2}, y_j)}^{(x_{i+1/2}, y_j)} - \Delta x \Delta z (c u_{w,y}) \Big|_{(x_i, y_{j-1/2})}^{(x_i, y_{j+1/2})}.$$

By omitting the subscripts w and using upwind approximations for the advective fluxes we obtain

$$\phi V_c \frac{S_{ij}^{n+1} c_{ij}^{n+1} - S_{ij}^n c_{ij}^n}{\Delta t} = -\Delta y \Delta z [c_{ij} u_{x,i} - c_{i-1,j} u_{x,i-1}] - \Delta x \Delta z [c_{ij} u_{y,j} - c_{i,j-1} u_{y,j-1}].$$

This can be rewritten as

$$S_{ij}^{n+1} c_{ij}^{n+1} = S_{ij}^n c_{ij}^n - \frac{\Delta t}{\phi \Delta x} [F_{i+1/2,j} - F_{i-1/2,j}] - \frac{\Delta t}{\phi \Delta y} [G_{i,j+1/2} - G_{i,j-1/2}] \quad (3.18)$$

with the first order flux functions

$$F_{i+1/2,j} = c_{ij} u_{x,i} \quad G_{i,j+1/2} = c_{ij} u_{y,j}.$$

This first order method for the advection equation in two dimensions is often called the *donor-cell upwind (DCU) method* [13]. High-resolution methods are obtained by changing the flux functions to

$$F_{i+1/2} = c_{ij} u_{x,i} + \Phi(\theta_{i+1/2,j}) \frac{1}{2} u_{x,i} \left(1 - u_{x,i} \frac{\Delta t}{\Delta x}\right) [c_{i+1,j} - c_{ij}]$$

$$G_{i+1/2} = c_{ij} u_{y,j} + \Phi(\theta_{i,j+1/2}) \frac{1}{2} u_{y,j} \left(1 - u_{y,j} \frac{\Delta t}{\Delta y}\right) [c_{i,j+1} - c_{ij}].$$

As in 1D explicit, semi-implicit and implicit first order and high resolution methods are obtained by changing the time levels at which c and \mathbf{u} are evaluated.

The fully multidimensional scheme (3.18) becomes very expensive if we're dealing with large amount of grid cells. For the explicit and semi-implicit fluxes, a more simple and relatively inexpensive approach to extend the one-dimensional method to more dimensions is to use dimensional splitting. In this approach the multidimensional problem is split into two one-dimensional problems

$$\phi \frac{\partial}{\partial t} (S_w c) + \frac{\partial}{\partial x} (c u_x) = 0$$

$$\phi \frac{\partial}{\partial t} (S_w c) + \frac{\partial}{\partial y} (c u_y) = 0$$

which are solved sequentially in the x and y -direction by

$$(S_{ij} c_{ij})^* = (S_{ij} c_{ij})^n - \frac{\Delta t}{\phi \Delta x} [F_{i+1/2,j} - F_{i-1/2,j}]$$

$$(S_{ij} c_{ij})^{n+1} = (S_{ij} c_{ij})^* - \frac{\Delta t}{\phi \Delta y} [G_{i,j+1/2}^* - G_{i,j-1/2}^*].$$

A downside of this approach is that it introduces a splitting error. However, since this error is often no worse than error introduced by the numerical methods in each direction this method is often very effective [13].

For the implicit fluxes a similar approach under the name *alternating direction implicit* (ADI) method exists. However, since most commercial simulators use the fully multidimensional approach, this approach is implemented in the tracer module.

For the DCU method (3.18) the necessary condition for stability is given by [13]:

$$\max_{S_w} \left(\left| \frac{\Delta t}{\phi \Delta x} \frac{u_{w,x}(S_w)}{S_w} \right| + \left| \frac{\Delta t}{\phi \Delta y} \frac{u_{w,y}(S_w)}{S_w} \right| \right) \leq 1,$$

This stability bound is quite strict and can be improved by using a numerical method that takes more fully account of the flow direction by incorporating the term $c_{i-1,j-1}$. However, since the focus in this research is on the one-dimensional case we stick to the simple DCU method.

In one dimension we encountered nonphysical solutions for the explicit first order and explicit high-resolution method. This same behaviour is encountered in two dimensions, but here the oscillations quickly die out since there is more numerical diffusion than in one dimension. The monotonicity of the two-dimensional scheme will be investigated in Chapter 4 as well.

3.4. Nonconservative methods

So far all proposed schemes solve the conservation law (3.2) for the unknown conserved quantity $S_w c$ and retrieve the tracer concentration c by taking the ratio $S_w c / S_w$. As mentioned in section 3.2.1, this can lead to some numerical difficulties. To avoid these difficulties LeVeque (2002) proposes to use the nonconservative transport equation, given by

$$\frac{\partial c}{\partial t} + \frac{u_w}{\phi S_w} \frac{\partial c}{\partial x} = 0$$

where the velocity $\frac{u_w}{\phi S_w}$ is determined by the nonlinear conservation laws for the flow. A simple first-order semi-implicit advection scheme for this equation is given by

$$c_i^{n+1} = c_i^n - \frac{\Delta t}{\Delta x} \frac{u_{w,i}^{n+1}}{\phi S_{w,i}^{n+1}} (c_i^n - c_{i-1}^n).$$

This can easily be extended to a high-resolution scheme

$$c_i^{n+1} = c_i^n - \frac{\Delta t}{\Delta x} \frac{u_{w,i}^{n+1}}{\phi S_{w,i}^{n+1}} (c_i^n - c_{i-1}^n) - \frac{\Delta t}{\Delta x} (\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2})$$

where

$$\tilde{F}_{i-1/2} = \frac{1}{2} \frac{u_{w,i}^{n+1}}{\phi S_{w,i}^{n+1}} \left(1 - \frac{\Delta t}{\Delta x} \frac{u_{w,i}^{n+1}}{\phi S_{w,i}^{n+1}} \right) \Phi_{i-1/2} (c_i^n - c_{i-1}^n).$$

There are two main concerns about nonconservative methods. First of all the fact that they are not conservative results in numerical mass loss, which means that part of the conserved quantity disappears during the simulation. Since the conserved quantity considered here is $S_w c$, the fraction of water that contains tracer, the numerical approximated fraction will be less than the actual fraction of water that should contain tracer. Note that since the flow solver is conservative, the total amount of water is not subject to numerical mass loss. The loss of $S_w c$ results in an underestimation of the tracer front. For the passive tracer this underestimation still gives quite satisfactory results and in some cases even leads to smaller numerical errors than the errors introduced by the conservative methods.

The second concern is that there is no guarantee that nonconservative methods converge to the correct weak solution, especially if this solution contains shock waves [12]. For conservative methods we do have this guarantee in the form of the Lax-Wendroff theorem. In case of a passive tracer the method seems to converge to the correct weak solution and no problems arise. However, when treating active tracers we do see that the nonconservative method converges to a different and incorrect weak solution. Since the goal is to develop a numerical method that is able to accurately solve both active and passive tracers, the nonconservative numerical method is omitted hereafter.

4

Monotonicity

One of the advantages of upwind methods is that they keep the solution monotonically varying in regions where the solution should be monotone [13]. This feature is also the reason for high-resolution methods to switch to first order upwind methods around discontinuities.

However, as shown in Figure 3.2a and 3.5 in the previous section, the explicit first order upwind method fails to keep the tracer solution monotone, which leads to oscillations in the solution obtained by the explicit high-resolution method. This is precisely the behaviour that high-resolution methods were meant to prevent. Further investigation showed that for initial water saturations closer to zero, this behaviour is even more present, see Figure 4.1a, and that similar behaviour arises in the semi-implicit high-resolution method, see Figure 4.1b. The implicit high-resolution method does seem to preserve the monotonic behaviour of the solution (see Figure 4.1c), but this method is the least accurate and computationally most costly of the three high-resolution schemes and should be avoided if possible.

In order to develop a stable, accurate and efficient method that is physically correct as well, more investigation into all three methods and their monotonicity is necessary. The fact that the oscillations in the high-resolution methods are amplified or only arise for initial water saturations close to zero, indicates a strong dependency of the numerical methods on the underlying discrete flow solution. This dependency and its effect on the monotonicity of the tracer solution is further investigated in this chapter.

4.1. Monotonicity analysis

To analyse the monotonicity of the schemes, the same one-dimensional problem as in Section 3.1 is considered, where $u_w > 0$ is given and water with a constant tracer concentration \bar{c} is injected at $x = 0$. A discrete version of the analytical solution of the tracer concentration (3.6) at time t_n is given by the piecewise constant data:

$$c(x_i, t_n) = c_i^n = \begin{cases} \bar{c} & \text{if } i \leq I \\ 0 & \text{if } i > I, \end{cases} \quad (4.1)$$

which is monotone, i.e. $c_i^n \geq c_{i+1}^n$ for all i . Furthermore the time step is assumed such that the CFL condition (3.11) is satisfied, which means that the tracer solution travels less than one grid cell per time step. All three first order schemes are then used to construct the solution at the next time step c^{n+1} . In order for the methods to be monotonicity-preserving this solution must satisfy $c_i^{n+1} \geq c_{i+1}^{n+1}$ for all i .

To prove or contradict this property, concentrations up to cell $I + 1$ are evaluated for each method and analysed using the underlying flow field which followed from the fully implicit scheme (2.11). Rewriting this scheme in terms of the water saturation S_w and velocity u_w , and omitting the subscript w in the notation, results in the simplified expression

$$S_i^{n+1} = S_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^{n+1} - u_{i-1}^{n+1}). \quad (4.2)$$

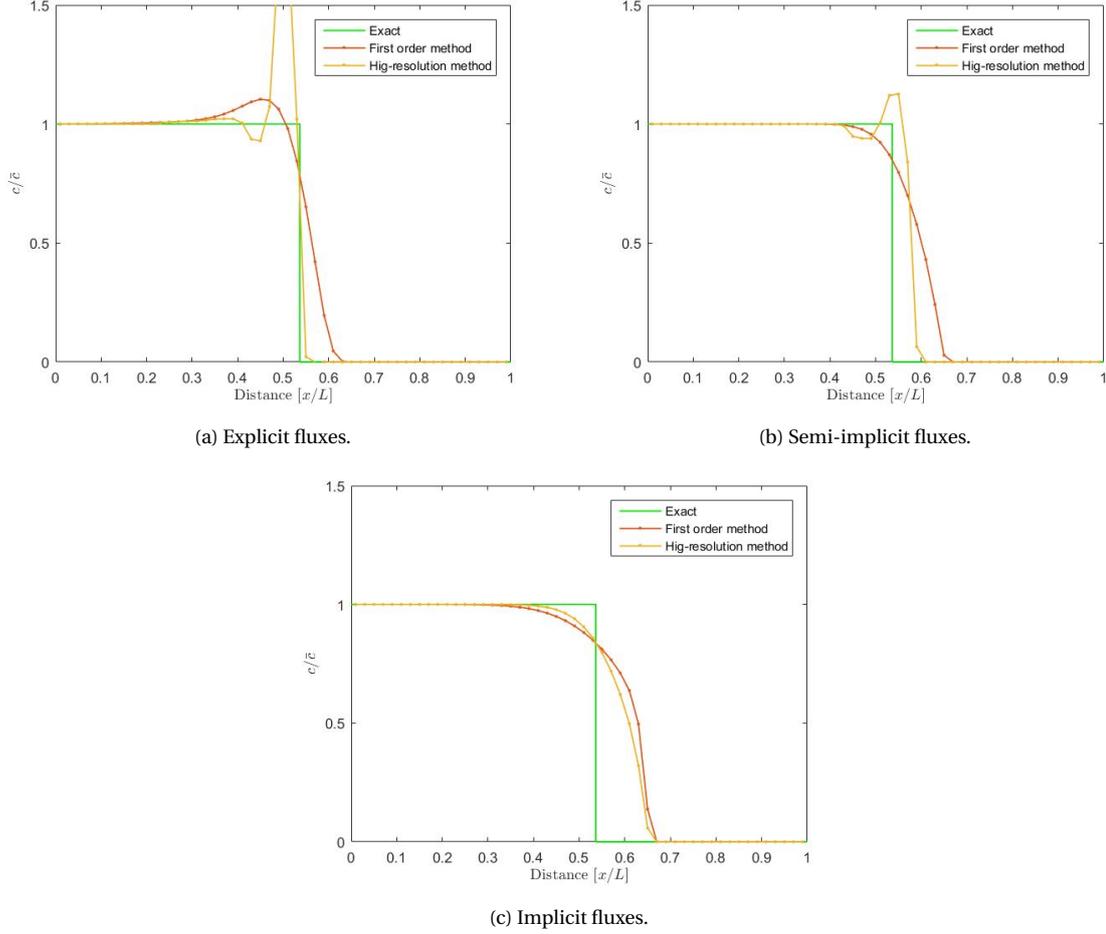


Figure 4.1: Numerical solution of $c(x, t)/\bar{c}$ for three different high-resolution methods at a certain time $t > 0$ for a low initial water saturation, $S_w^0 = S_{wc} = 0.015$.

4.1.1. Explicit scheme

The explicit scheme is applied to (4.1). At time $n + 1$ the solution in cells $i = 1 \dots I$ is then given by

$$\begin{aligned} S_i^{n+1} c_i^{n+1} &= S_i^n c_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^n c_i^n - u_{i-1}^n c_{i-1}^n) \\ &= \left[S_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^n - u_{i-1}^n) \right] \bar{c}. \end{aligned}$$

The expression $S_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^n - u_{i-1}^n)$ on the right hand side can be seen as the explicit approximation of S_i^{n+1} , which will be denoted by \tilde{S}_i^{n+1} . The solution c_i^{n+1} thus depends on the ratio between the explicit and implicit approximation of $S(x_i, t_{n+1})$:

$$c_i^{n+1} = \frac{\tilde{S}_i^{n+1}}{S_i^{n+1}} \bar{c} \quad \text{for } i = 1 \dots I. \quad (4.3)$$

Since the implicit method is more diffusive than the explicit method, the explicit approximation \tilde{S}_i^{n+1} will be closer to the analytical solution $S(x_i, t_{n+1})$ than the implicit approximation S_i^{n+1} . This means that up to the water front, the explicit approximation lies above the implicit approximation, while after the front the implicit approximation takes over, i.e.

$$\begin{aligned} \tilde{S}_i^{n+1} &> S_i^{n+1} & \text{for } i = 1 \dots J \\ \tilde{S}_i^{n+1} &\leq S_i^{n+1} & \text{for } i = J + 1 \dots N, \end{aligned}$$

for some grid cell J close to the shock front, see Figure 4.2. The ratio between the two approximations is thus given by

$$\frac{\tilde{S}_i^{n+1}}{S_i^{n+1}} > 1 \quad \text{for } i = 1 \dots J$$

$$\frac{\tilde{S}_i^{n+1}}{S_i^{n+1}} \leq 1 \quad \text{for } i = J + 1 \dots N.$$

From Figure 4.3 it can be seen that the ratio is not only bigger than one up to $i = J$, but also increasing up to some cell $K < J$ before it drops below one. This is used in equation (4.3) to obtain the solution

$$\bar{c} < c_1^{n+1} < \dots < c_{i-1}^{n+1} < c_i^{n+1} \quad \text{for } i = 1 \dots K.$$

This solution is clearly not monotone and it can therefore be concluded that the explicit scheme is not monotonicity-preserving.

For lower initial water saturations, the difference between the implicit and explicit approximations becomes larger, leading to larger variations in the tracer solution c^{n+1} . This explains the difference between the solution shown in Figure 3.2a and the one shown in Figure 4.1a.

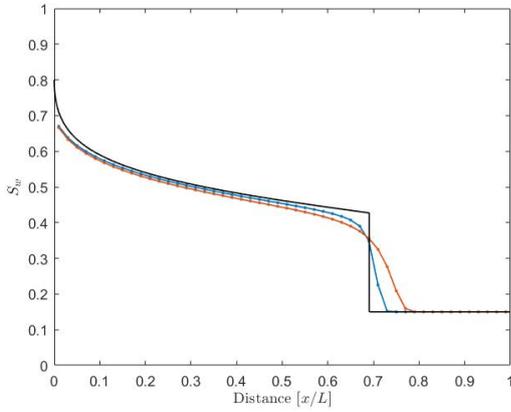


Figure 4.2: Numerical solution of the water saturation $S_w(x, t)$ using an explicit method (blue) and an implicit method (red) together with the analytical solution (black).

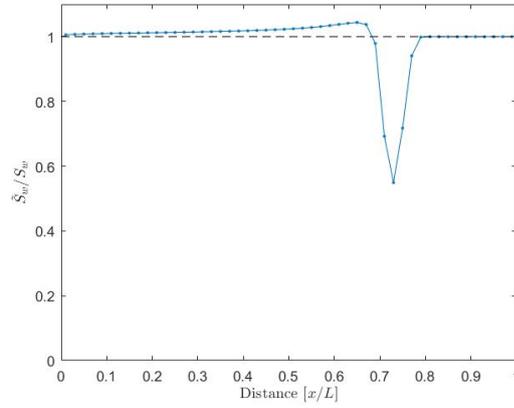


Figure 4.3: Ratio between the explicit approximation $\tilde{S}_{w,i}^n$ and the implicit approximation $S_{w,i}^n$.

Note that if the flow would be solved explicitly then equation (4.3) would result in $c_i^{n+1} = \bar{c} = c_i^n$ for $i = 1 \dots I$. In cell $I + 1$ the solution would then be given by

$$\tilde{S}_{I+1}^{n+1} c_{I+1}^{n+1} = S_{I+1}^n c_{I+1}^n - \frac{\Delta t}{\phi \Delta x} (u_{I+1}^n c_{I+1}^n - u_I^n c_I^n)$$

$$= \frac{\Delta t}{\phi \Delta x} u_I^n \bar{c}.$$

Rewriting the expression for \tilde{S}_{I+1}^n into $\frac{\Delta t}{\phi \Delta x} u_I^n = \tilde{S}_{I+1}^{n+1} - S_{I+1}^n + \frac{\Delta t}{\phi \Delta x} u_{I+1}^n$ and substituting this gives

$$\tilde{S}_{I+1}^{n+1} c_{I+1}^{n+1} = \left[\tilde{S}_{I+1}^{n+1} - S_{I+1}^n + \frac{\Delta t}{\phi \Delta x} u_{I+1}^n \right] \bar{c}$$

so the tracer concentration in cell $I + 1$ is given by

$$c_{I+1}^{n+1} = \frac{\tilde{S}_{I+1}^{n+1} - \left(S_{I+1}^n - \frac{\Delta t}{\phi \Delta x} u_{I+1}^n \right)}{\tilde{S}_{I+1}^{n+1}} \bar{c}$$

$$= \left[1 - \frac{S_{I+1}^n - \frac{\Delta t}{\phi \Delta x} u_{I+1}^n}{\tilde{S}_{I+1}^{n+1}} \right] \bar{c}.$$

From the discrete form of the CFL condition (3.11)

$$\frac{\Delta t}{\phi \Delta x} u_i^n \leq S_i^n \quad \forall i, n, \quad (4.4)$$

it follows that $\left(S_{I+1}^n - \frac{\Delta t}{\phi \Delta x} u_{I+1}^n\right) \geq 0$. Furthermore it holds that $\tilde{S}_{I+1}^{n+1} \geq S_{I+1}^n$ as well, which results in

$$c_{I+1}^{n+1} \leq \bar{c} = c_I^n.$$

It can thus be concluded that $c_{i+1}^{n+1} \leq c_i^{n+1}$ for all i , meaning that the solution at time t_{n+1} is monotone.

This shows that the explicit scheme for the transport equation is only monotonicity-preserving if the underlying flow is also solved using explicit fluxes, assuming that the CFL condition is met. For a fully implicit discretization of the flow, the explicit transport scheme always results in a non-monotonous solution, regardless of whether the CFL condition is met or not.

4.1.2. Semi-implicit scheme

If the semi-implicit scheme is applied to (4.1), then the solution in cells $i = 1 \dots I$ at time $n + 1$ is given by

$$\begin{aligned} S_i^{n+1} c_i^{n+1} &= S_i^n c_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^{n+1} c_i^n - u_{i-1}^{n+1} c_{i-1}^n) \\ &= \left[S_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^{n+1} - u_{i-1}^{n+1}) \right] \bar{c} \\ &= S_i^{n+1} \bar{c}, \end{aligned}$$

where the last steps follow from equation (4.2). The solution at time $n + 1$ thus equals the solution at time n for $i = 1 \dots I$:

$$c_i^{n+1} = \bar{c} = c_i^n.$$

In cell $I + 1$ the solution is given by

$$\begin{aligned} S_{I+1}^{n+1} c_{I+1}^{n+1} &= S_{I+1}^n c_{I+1}^n - \frac{\Delta t}{\phi \Delta x} (u_{I+1}^{n+1} c_{I+1}^n - u_I^{n+1} c_I^n) \\ &= \frac{\Delta t}{\phi \Delta x} u_I^{n+1} \bar{c}. \end{aligned} \quad (4.5)$$

By rewriting equation (4.2), u_I^{n+1} can be expressed in terms of values in the cell $I + 1$ only:

$$\frac{\Delta t}{\phi \Delta x} u_I^{n+1} = S_{I+1}^{n+1} - S_{I+1}^n + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1}.$$

Substituting this into equation (4.5) results in

$$\begin{aligned} S_{I+1}^{n+1} c_{I+1}^{n+1} &= \left[S_{I+1}^{n+1} - S_{I+1}^n + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1} \right] \bar{c} \\ c_{I+1}^{n+1} &= \left[1 - \left(\frac{S_{I+1}^n}{S_{I+1}^{n+1}} - \frac{\Delta t}{\phi \Delta x} \frac{u_{I+1}^{n+1}}{S_{I+1}^{n+1}} \right) \right] \bar{c}. \end{aligned}$$

For the scheme to be monotonicity-preserving it must satisfy $c_{I+1}^{n+1} \leq c_I^{n+1} = \bar{c}$, i.e.

$$\frac{S_{I+1}^n}{S_{I+1}^{n+1}} - \frac{\Delta t}{\phi \Delta x} \frac{u_{I+1}^{n+1}}{S_{I+1}^{n+1}} \geq 0.$$

Assuming that $S_{I+1}^{n+1} > 0$, the condition simplifies to

$$\frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1} \leq S_{I+1}^n. \quad (4.6)$$

If $S_{I+1}^n > 0$, the time step can always be chosen such that this relation holds. Consequence is that for small saturation values, a very small time step might be needed to ensure monotonicity of the scheme. This time step constraint is much stricter than the CFL condition (4.4), meaning that the stability of the scheme offers no guarantee on the monotonicity of the scheme.

If $S_{I+1}^n = 0$, which is only possible if the connate water saturation is equal to zero and the tracer and water discontinuity align, then the relation can never be met since we assumed that $S_{I+1}^{n+1} > 0$ and $u_w(S_w) = u_T f(S_w) > 0$ for $S_w > S_{wc} = 0$. The semi-implicit scheme is therefore never monotonous in case the tracer is injected simultaneously with water into a reservoir with an initial water saturation equal to zero. For situations approaching this limit case, the monotonicity of the scheme is dependent on the time step. If enough water is present in the reservoir, then the condition (4.6) is easily satisfied.

The non-monotonous behaviour of the first order solution is so limited that it is not visible in Figure 4.1b. However, the high-resolution method is very sensitive to any non-monotonous behaviour in the first order solution, as can be seen from the oscillation in Figure 4.1b. This will be further discussed at the end of this section.

4.1.3. Implicit scheme

If the implicit scheme is applied to (4.1), then the solution at time $n + 1$ in cells $i = 1 \dots I$ is given by

$$\begin{aligned} S_i^{n+1} c_i^{n+1} &= S_i^n c_i^n - \frac{\Delta t}{\phi \Delta x} (u_i^{n+1} c_i^{n+1} - u_{i-1}^{n+1} c_{i-1}^{n+1}) \\ \left[S_i^{n+1} + \frac{\Delta t}{\phi \Delta x} u_i^{n+1} \right] c_i^{n+1} &= S_i^n \bar{c} + \frac{\Delta t}{\phi \Delta x} u_{i-1}^{n+1} c_{i-1}^{n+1}. \end{aligned} \quad (4.7)$$

From equation (4.2) it follows that $S_i^{n+1} + \frac{\Delta t}{\phi \Delta x} u_i^{n+1} = S_i^n + \frac{\Delta t}{\phi \Delta x} u_{i-1}^{n+1}$. This means that if $c_{i-1}^{n+1} = \bar{c}$ then $c_i^{n+1} = \bar{c}$ as well. Following proof by induction, it satisfies to show that $c_1^{n+1} = \bar{c}$ in order to conclude that $c_i^{n+1} = \bar{c}$ for $i = 1 \dots I$. In the first cell $i = 1$ we have

$$\left[S_1^{n+1} + \frac{\Delta t}{\phi \Delta x} u_1^{n+1} \right] c_1^{n+1} = S_1^n \bar{c} + \frac{\Delta t}{\phi \Delta x} u_0^{n+1} c_0^{n+1}.$$

Substituting the boundary condition $c_0 = \bar{c}$ indeed shows that

$$\begin{aligned} \left[S_1^{n+1} + \frac{\Delta t}{\phi \Delta x} u_1^{n+1} \right] c_1^{n+1} &= \left[S_1^n + \frac{\Delta t}{\phi \Delta x} u_0^{n+1} \right] \bar{c} \\ c_1^{n+1} &= \bar{c} \end{aligned}$$

and therefore by induction that

$$c_0^{n+1} = \dots = c_{I-1}^{n+1} = c_I^{n+1} = \bar{c}.$$

In cell $I + 1$ the solution is given by

$$\begin{aligned} S_{I+1}^{n+1} c_{I+1}^{n+1} &= S_{I+1}^n c_{I+1}^n - \frac{\Delta t}{\phi \Delta x} (u_{I+1}^{n+1} c_{I+1}^{n+1} - u_I^{n+1} c_I^{n+1}) \\ \left[S_{I+1}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1} \right] c_{I+1}^{n+1} &= \frac{\Delta t}{\phi \Delta x} u_I^{n+1} c_I^{n+1} = \frac{\Delta t}{\phi \Delta x} u_I^{n+1} \bar{c}. \end{aligned}$$

so

$$\begin{aligned} c_{I+1}^{n+1} &= \frac{\frac{\Delta t}{\phi \Delta x} u_I^{n+1}}{S_{I+1}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1}} \bar{c} = \frac{S_{I+1}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1} - S_{I+1}^n}{S_{I+1}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1}} \bar{c} \\ &= \left[1 - \frac{S_{I+1}^n}{S_{I+1}^{n+1} + \frac{\Delta t}{\phi \Delta x} u_{I+1}^{n+1}} \right] \bar{c} \leq \bar{c} \end{aligned}$$

since $S_i^{n+1} \geq S_i^n$ for all i . If $S_{I+1}^n = 0$ we obtain $c_{I+1}^{n+1} = \bar{c}$.

This shows that $c_i^{n+1} \geq c_{i+1}^{n+1}$ for all i , hence the implicit scheme is monotonicity-preserving.

4.1.4. High-resolution schemes

High-resolution schemes were implemented to reduce the numerical diffusion of the first order methods and to improve the approximation of the discontinuity without introducing non-monotonous behaviour in the form of oscillations. However, in some situations the first order explicit and semi-implicit methods already show non-monotonous behaviour. Instead of reducing this behaviour, the high-res scheme enhances the effects, which results in oscillations. If the underlying first order method is monotonicity-preserving, this oscillatory behaviour disappears and the high-resolution method works as intended.

Ensuring the first order method is monotonous is therefore enough to obtain a monotonous high-resolution scheme. Unfortunately, the only first order method that can guarantee a monotone solution is the implicit scheme. Thee implicit high-resolution method is the least accurate and computationally most expensive method of the three, so it should be avoided where possible.

This calls for the development of a new method that produces a monotone solution with an accuracy close to that of the semi-implicit high-resolution method.

4.2. Monotonicity preserving scheme

The idea in high-resolution methods is to switch from an oscillatory second order scheme to a non-oscillatory first order scheme around discontinuities in order to obtain accurate, non-oscillatory solutions. Applying the same reasoning to our monotonicity problem, we can switch from a more accurate non-monotonous scheme to a less accurate monotonous scheme where necessary, in order to obtain an overall accurate and monotonous scheme. From the analysis it was clear that the explicit method is never monotonous, the semi-implicit method is conditionally monotonous (in case $S_w^0 > 0$) and the implicit method is unconditionally monotonous. Therefore a logical choice is to combine the non-monotonous semi-implicit high resolution scheme with a monotonous implicit scheme. Since the semi-implicit scheme is only non-monotonous around the tracer discontinuity, where high-resolution methods will switch to the first order fluxes anyway, implementing the computationally expensive high-resolution implicit scheme has no advantages over the simpler and less expensive first order implicit method. So the non-monotonous semi-implicit high resolution scheme is combined with the monotonous first order implicit scheme, resulting in a partially implicit method.

This idea of constructing a partially implicit scheme is already used for problems where the stability constraint of the explicit method leads to problems [5]. In these cases a local stability constraint is used to determine the degree of implicitness necessary to ensure an accurate and globally stable solution. In this case a local monotonicity constraint has to be constructed to guarantee a globally monotone solution.

A partially implicit scheme can generally be written as

$$(Sc)_i^{n+1} = (Sc)_i^n - \frac{\Delta t}{\phi \Delta x} (1 - \beta_i) [F_{i+1/2}^n - F_{i-1/2}^n] - \frac{\Delta t}{\phi \Delta x} \beta_i [F_{i+1/2}^{n+1} - F_{i-1/2}^{n+1}], \quad (4.8)$$

where β_i denotes the degree of implicitness in cell i , $F_{i\pm 1/2}^n$ are the explicit, or in this case semi-implicit, high-resolution fluxes:

$$F_{i+1/2,j}^n = c_i^n u_i^{n+1} + \frac{\Phi(\theta_{i+1/2}^n)}{2} u_i^{n+1} \left(1 - u_i^{n+1} \frac{\Delta t}{\Delta x} \right) [c_{i+1}^n - c_i^n].$$

and $F_{i\pm 1/2}^{n+1}$ are the first order implicit fluxes

$$F_{i+1/2}^{n+1} = c_i^{n+1} u_i^{n+1}.$$

Taking $\beta_i = 1$ for all i gives the first order implicit scheme while $\beta_i = 0$ results in the semi-implicit high-resolution scheme. The values β_i need to be chosen such that the resulting scheme is globally monotonous. We assume for now that β can only be zero or one. The semi-implicit high-resolution scheme becomes non-monotonous if the underlying first order semi-implicit scheme becomes non-monotonous. This is the case in the cells i where

$$S_i^n - \frac{\Delta t}{\phi \Delta x} u_i^{n+1} < 0, \quad (4.9)$$

therefore we choose $\beta_i = 1$ in the cells where (4.9) holds and $\beta_i = 0$ elsewhere. This results in a globally monotonous scheme.

Note that for high initial water saturations the statement (4.9) is false for all i and we simply obtain the semi-implicit high-resolution method. The partially implicit method is only designed such that a monotonous solution scheme is obtained in all cases, also when the initial water saturation is relatively low.

4.2.1. Accuracy, stability and efficiency

For the situation shown in Figure 4.1b the solution obtained by the partially implicit scheme is shown in Figure 4.4. The oscillations at the front have disappeared, and the accuracy seems to match the accuracy of the semi-implicit scheme. However, the accuracy is way lower than in Figure 4.5, where the initial water saturation is ten times higher. This can easily be explained by looking at the underlying solution of the water saturation. For low S_w^0 the tracer front and the water front are very close to each other. Since the water front is severely smeared by the implicit numerical method, the accuracy of the water saturation becomes less towards the front. The tracer concentration is calculated by a numerical scheme that involves both the water saturation and velocity, so the less accurate the water saturation, the less accurate the resulting tracer concentration. This is confirmed by the log-log plot of the error of the partially implicit scheme versus the amount of grid cells N is shown in Figure 4.6. Only for a very fine grid the accuracy of the method with a low initial water saturation is better than that of the same method with a high initial water saturation. Another conclusion from Figure 4.6 is that the partially implicit method is approximately first order accurate for $S_w^0 = 0.015$.

The analysis prior to the construction of our partially implicit scheme assumes that the CFL condition (3.11) is satisfied. So the partially implicit scheme at least requires that the necessary stability condition of the semi-implicit method is satisfied.

To be able to say something about the efficiency of the partially implicit method, we look at the implementation of (4.8). Since first order implicit fluxes are used for $F_{i\pm 1/2}^{n+1}$, the scheme is linear in the unknown c_i^{n+1} and can be written in matrix-vector notation as

$$A \mathbf{c}^{n+1} = B(\mathbf{c}^n) = b, \quad (4.10)$$

where matrix A has values $\phi S_i^{n+1} + \beta_i \frac{\Delta t}{\Delta x} u_i^{n+1}$ on the diagonal and $-\beta_i \frac{\Delta t}{\Delta x} u_{i-1}^{n+1}$ on the off-diagonal. Matrix $B(\mathbf{c}^n)$ can split into $B_1 \mathbf{c}^n$ and $B_2(\mathbf{c}^n)$, where B_1 represents the first order part with $\phi S_i^n + (1 - \beta_i) \frac{\Delta t}{\Delta x} u_i^{n+1}$ on the diagonal and $-(1 - \beta_i) \frac{\Delta t}{\Delta x} u_i^{n+1}$ on the off-diagonal. The second part, $B_2(\mathbf{c}^n)$, contains the high-resolution correction terms, which are nonlinear in the known vector \mathbf{c}^n . After constructing the right hand side of (4.10), the unknown solution \mathbf{c}^{n+1} can be found by solving the resulting linear system. This method thus requires more operations than the semi-implicit high-resolution method, but since the resulting system is still linear the method remains less expensive than the fully implicit high-resolution scheme. This is confirmed by the CPU times shown in Table 4.1. Since there are only a few nonzero β_i and therefore only few elements on the off-diagonal of matrices A and B , one would expect that for $N \rightarrow \infty$ the CPU time of the partially implicit method converges to that of the semi-implicit high-resolution method. However, the full vector β and resulting matrix diagonals are constructed first before the matrices A and B are made sparse. This implementation is quite expensive and keeps the CPU time higher than that of the semi-implicit high-resolution method.

It can be concluded that the partially implicit method is a simple and effective method that requires very few computations and always produces a monotone solution. However, although the computation time is very low compared to that of the flow solver, there is some room for improvement in terms of efficiency of the implementation.

N	CPU-time [s]		
	Semi-implicit high-res method	Partially implicit method	Implicit high-res method
10	9.5823e-04	2.0739e-03	4.0240e-02
40	3.7404e-03	9.1873e-03	2.0566e-01
100	1.1345e-02	2.8697e-02	5.1546e-01
400	9.0978e-02	2.4553e-01	1.7706e-00
1000	4.6072e-01	1.6027e-00	5.8027e-00

Table 4.1: CPU-times of the semi-implicit high-resolution, partially implicit and implicit high-resolution method for increasing amount of grid cells.

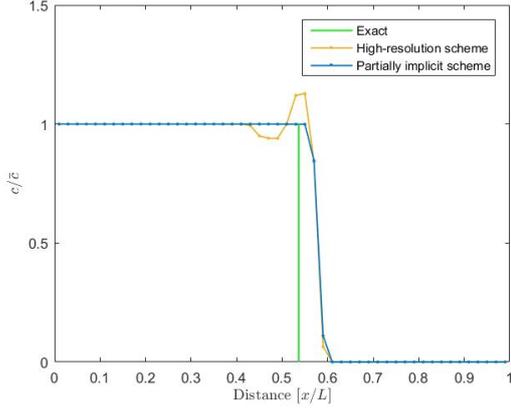


Figure 4.4: Partially-implicit approximation of $c(x, t)/\bar{c}$ at a certain time $t > 0$ for a low initial water saturation, $S_w^0 = S_{wc} = 0.015$.

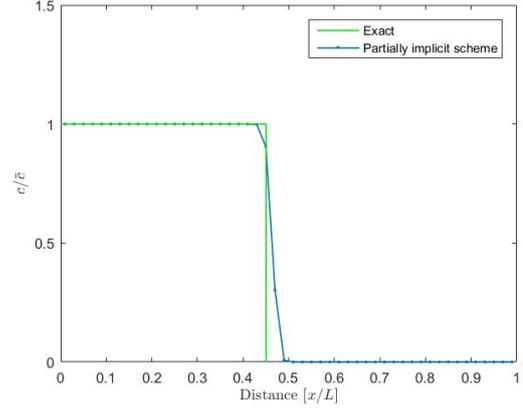


Figure 4.5: Partially-implicit approximation of $c(x, t)/\bar{c}$ at a certain time $t > 0$ for a high initial water saturation, $S_w^0 = S_{wc} = 0.15$.

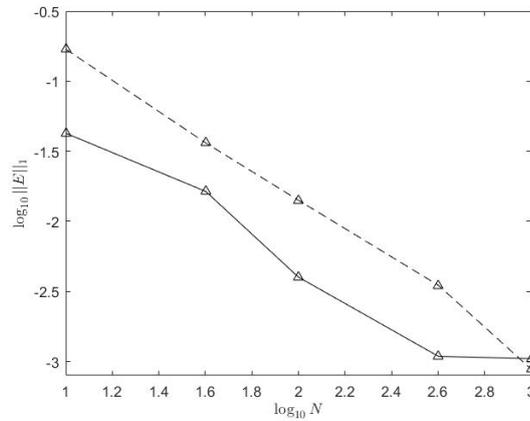


Figure 4.6: Log-log plot of the 1-norm error versus the amount of grid cells N for the partially implicit method with $S_w^0 = 0.15$ (solid) and $S_w^0 = 0.015$ (dashed).

4.2.2. Extension to two dimensions

The partially implicit method is extended to two dimensions by combining the two-dimensional semi-implicit high-resolution method with the two-dimensional first order implicit method. In this way the following fully multidimensional monotonicity-preserving method is obtained:

$$(Sc)_{ij}^{n+1} = (Sc)_{ij}^n - \frac{\Delta t}{\phi \Delta x} (1 - \beta_{ij}) [F_{i+1/2}^n - F_{i-1/2}^n] - \frac{\Delta t}{\phi \Delta y} (1 - \beta_{ij}) [G_{i,j+1/2}^n - G_{i,j-1/2}^n] \quad (4.11)$$

$$- \frac{\Delta t}{\phi \Delta x} \beta_{ij} [F_{i+1/2,j}^{n+1} - F_{i-1/2,j}^{n+1}] - \frac{\Delta t}{\phi \Delta y} \beta_{ij} [G_{i,j+1/2}^{n+1} - G_{i,j-1/2}^{n+1}] \quad (4.12)$$

with

$$F_{i+1/2,j}^n = c_{ij}^n u_{x,i}^{n+1} + \frac{\Phi(\theta_{i+1/2,j}^n)}{2} u_{x,i}^{n+1} \left(1 - u_{x,i}^{n+1} \frac{\Delta t}{\Delta x}\right) [c_{i+1,j}^n - c_{ij}^n]$$

$$G_{i,j+1/2}^n = c_{ij}^n u_{y,j}^{n+1} + \frac{\Phi(\theta_{i,j+1/2}^n)}{2} u_{y,j}^{n+1} \left(1 - u_{y,j}^{n+1} \frac{\Delta t}{\Delta y}\right) [c_{i,j+1}^n - c_{ij}^n]$$

$$F_{i+1/2,j}^{n+1} = c_{ij}^{n+1} u_{x,i}^{n+1}$$

$$G_{i,j+1/2}^{n+1} = c_{ij}^{n+1} u_{y,j}^{n+1}$$

As in the one-dimensional case, the two-dimensional high-resolution method becomes oscillatory due to non-monotonicity of the underlying first-order method. From the analysis in Appendix B it follows that the

two-dimensional semi-implicit first order scheme becomes non-monotonous in cells (i, j) where

$$S_{ij}^n - \frac{\Delta t}{\phi} \left[\frac{u_{x,i}^{n+1}}{\Delta x} + \frac{u_{y,j}^{n+1} - u_{y,j-1}^{n+1}}{\Delta y} \right] < 0$$

or

$$S_{ij}^n - \frac{\Delta t}{\phi} \left[\frac{u_{x,i}^{n+1} - u_{x,i-1}^{n+1}}{\Delta x} + \frac{u_{y,j}^{n+1}}{\Delta y} \right] < 0$$

In these cells we choose $\beta_{ij} = 1$ and elsewhere $\beta_{ij} = 0$. This again results in a globally monotonous scheme.

5

Transport of polymer

As introduced in Chapter 3, active tracers are chemical substances that influence the flow by changing the physical properties of the host fluid. Typical examples are the substances involved in chemical EOR techniques, where chemicals are added to the injected water to improve the efficiency of the flood. This section describes the numerical modeling and simulation of a widely used chemical EOR technique, polymer flooding. Polymer is water soluble and classified as a non-partitioning active tracer. It is added to the injected water to increase its viscosity and hereby enhance its ability to push oil through the rock due to a more favourable mobility ratio between the injected water and the oil [12]. This effect is most apparent when the water is extremely mobile compared to the oil, e.g., in case of very viscous oil. Since the residual water saturation does not change during a conventional polymer flood, both water flooding and polymer flooding will theoretically produce all the movable oil over a very long timescale. So in principle the benefit of polymer is only the acceleration of the oil production by delaying the water breakthrough. In practice however, the timescale of production is limited by economical considerations, and the acceleration can lead to a significant increase in the ultimate oil recovery as shown in Figure 5.1 .

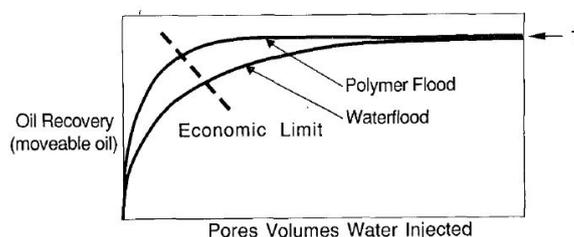


Figure 5.1: Production profiles for a water flood and a polymer flood showing the benefit of the polymer flood at the economical limit [21].

Since the water viscosity is strongly influenced by the polymer concentration, it is crucial that polymer fronts are solved accurately in order to capture the displacement mechanisms correctly. This is essential for an accurate prediction of the enhanced oil recovery effect due to the polymer flood. As will be shown in this chapter and as shown previously by among others Mykkeltvedt (2014), first order methods fail in this area since their large numerical diffusion tends to smear the polymer bank. As in Chapter 3, high-resolution methods are applied to reduce the numerical diffusion and resolve the polymer bank more sharply.

During polymer flooding there is usually also adsorption of polymer onto the rock. This adsorption potentially reduces the effective water permeability and decreases the polymer concentration which results in a delay of the polymer front. These effects are only temporary and last until the maximum level of adsorption has been reached. This and the fact that adsorption is not known for causing any mathematical problems has ensured that adsorption of polymer is omitted from this research.

Another effect that occurs during polymer flooding and that does cause some mathematical issues is the velocity enhancement effect, the fact that polymer travels faster than water, also referred to as *hydrodynamic acceleration*. A physical explanation of this effect and several ways to model it will be given at the end of this chapter. We start by deriving and analysing the polymer flooding model without hydrodynamic acceleration.

5.1. Model equations

The transport equation for polymer is similar to that for the passive tracer given in (3.1). The only difference is that the water velocity now depends on the polymer concentration, since polymer influences the water mobility. We thus obtain the following coupled system of non-linear conservation equations for oil, water and polymer:

$$\phi \frac{\partial}{\partial t} (S_\alpha) + \nabla \cdot (\mathbf{u}_\alpha(S_\alpha, c_p)) = 0, \quad \alpha \in \{o, w\} \quad (5.1)$$

$$\phi \frac{\partial}{\partial t} (S_w c_p) + \nabla \cdot (c_p \mathbf{u}_w(S_w, c_p)) = 0, \quad (5.2)$$

where c_p is the polymer concentration. Rewriting this into fractional flow formulation we obtain

$$\nabla \cdot \mathbf{u}_T = 0$$

$$\phi \frac{\partial}{\partial t} (S_w) + \mathbf{u}_T \nabla \cdot (f_w(S_w, c_p)) = 0 \quad (5.3)$$

$$\phi \frac{\partial}{\partial t} (S_w c_p) + \mathbf{u}_T \nabla \cdot (c_p f_w(S_w, c_p)) = 0 \quad (5.4)$$

where $f_w(S_w, c_p)$ is the water fractional flow in the presence of polymer given by

$$f_w(S_w, c_p) = \frac{\lambda_w(c_p)}{\lambda_w(c_p) + \lambda_o} = \frac{k_{rw}(S_w)}{k_{rw}(S_w) + \frac{\mu_w(c_p)}{\mu_o} k_{ro}(S_w)}. \quad (5.5)$$

A higher polymer concentration shifts the fractional flow curve to the right. An example of this is shown in Figure 5.2.

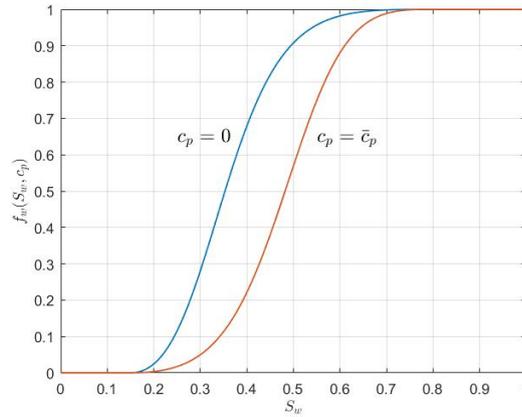


Figure 5.2: Fractional flow curves for water without polymer ($c_p = 0$) and water with a fixed polymer concentration \bar{c}_p .

The dependence of the water viscosity on the polymer concentration is typically modelled by the Flory-Huggins equation [12]:

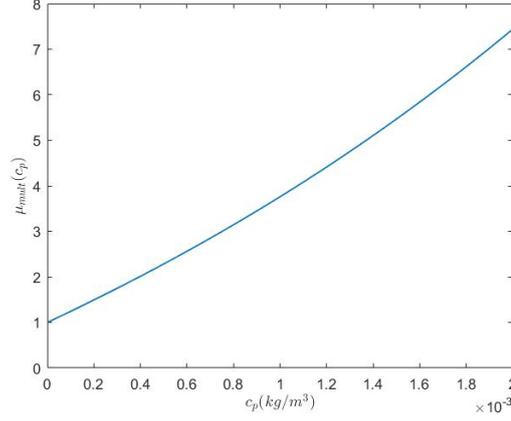
$$\mu_w(c_p) = \mu_w^0 \left(1 + \alpha_1 c_p + \alpha_2 c_p^2 + \alpha_3 c_p^3 \right) = \mu_w^0 \mu_{\text{mult}}(c_p), \quad (5.6)$$

with μ_w^0 the original water viscosity without polymer, μ_{mult} the viscosity multiplier function and α_i constants. Looking at the mobility ratio M , defined as

$$M = \frac{\lambda_o}{\lambda_w} = \frac{\mu_o k_{rw}}{\mu_w k_{ro}}$$

we indeed see that the effect of polymer on the mobility is greatest when the initial ratio μ_o/μ_w^0 is high.

A typical viscosity multiplier function is shown in Figure 5.3, where we selected $\alpha_1 = 24$, $\alpha_2 = 31$ and $\alpha_3 = 50$ and assumed that water with 0.2% polymer was injected.

Figure 5.3: Typical viscosity multiplier function $\mu_{\text{mult}}(c_p)$.

5.2. Analytical solution

In 1D equations (5.3) and (5.4) turn into

$$\begin{aligned} \phi \frac{\partial}{\partial t}(S_w) + u_T \frac{\partial}{\partial x}(f_w(S_w, c_p)) &= 0 \\ \phi \frac{\partial}{\partial t}(S_w c_p) + u_T \frac{\partial}{\partial x}(c_p f_w(S_w, c_p)) &= 0 \end{aligned} \quad (5.7)$$

Analytical solutions for S_w and c_p on $x \in \mathbb{R}$ are derived for polymer flooding in both secondary and tertiary mode. For polymer flooding in secondary mode the same Riemann initial data is imposed on the water saturation S_w as was done for the water flooding

$$S_w(x, 0) = \begin{cases} S_w^{\text{inj}} = 1 - S_{or} & x < 0 \\ S_w^0 & x \geq 0. \end{cases} \quad (5.8)$$

and on c_p the same Riemann initial data is imposed as on the passive tracer:

$$c_p(x, 0) = \begin{cases} \bar{c}_p & x < 0 \\ 0 & x \geq 0. \end{cases} \quad (5.9)$$

As in Section 2.2 we assume $u_T > 0$, such that the solution for $x > 0$ represents a reservoir on $x > 0$ where from $t = 0$ water with a constant polymer concentration \bar{c}_p is injected at $x = 0$. This initial polymer concentration profile travels through the reservoir and at the same time influences the water saturation profile. During a polymer flood generally two saturation shocks form [17], one at the polymer concentration front where water with polymer contacts water without polymer and one shock where the water saturation jumps to its initial value.

The saturation S^{*1} corresponding to the polymer front can be found in a similar way as for the passive tracer, namely by equating the the characteristic velocity of the polymer concentration to the characteristic velocity of the water containing polymer. By introducing the fractional flow function $f_{wp}(S_w) = f_w(S_w, \bar{c}_p)$, the characteristic velocities of the polymer concentration and the water in the presence of polymer can be written as

$$\begin{aligned} \left(\frac{dX}{dt} \right)_{c_p} &= \frac{u_T f_{wp}(S_w)}{\phi S_w} \\ \left(\frac{dX}{dt} \right)_{S_w} &= \frac{u_T}{\phi} \frac{df_{wp}}{dS_w}(S_w). \end{aligned}$$

Thus S^{*1} can be found by solving

$$\frac{f_{wp}(S^{*1})}{S^{*1}} = \frac{df_{wp}}{dS_w}(S^{*1}).$$

The water saturation right after the shock, S^{*2} is found from the jump conditions across the shock:

$$\frac{f_{wp}(S^{*1})}{S^{*1}} = \frac{f_{wp}(S^{*1}) - f_w(S^{*2}, 0)}{S^{*1} - S^{*2}}.$$

From S^{*2} the water saturation jumps to the initial water saturation $S_w^0 = S_{wc}$. The resulting analytic solution together with the schematic illustration on how to find the shock saturations is shown in Figure 5.4. For a more detailed derivation of this analytical solution, see Appendix A.

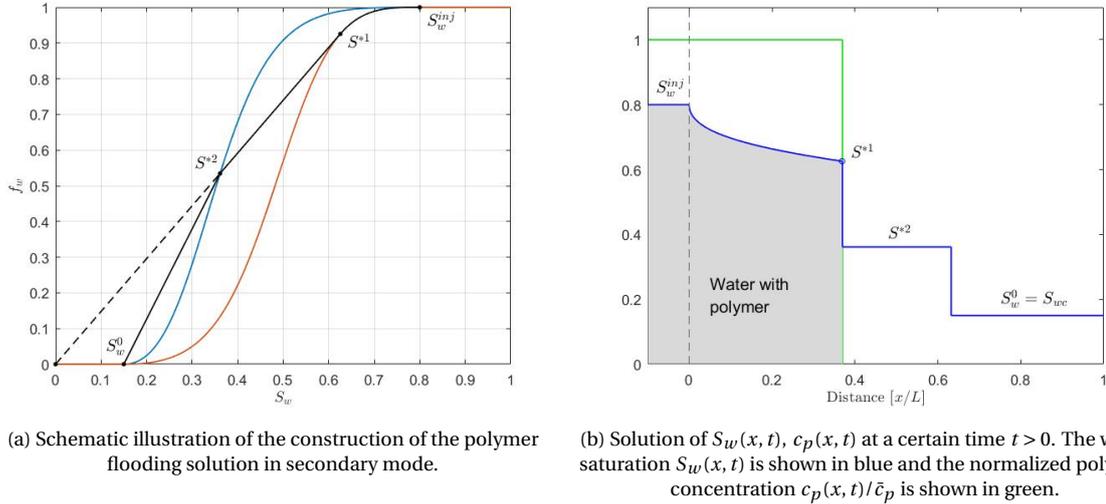


Figure 5.4: Analytical solution of polymer flooding in secondary mode for certain time $t > 0$.

For polymer flooding in tertiary mode the injection of polymer starts after some period T of water flooding. For simplicity it is assumed that after water flooding, the water saturation in the reservoir is uniform, i.e. $S_w(x, T) = S_w^0$ for some constant value S_w^0 with $f_w(S_w^0, 0)$ near one. Looking at the analytical solution for $t \geq T$ compared to the solution in Figure 5.4b, the only difference is that the water saturation jumps from S^{*2} to a higher value S_w^0 instead of to the connate water saturation S_{wc} , see Figure 5.5.

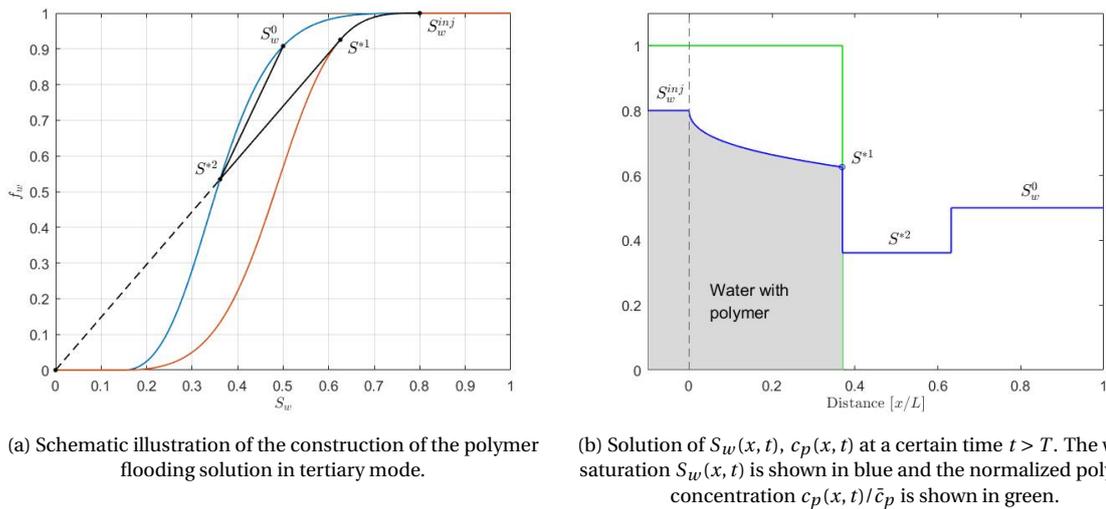


Figure 5.5: Analytical solution of polymer flooding in tertiary mode for certain time $t > T$.

5.3. Numerical solution

For more complicated problems than the ones considered in this research, the polymer flooding must be solved numerically. Since the polymer concentration influences the flow, the equations become coupled and can't be solved separately as in section 3.2. Instead they must be solved either simultaneously or sequentially each time step. In MRST the simultaneous approach is chosen and the equations for oil, water and polymer (equations (5.1) and (5.2)) are solved together using a fully implicit first upwind order method. In one dimension, this results in the following discretization scheme

$$\begin{cases} S_{o,i}^{n+1} &= S_{o,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_o \left(S_{o,i}^{n+1}, c_i^{n+1} \right) - u_o \left(S_{o,i-1}^{n+1}, c_{i-1}^{n+1} \right) \right] \\ S_{w,i}^{n+1} &= S_{w,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_w \left(S_{w,i}^{n+1}, c_i^{n+1} \right) - u_w \left(S_{w,i-1}^{n+1}, c_{i-1}^{n+1} \right) \right] \\ (S_w c)_i^{n+1} &= (S_w c)_i^n - \frac{\Delta t}{\phi \Delta x} \left[u_w \left(S_{w,i}^{n+1}, c_i^{n+1} \right) c_i^{n+1} - u_w \left(S_{w,i-1}^{n+1}, c_{i-1}^{n+1} \right) c_{i-1}^{n+1} \right] \end{cases}$$

where $c = c_p$ is the polymer concentration. This scheme is nonlinear in both S_w^{n+1} and c^{n+1} and must be solved using Newton's method, which is computationally expensive. Also, since a first order implicit method is used, the accuracy of the approximations is low.

In this research the focus is on decoupling the flow and transport, so a sequential approach is implemented as well. Each time step the flow equations are solved using the polymer concentration from the previous time step and subsequently the polymer equation is solved using the newly obtained flow solution. A single time step from t^n to t^{n+1} thus consists of the following steps:

1) Solve the flow equations: compute S_w^{n+1} using c^n via the fully implicit nonlinear discretization scheme

$$\begin{cases} S_{o,i}^{n+1} &= S_{o,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_o \left(S_{o,i}^{n+1}, c_i^n \right) - u_o \left(S_{o,i-1}^{n+1}, c_{i-1}^n \right) \right] \\ S_{w,i}^{n+1} &= S_{w,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_w \left(S_{w,i}^{n+1}, c_i^n \right) - u_w \left(S_{w,i-1}^{n+1}, c_{i-1}^n \right) \right] \end{cases}$$

2) Solve the polymer transport equation: compute c^{n+1} using S_w^{n+1}

$$(S_w c)_i^{n+1} = (S_w c)_i^n - \frac{\Delta t}{\phi \Delta x} \left[F_{i+1/2} \left(S_w^{n+1}, c^n, c^{n+1} \right) - F_{i-1/2} \left(S_w^{n+1}, c^n, c^{n+1} \right) \right].$$

In the first step the system of equations is solved using Newton's method. The solution scheme of the second step depends on how the fluxes $F_{i\pm 1/2}$ are chosen. Two first order options followed by two high-resolution options are discussed and compared.

5.3.1. First order methods

Since the latest flow solution is used to solve the polymer transport equations, we only consider two first order fluxes in stead of three as in Section 3.2:

- Explicit:

$$F_{i+1/2} = F_{i+1/2}^n = u_{w,i}^{n+1} c_i^n = u_w \left(S_{w,i}^{n+1}, c_i^n \right) c_i^n \quad (5.10)$$

- Implicit:

$$F_{i+1/2} = F_{i+1/2}^{n+1} = u_{w,i}^{n+1} c_i^{n+1} = u_w \left(S_{w,i}^{n+1}, c_i^n \right) c_i^{n+1}. \quad (5.11)$$

The water velocity u_w^{n+1} is known from the first step of our sequential procedure, so both approximations result in a linear expression of the unknown c_i^{n+1} . Note that the implicit flux is not fully implicit since the velocity is still based on c_i^n and not on c_i^{n+1} .

Numerical solutions for the polymer flooding situations shown in Figure 5.4b and 5.5b are shown in Figure 5.7. Similar to what we saw in Section 3.2, the first order upwind methods introduce a great deal of numerical diffusion, causing severe smearing of the polymer concentration front. Because the flow solution depends strongly on the polymer concentration, this smearing is also visible in the solution of $S_w(x, t)$. The explicit method performs better than the implicit method, but both methods fail to accurately capture the water bank. Refining the grid ten times (and scaling the time steps appropriately) severely reduces the numerical diffusion, but the polymer concentration, and thereby the first shock in the water saturation, remains smeared. Furthermore, the resolution of the second shock has greatly improved. Around this shock no polymer is present, so the solution is less sensitive to errors in the polymer concentration.

Note that we have now showed the solutions for two very specific cases of polymer flooding. For different situations with a less distinctive profile, the first order methods can fail to capture the intermediate water bank completely, even on fine grids. It can therefore be concluded that first order methods fail to capture the effect of polymer accurately, given a reasonable amount of grid cells and time steps. To improve the approximations, high-resolution methods are implemented.

Results were also compared to the simultaneous approximation. But even on coarse grids, there is little difference between the simultaneous approximation and the first order implicit sequential approximation.

5.3.2. High-resolution methods

As in section 3.2, flux-limiting high-resolution methods are derived using the first order and second order upwind fluxes. This results in the following high-resolution fluxes:

- Explicit High-Resolution:

$$F_{i+1/2} = F_{i+1/2}^n = u_{w,i}^{n+1} c_i^n + \Phi(\theta_{i+1/2}^n) \frac{1}{2} u_{w,i}^{n+1} \left(1 - u_{w,i}^{n+1} \frac{\Delta t}{\Delta x} \right) [c_{i+1}^n - c_i^n] \quad (5.12)$$

- Implicit High-Resolution:

$$F_{i+1/2} = F_{i+1/2}^{n+1} = u_{w,i}^{n+1} c_i^{n+1} + \Phi(\theta_{i+1/2}^{n+1}) \frac{1}{2} u_{w,i}^{n+1} \left(1 - u_{w,i}^{n+1} \frac{\Delta t}{\Delta x} \right) [c_{i+1}^{n+1} - c_i^{n+1}], \quad (5.13)$$

where the van Leer limiter is again used as flux limiter. Numerical solutions for the polymer flooding situations shown in Figure 5.4b and 5.5b are shown in Figure 5.9 and 5.10. Based on these results similar conclusions can be drawn as in section 3.2.2. First of all, the explicit high-resolution scheme performs best and strongly reduces the numerical diffusion in the approximation of the polymer concentration. This also results in a more accurate approximation of the water saturation. Secondly, the fully implicit high-resolution scheme barely outperforms the first order explicit scheme and should therefore only be used in case the time step prohibits to use explicit methods.

Since reservoir simulators are used to predict and optimize oil recovery, we also look at the effect of the different methods on the total oil production over time. We compare the production resulting from a water flood with the production resulting from a secondary polymer flood approximated by a first order and high-resolution explicit method. What can be seen from Figure 5.6 is that although both methods eventually reach the same production level, the first order method overestimates the production for some period of time.

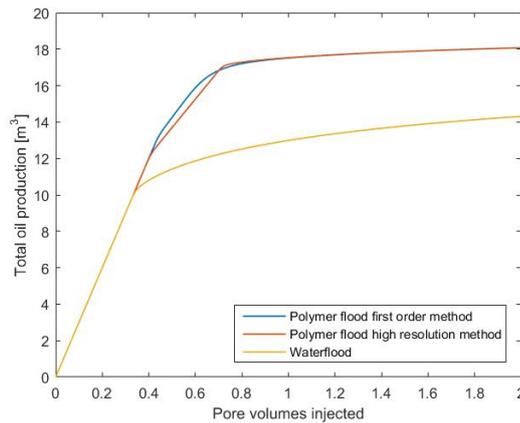


Figure 5.6: Production profiles for a polymer flood using an explicit first order method and an explicit high-resolution method compared to the production profile of a waterflood.

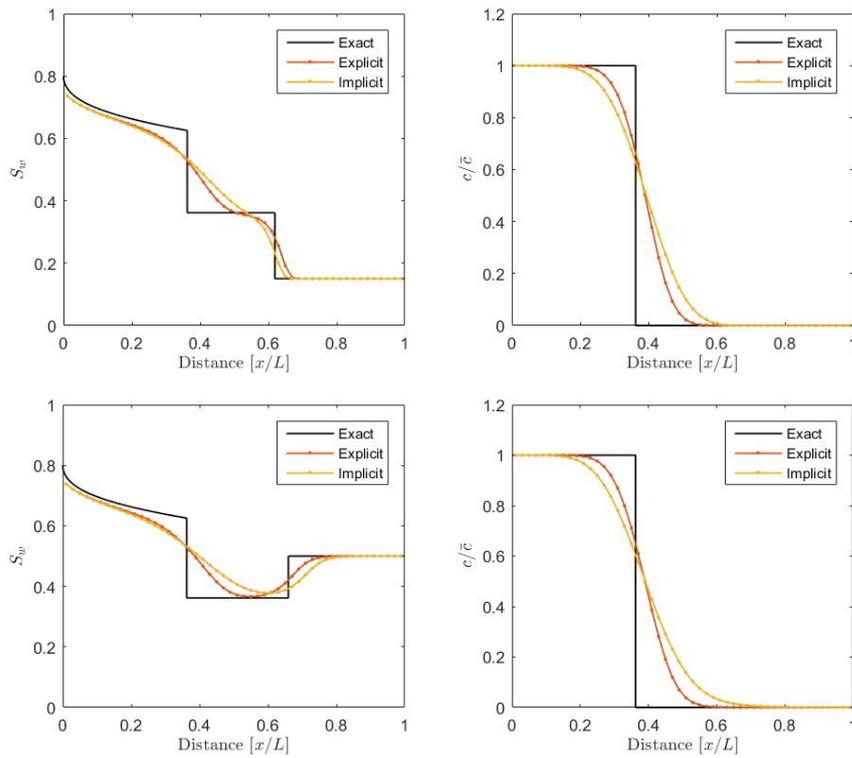


Figure 5.7: First order numerical solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 50 cells and time steps.

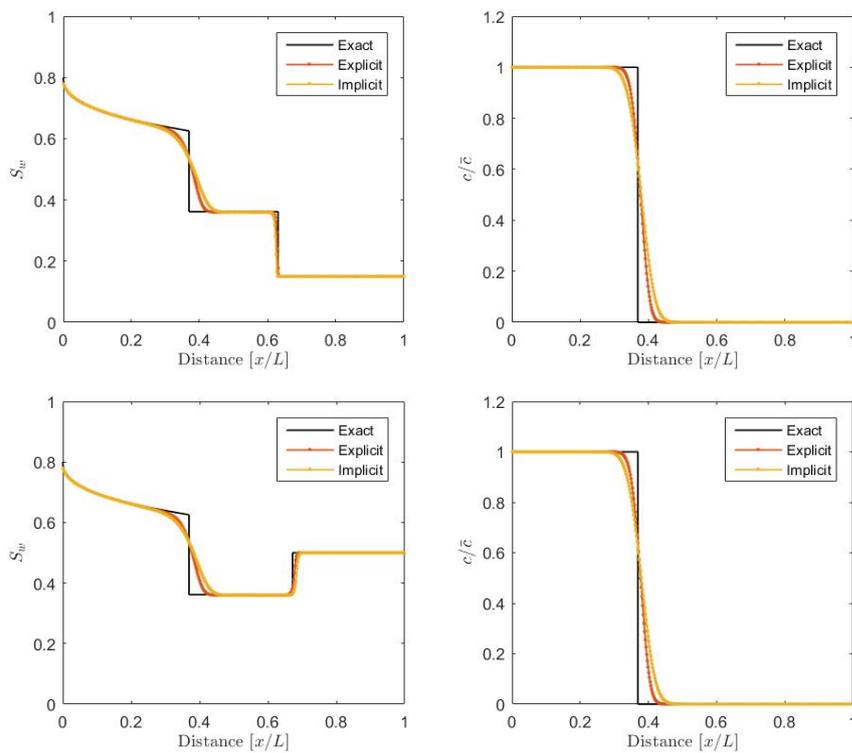


Figure 5.8: First order numerical solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 500 cells and time steps.

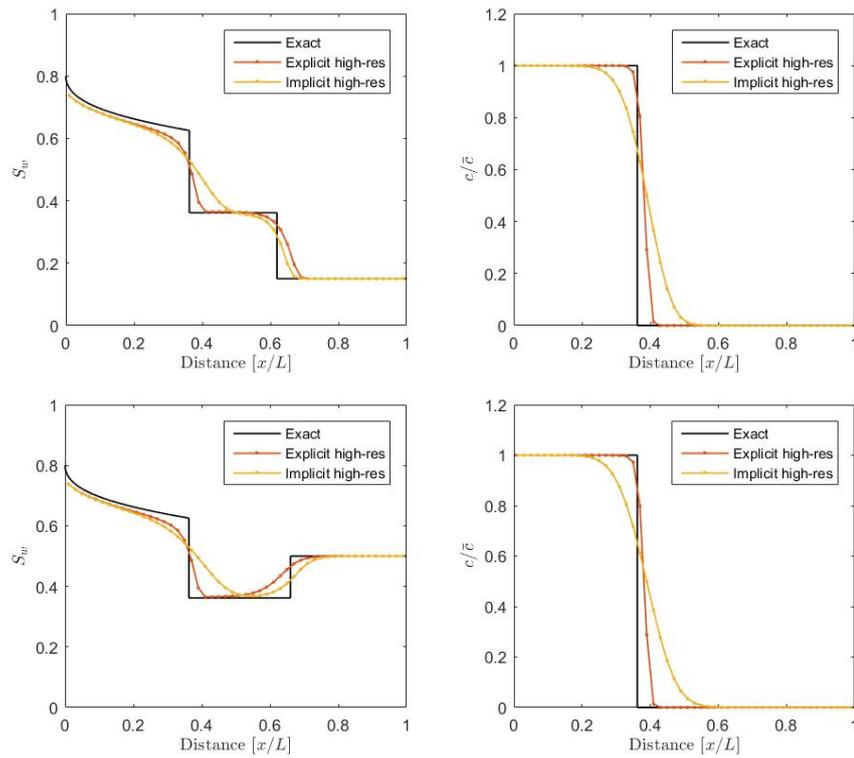


Figure 5.9: High-resolution solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 50 cells and time steps.

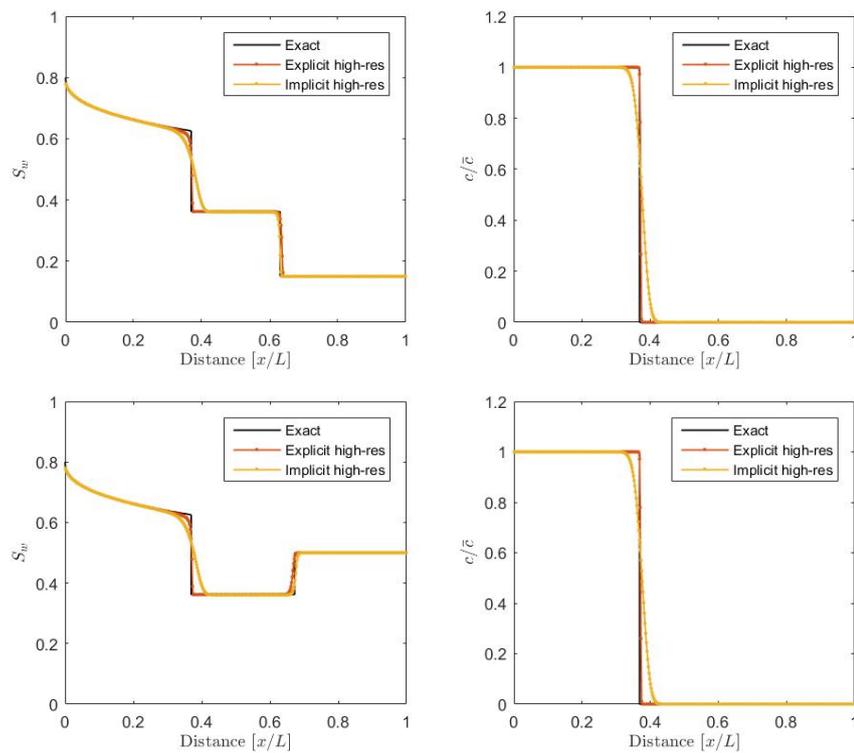


Figure 5.10: High-resolution solutions at a certain time $t > 0$ of polymer flooding in secondary mode (top) and tertiary mode (bottom). Both solved using 500 cells and time steps.

5.3.3. Accuracy, stability and efficiency

To quantify the difference in accuracy between the first order and high-resolution methods the 1-norm errors are observed for both the water saturation and the normalised polymer concentration, see Figure 5.11. The error of the simultaneous method is comparable with the error of the first order implicit method and therefore left out of the analysis. In terms of accuracy the advantage of the explicit high-resolution method is clear, but also in terms of computational costs this method performs satisfactory. As can be seen in Table 5.1, the CPU times for the explicit high-resolution method are similar to those of the explicit first order method. Furthermore it can be concluded that all sequential methods are computationally less expensive than the simultaneous first order implicit method. Since the accuracy of the simultaneous method is almost equal to the accuracy of the implicit first order sequential method, the simultaneous approach has no advantages over the sequential approach.

The CPU times of polymer flooding show a relatively small increase compared to the CPU times of water-flooding shown in Table 3.1. This indicates that most of the computational time is taken up by the flow solver and not the transport module. Looking at the CPU times of the flow and transport separately we indeed see that this is the case. For example, the explicit high-resolution scheme spends even less than 2% of the CPU time on solving the polymer transport.

Besides the accuracy and efficiency, the stability of the numerical polymer flooding scheme has to be considered as well. Since we are now dealing with a system of coupled nonlinear equations (5.7), stability analysis becomes even more complex than in the cases considered previously. First of all, the polymer transport equation in (5.7) can be rewritten in the nonconservative form

$$\frac{\partial c_p}{\partial t} + \frac{u_T f_w(S_w, c_p)}{\phi S_w} \frac{\partial c_p}{\partial x} = 0,$$

which is simply the advection equation with variable coefficients. As for the passive tracer transport equation, a local stability criterion can be found via von Neumann analysis if the coefficients are frozen at a certain point. This results in the necessary stability condition

$$C_P = \max_{S_w, c_p} \left| \frac{\Delta t}{\phi \Delta x} \frac{u_T(S_w, c_p)}{S_w} \right| = \max_{S_w, c_p} \left| \frac{\Delta t}{\Delta x} \frac{u_T}{\phi} \frac{f_w(S_w, c_p)}{S_w} \right| \leq 1. \quad (5.14)$$

Since a nonzero polymer concentration shifts the fractional flow curve f_w to the right, the maximum is attained at $c_p = 0$:

$$\max_{S_w, c_p} \left| \frac{f_w(S_w, c_p)}{S_w} \right| = \max_{S_w} \left| \frac{f_w(S_w)}{S_w} \right|$$

which means that $C_T = C_P$ and thus that the necessary stability condition for the polymer transport is the same as for the transport of a passive tracer.

Stability of the implicit scheme for the nonlinear flow equation is again difficult. As in section 2.3.3, all we can say is that the scheme is much more stable than explicit or semi-implicit alternatives and that time steps should not be taken too large to ensure convergence to the correct solution.

N	CPU-time [s]				
	Simultaneous implicit method	Explicit 1st order method	Implicit 1st order method	Explicit high-res method	Implicit high-res method
10	2.1060	1.2168	0.9204	0.8892	1.0608
40	3.6660	2.9952	3.3852	3.1200	3.6504
100	9.8749	7.9093	7.5348	7.7688	8.2525
400	41.0127	33.0566	33.4310	33.9770	35.4434
1000	121.1504	97.9218	96.9390	97.6878	99.2010

Table 5.1: CPU-times of all numerical methods applied to polymer flooding for increasing amount of grid cells.

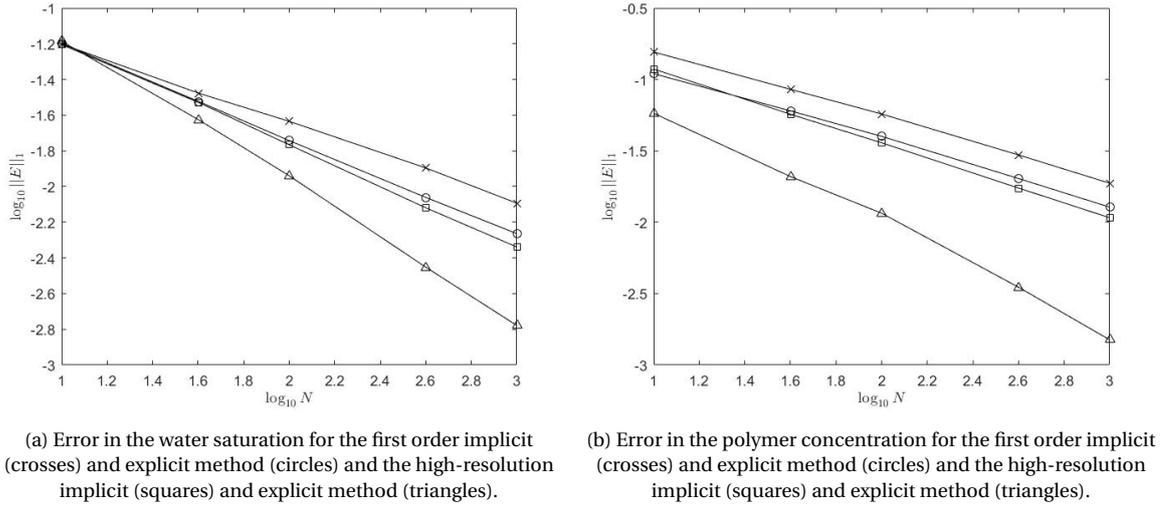


Figure 5.11: Log-log plot of the 1-norm error versus the amount of grid cells N for the water saturation (left) and the polymer concentration (right).

5.4. Hydrodynamic acceleration

Physical experiments show that polymer travels faster through water than passive tracers, meaning that the average velocity of polymer molecules is higher than the average water velocity. Two different physical explanations exist for this phenomenon. The most common explanation is that the smallest pores are inaccessible to the large polymer molecules [12], causing the polymer to move only in part of the pore space and thereby move ahead of the water. The region of the pore space which is inaccessible to polymer is called the inaccessible pore volume (IPV). The other explanation is that polymer molecules tend to move in the center part of the pore flow channels, where the flow velocity is higher than close to the pore walls. This explanation is referred to as the excluded pore volume model (EPV). Both explanations result in the fact that only part of the available pore space is used by the polymer, and can therefore be modelled the same way.

In reservoir simulators the most common way to model hydrodynamic acceleration is by introducing a constant velocity enhancement factor. As was shown by Bartelds et al. (1997), this approach leads to ill-posedness of the mathematical model which results in nonphysical solutions around the polymer front. Another model with a saturation dependent enhancement factor was proposed by Bartelds et al. and later extended by Hilden et al. (2016) to obtain a well-posed mathematical system. It will be shown here that the simple adjustment proposed by Bartelds et al. indeed results in a well-posed system. This model converges to a different solution than the model with a constant enhancement factor. Whether this solution matches the physical phenomenon we wish to capture is still an open question.

5.4.1. Constant factor

By introducing the constant velocity enhancement factor α , the system of conservation equations (5.7) turns into:

$$\begin{aligned} \phi \frac{\partial}{\partial t} (S_w) + u_T \frac{\partial}{\partial x} (f_w(S_w, c_p)) &= 0 \\ \phi \frac{\partial}{\partial t} (S_w c_p) + \alpha u_T \frac{\partial}{\partial x} (c_p f_w(S_w, c_p)) &= 0. \end{aligned} \quad (5.15)$$

This system can be rewritten into quasi-linear form as:

$$\begin{aligned} \frac{\partial}{\partial t} (S_w) + \frac{u_T}{\phi} \left(\frac{\partial f_w}{\partial S_w} \frac{\partial S_w}{\partial x} + \frac{\partial f_w}{\partial c_p} \frac{\partial c}{\partial x} \right) &= 0 \\ \frac{\partial c_p}{\partial t} + \left(\frac{\alpha u_T}{\phi} \frac{f_w}{S_w} + \frac{c_p(\alpha - 1) u_T}{\phi S_w} \frac{\partial f_w}{\partial c_p} \right) \frac{\partial c_p}{\partial x} + \left(\frac{c_p(\alpha - 1) u_T}{\phi S_w} \frac{\partial f_w}{\partial S_w} \right) \frac{\partial S_w}{\partial x} &= 0, \end{aligned} \quad (5.16)$$

which in matrix-vector notation becomes

$$\begin{pmatrix} S_w \\ c_p \end{pmatrix}_t + A \begin{pmatrix} S_w \\ c_p \end{pmatrix}_x = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (5.17)$$

with

$$A = \frac{u_T}{\phi} \begin{pmatrix} \frac{\partial f_w}{\partial S_w} & \frac{\partial f_w}{\partial c_p} \\ \frac{c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial S_w} & \frac{\alpha f_w + c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial c_p} \end{pmatrix}.$$

We will follow the approach of Bartelds et al.(1997) to show that this system is hyperbolic for $\alpha = 1$ (no velocity enhancement) but contains an elliptic region for $\alpha > 1$.

The system (5.17) is called hyperbolic if the eigenvalues of matrix A are real for all S_w, c_p , which is true if the discriminant of the quadratic equation for the eigenvalues is nonnegative [3], i.e.

$$D = \left(\frac{u_T}{\phi}\right)^2 \left[\left(\frac{\partial f_w}{\partial S_w} - \frac{\alpha f_w + c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial c_p} \right)^2 + 4 \left(\frac{c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial c_p} \frac{\partial f_w}{\partial S_w} \right) \right] \geq 0.$$

For the situation without velocity enhancement, $\alpha = 1$, this condition is always satisfied since

$$D = \left(\frac{u_T}{\phi}\right)^2 \left(\frac{\partial f_w}{\partial S_w} - \frac{f_w}{S_w} \right)^2 \geq 0.$$

From the analytical solution we know that at the polymer front $\frac{\partial f_w}{\partial S_w} = \frac{f_w}{S_w}$, meaning that $D = 0$ and thus that the two eigenvalues coincide. The set of all pairs (S_w, c_p) for which this holds is denoted by H

$$H = \left\{ (S_w, c_p) \left| \frac{\partial f_w}{\partial S_w}(S_w, c_p) = \frac{f_w(S_w, c_p)}{S_w} \right. \right\}.$$

Now let $\alpha = 1 + \varepsilon$ with $\varepsilon > 0$ sufficiently small, then we obtain

$$D = \left(\frac{u_T}{\phi}\right)^2 \left[\left(\frac{\partial f_w}{\partial S_w} - \frac{f_w + \varepsilon f_w + \varepsilon c_p \frac{\partial f_w}{\partial c_p}}{S_w} \right)^2 + 4 \left(\frac{\varepsilon c_p}{S_w} \frac{\partial f_w}{\partial c_p} \frac{\partial f_w}{\partial S_w} \right) \right].$$

For $(S_w, c_p) \in H$ this simplifies to

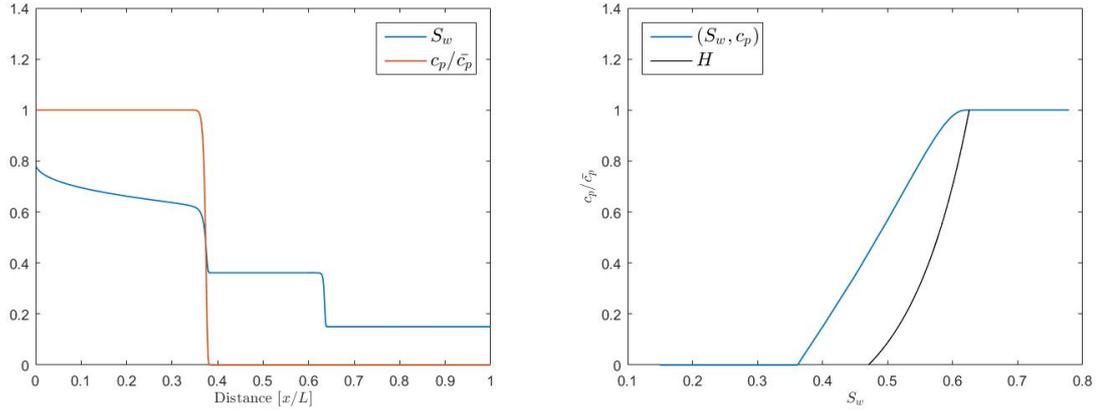
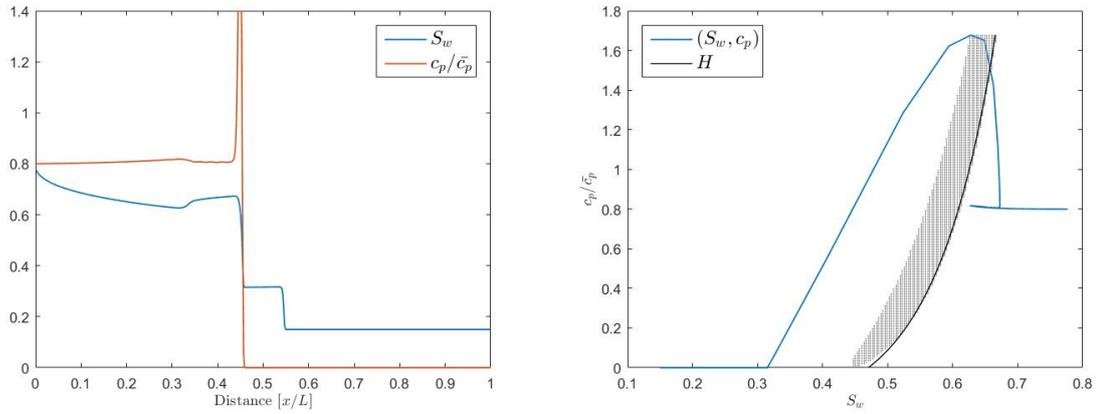
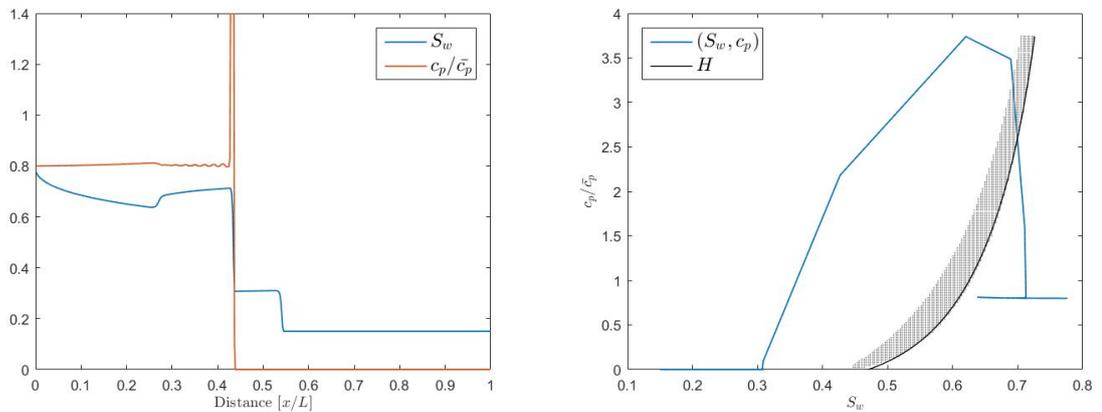
$$\begin{aligned} D &= \left(\frac{u_T}{\phi}\right)^2 \left[\left(\frac{\varepsilon f_w + \varepsilon c_p \frac{\partial f_w}{\partial c_p}}{S_w} \right)^2 + 4 \left(\frac{\varepsilon c_p}{S_w} \frac{\partial f_w}{\partial c_p} \frac{\partial f_w}{\partial S_w} \right) \right] \\ &= \left(\frac{u_T}{\phi}\right)^2 \frac{\varepsilon}{S_w} \left[\frac{\varepsilon}{S_w} \left(f_w + c_p \frac{\partial f_w}{\partial c_p} \right)^2 + 4 c_p \frac{\partial f_w}{\partial c_p} \frac{\partial f_w}{\partial S_w} \right]. \end{aligned}$$

As ε was chosen sufficiently small, the term $4c_p \frac{\partial f_w}{\partial c_p} \frac{\partial f_w}{\partial S_w}$ dominates the discriminant. Looking at the fractional flow function (5.5), it holds that

$$\frac{\partial f_w}{\partial c_p} < 0, \quad \frac{\partial f_w}{\partial S_w} \geq 0,$$

meaning that the dominant term is negative and therefore the discriminant itself is negative. This implies that both eigenvalues of our system are complex, resulting in an elliptic region around the set H and an ill-posed model.

To illustrate the consequences of this elliptic region, the secondary polymer flooding case introduced earlier in this section is considered. Results using the explicit high-resolution method for $\alpha = 1$ and $\alpha = 1.25$ are compared for two different time step sizes. Since no analytical solution is known for this model, solutions are calculated on a very fine grid. As can be seen in Figure 5.12, a constant velocity enhancement factor results in a peak in the polymer concentration at the front and slight instabilities right before the peak. The height of the peak depends on both the coarseness of the grid and the time step size. Looking at the phase-space solutions, we see that the increase in polymer concentration occurs near the elliptic region. Since the initial condition and the boundary condition (injection values) are located on opposite sides of this elliptic region, the numerical solution cannot avoid crossing this region of instability.

(a) Solution for $\alpha = 1$ (solved using 2000 time steps) in x -space (left) and phase-space (right).(b) Solution for $\alpha = 1.25$ (solved using 2000 time steps) in x -space (left) and phase-space (right).(c) Solution for $\alpha = 1.25$ (solved using 500 time steps) in x -space (left) and phase-space (right).Figure 5.12: Numerical solutions of polymer flooding in secondary mode for different constant velocity enhancement factors α using 500 grid cells. Solutions are shown in x -space (left) and in phase-space (right), where the shaded area denotes the elliptic region.

5.4.2. Saturation dependent factor

In the paper by Bartelds et al. (1997) a saturation dependent velocity enhancement factor is proposed

$$\alpha(S_w) = \frac{S_w}{S_w - S^*}, \quad (5.18)$$

where S^* is the threshold saturation needed to be reached before polymer can enter the porous medium. In the presence of adsorption, a necessary requirement for well-posedness of this model is that $S^* < S_{wc}$. Since

this requirement is quite restrictive, Hilden et al (2016) propose an extended model that is well-posed for all values of S^* . However, since our model does not contain an adsorption term, the simple approach of Bartelds results in a well-posed model for all values of S^* as will be demonstrated below.

With the saturation dependent velocity enhancement factor $\alpha(S_w)$, the system of conservation equations (5.7) is given by:

$$\begin{aligned}\phi \frac{\partial}{\partial t}(S_w) + u_T \frac{\partial}{\partial x}(f_w(S_w, c_p)) &= 0 \\ \phi \frac{\partial}{\partial t}(S_w c_p) + u_T \frac{\partial}{\partial x}(\alpha(S_w) c_p f_w(S_w, c_p)) &= 0.\end{aligned}\tag{5.19}$$

In this case the quasilinear form turns into:

$$\begin{aligned}\frac{\partial}{\partial t}(S_w) + \frac{u_T}{\phi} \left(\frac{\partial f_w}{\partial S_w} \frac{\partial S_w}{\partial x} + \frac{\partial f_w}{\partial c_p} \frac{\partial c}{\partial x} \right) &= 0 \\ \frac{\partial c_p}{\partial t} + \frac{u_T}{\phi} \left(\frac{\alpha f_w}{S_w} + \frac{c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial c_p} \right) \frac{\partial c_p}{\partial x} + \frac{u_T}{\phi} \left(\frac{c_p f_w}{S_w} \frac{\partial \alpha}{\partial S_w} + \frac{c_p(\alpha-1)}{S_w} \frac{\partial f_w}{\partial S_w} \right) \frac{\partial S_w}{\partial x} &= 0,\end{aligned}\tag{5.20}$$

which in matrix-vector notation becomes

$$\begin{pmatrix} S_w \\ c_p \end{pmatrix}_t + A \begin{pmatrix} S_w \\ c_p \end{pmatrix}_x = \begin{pmatrix} 0 \\ 0 \end{pmatrix}\tag{5.21}$$

with

$$A = \frac{u_T}{\phi} \begin{pmatrix} \frac{\partial f_w}{\partial S_w} & \frac{\partial f_w}{\partial c_p} \\ \frac{c_p f_w \frac{\partial \alpha}{\partial S_w} + c_p(\alpha-1) \frac{\partial f_w}{\partial S_w}}{S_w} & \frac{\alpha f_w + c_p(\alpha-1) \frac{\partial f_w}{\partial c_p}}{S_w} \end{pmatrix}.$$

The discriminant is now given by

$$D = \left(\frac{u_T}{\phi} \right)^2 \left[\left(\frac{\partial f_w}{\partial S_w} - \frac{\alpha f_w + c_p(\alpha-1) \frac{\partial f_w}{\partial c_p}}{S_w} \right)^2 + 4 \left(\frac{\partial f_w}{\partial c_p} \frac{c_p f_w \frac{\partial \alpha}{\partial S_w} + c_p(\alpha-1) \frac{\partial f_w}{\partial S_w}}{S_w} \right) \right].$$

From (5.18) we easily find that

$$\frac{d\alpha}{dS_w} = -\frac{S^*}{(S_w - S^*)^2} = -\frac{\alpha(\alpha-1)}{S_w}.$$

Substituting this, the discriminant can be written as

$$D = \left(\frac{1}{S_w} \right)^2 \left(\frac{u_T}{\phi} \right)^2 \left[\left(\frac{\partial f_w}{\partial S_w} S_w - \alpha f_w - c_p(\alpha-1) \frac{\partial f_w}{\partial c_p} \right)^2 + 4 \frac{\partial f_w}{\partial c_p} \left(-\alpha(\alpha-1) c_p f_w + c_p(\alpha-1) \frac{\partial f_w}{\partial S_w} S_w \right) \right].$$

Expanding the quadratic term allows us to cancel terms and we end up with

$$D = \left(\frac{1}{S_w} \right)^2 \left(\frac{u_T}{\phi} \right)^2 \left[\left(\frac{\partial f_w}{\partial S_w} S_w - \alpha f_w + c_p(\alpha-1) \frac{\partial f_w}{\partial c_p} \right)^2 \right] > 0,$$

which means that our system is indeed strictly hyperbolic. Note that we have not used any assumptions in the value of S^* to deduce this and therefore the model is well-posed for all values of S^* .

Numerical solutions for this saturation dependent model with threshold saturation $S^* = 0.14$ are shown in Figure 5.13. The solution profile is clearly dependent on the chosen amount of time steps. As $\Delta t \downarrow 0$, the peak in the polymer concentration disappears and the front moves further up the reservoir in the same time. The solution converges to a solution that shows a monotone polymer concentration. This means that for a finite Δt , the peak in polymer concentration at the front is numerical and not physical, i.e. not part of the model. One of the models proposed by Hilden et al. (2016) does not show this numerical accumulation of polymer concentration at the front. In all examples considered in their paper, results of this model show a monotone polymer concentration. However, no guarantee is given that this will always be the case, so the solution profile could still be dependent on the grid and time step size.

Since polymer accumulation near the water front is not necessarily unphysical [9, 4], it is also unsure if the solutions showing a monotone polymer concentration are physically correct. In order to make statements about this, the models should be validated against experimental data first.

What we can conclude is that the conventional approach of choosing a constant velocity enhancement factor results in an ill-posed system and an unstable solution. To overcome this, some form of saturation dependency could be used to model the velocity enhancement. What form this should be depends on which physical results one wishes to mimic. Also, one must make sure that the numerical model is such that the results correspond to the physical model. Numerical effects like the overshoot in polymer concentration shown in Figure 5.13 have to be avoided.

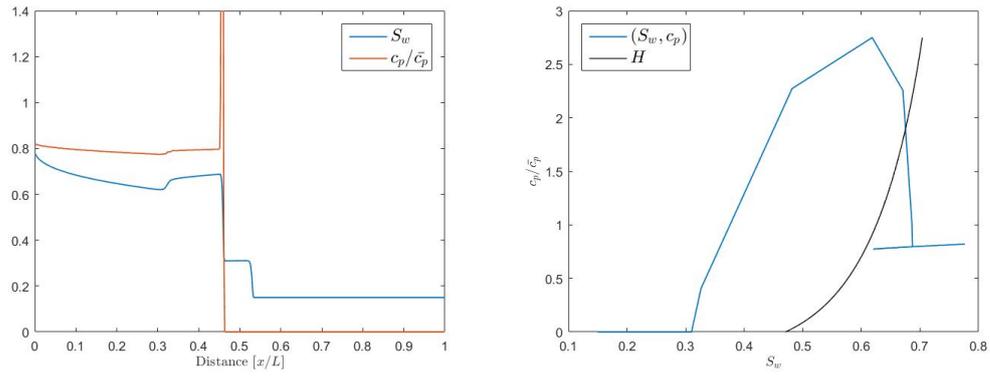
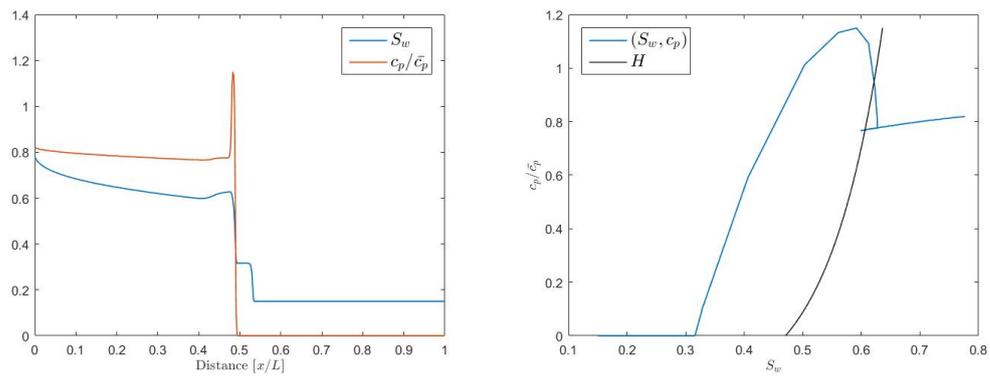
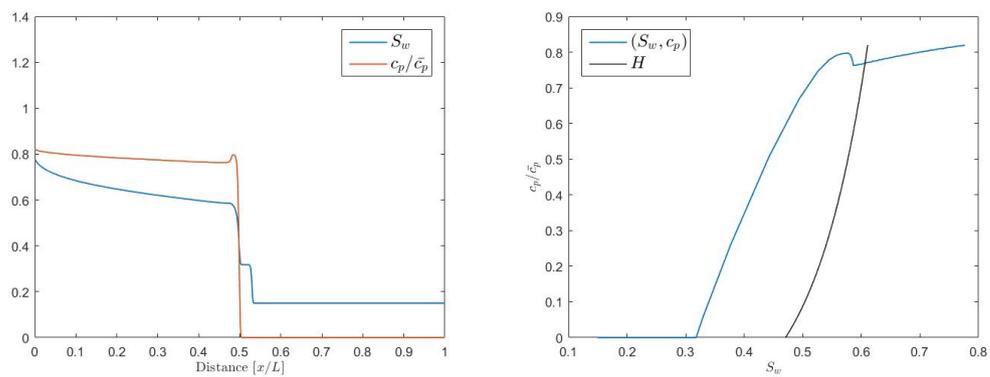
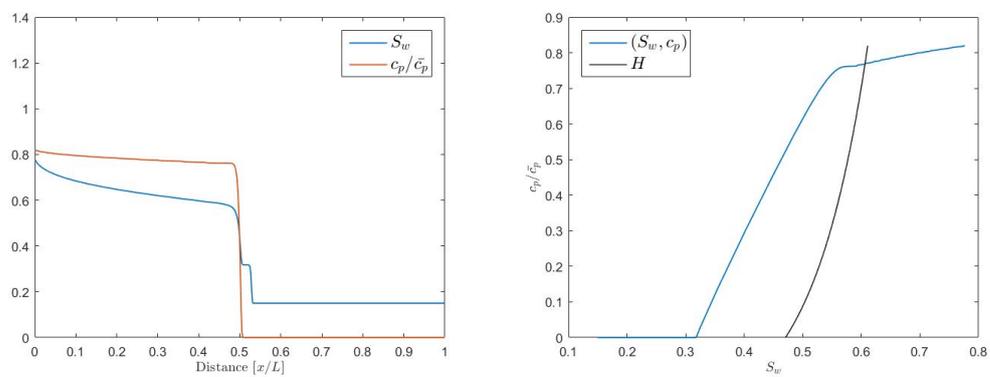
(a) Solution using 500 time steps in x -space (left) and phase-space (right).(b) Solution using 1000 time steps in x -space (left) and phase-space (right).(c) Solution using 2000 time steps in x -space (left) and phase-space (right).(d) Solution using 4000 time steps in x -space (left) and phase-space (right).

Figure 5.13: Numerical solutions of polymer flooding in secondary mode with a saturation dependent velocity enhancement factor using 500 grid cells and different amount of time steps.

6

Transport of surfactant

In addition to polymer flooding, another widely used EOR technique is surfactant flooding. Surfactant is soluble in both oil and water and classified as a partitioning active tracer. Injection of surfactants is often combined with the injection of polymer (SP flooding), polymer and alkaline (ASP flooding) or foam (SF flooding). In this section we solely treat SP flooding.

As mentioned in the introduction of Chapter 5, polymer flooding accelerates the oil production but does not change the residual oil saturation and therefore does not change the total amount of movable oil present in the reservoir. Due to the interfacial tension (IFT) between oil and water, a large amount of oil remains trapped in the porous medium. By decreasing the IFT, part of this trapped oil can be released, resulting in a lower residual oil saturation and an increase in the oil recovery. The leading EOR technique for achieving a low IFT is the use of surfactants [12]. Besides the impact on the residual oil saturation, lowering the IFT also alters the two-phase flow properties, causing the relative permeability curves to shift upwards to more straight lines, see Figure 6.1 for an example.

Already at a very low surfactant concentration the IFT reaches its lowest value, see Figure 6.2. Since typical surfactant concentrations are around 0.5%, the transition from the set of relative permeabilities of an oil-water system to the set of relative permeabilities for an oil-water-surfactant system will be rather sudden. Therefore an instantaneous switch in relative permeabilities is an obvious choice to model this transition. However, it will be shown in this section that this instantaneous switch will lead to numerical difficulties, causing oscillations in the solution profile over time. Some form of interpolation could be used to diminish these oscillations.

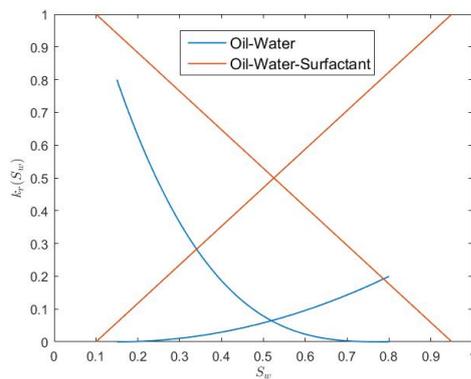


Figure 6.1: Typical set of relative permeability functions for an oil-water and oil-water-surfactant system.

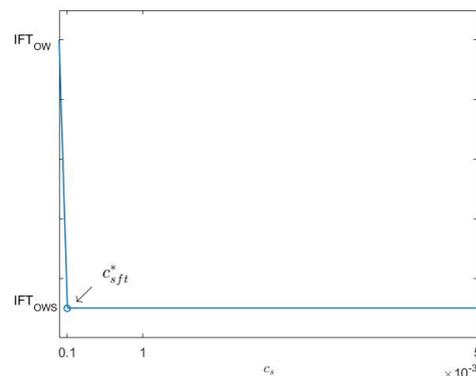


Figure 6.2: Interfacial tension (IFT) as a function of the surfactant concentration c_s , showing that the lowest IFT value is reached for a very low surfactant concentration.

As for polymer flooding, it is very important for the prediction of SP flooding that the surfactant and polymer fronts are solved accurately. In Chapter 5 it was shown that of all numerical methods considered,

the explicit high resolution method was most accurate and therefore most suited to sharply resolve the fronts in the polymer flooding solution. This numerical method is therefore applied to all SP flooding problems.

Also as in chapter 5, adsorption of surfactant (and polymer) onto the rock is omitted from the transport model.

6.1. Model equations

The transport equation of surfactant is different from that of polymer, since surfactant partitions between the oil and water phase. This partitioning is described by $c_{s,o} = K c_{s,w}$, where $c_{s,o}$ is the surfactant concentration in oil, $c_{s,w}$ the surfactant concentration in water and K the partitioning coefficient. By defining $c_s = c_{s,w}$, $K_w = 1$ and $K_o = K$, the conservation equation for surfactant can be written as

$$\begin{aligned} \frac{\partial}{\partial t} \left(\phi \sum_{\alpha \in \{o,w\}} S_\alpha K_\alpha c_s \right) &= -\nabla \cdot \left(\sum_{\alpha \in \{o,w\}} \mathbf{u}_\alpha K_\alpha c_s \right) \\ \phi \frac{\partial}{\partial t} [(S_w + K S_o) c_s] &= -\nabla \cdot [(\mathbf{u}_w + K \mathbf{u}_o) c_s]. \end{aligned}$$

After solving this equation for c_s , the surfactant concentration in oil simply follows from multiplication with the partitioning coefficient K .

For surfactant flooding, the oil and water velocity depend on the surfactant concentration and we obtain the following coupled system of non-linear conservation equations for oil, water and surfactant

$$\phi \frac{\partial}{\partial t} (S_\alpha) + \nabla \cdot (\mathbf{u}_\alpha (S_\alpha, c_s)) = 0, \quad \alpha \in \{o, w\} \quad (6.1)$$

$$\phi \frac{\partial}{\partial t} [(S_w + K S_o) c_s] + \nabla \cdot [(\mathbf{u}_w (S_w, c_s) + K \mathbf{u}_o (S_o, c_s)) c_s] = 0. \quad (6.2)$$

As mentioned in the introduction of this chapter, surfactant injection is often combined with polymer injection, a chemical EOR technique called *SP flooding*. In case of SP flooding, the oil and water velocity depend on both the polymer and the surfactant concentration. The coupled system of non-linear conservation equations for oil, water, polymer and surfactant is obtained by combining equations (6.1), (6.2) and (5.2):

$$\phi \frac{\partial}{\partial t} (S_\alpha) + \nabla \cdot (\mathbf{u}_\alpha (S_\alpha, c_p, c_s)) = 0, \quad \alpha \in \{o, w\}$$

$$\phi \frac{\partial}{\partial t} (S_w c_p) + \nabla \cdot (c_p \mathbf{u}_w (S_w, c_p, c_s)) = 0, \quad (6.3)$$

$$\phi \frac{\partial}{\partial t} [(S_w + K S_o) c_s] + \nabla \cdot [(\mathbf{u}_w (S_w, c_p, c_s) + K \mathbf{u}_o (S_o, c_p, c_s)) c_s] = 0.$$

If $c_p \equiv 0$ we simply obtain the surfactant flooding model and if $c_s \equiv 0$ the polymer flooding model is obtained. In fractional flow formulation the SP flooding equations become

$$\nabla \cdot \mathbf{u}_T = 0$$

$$\phi \frac{\partial}{\partial t} (S_w) + \mathbf{u}_T \nabla \cdot (f_w (S_w, c_p, c_s)) = 0 \quad (6.4)$$

$$\phi \frac{\partial}{\partial t} (S_w c_p) + \mathbf{u}_T \nabla \cdot (c_p f_w (S_w, c_p, c_s)) = 0 \quad (6.5)$$

$$\phi \frac{\partial}{\partial t} [(S_w + K S_o) c_s] + \mathbf{u}_T \nabla \cdot ((f_w (S_w, c_p, c_s) + K f_o (S_o, c_p, c_s)) c_s) = 0, \quad (6.6)$$

where $f_w (S_w, c_p, c_s)$ is the fractional flow function in the presence of polymer and surfactant. Since $\sum_{\alpha \in \{o,w\}} S_\alpha = 1$ and $\sum_{\alpha \in \{o,w\}} f_\alpha = 1$, equation (6.6) can be simplified to

$$\phi \frac{\partial}{\partial t} [(K + (1 - K) S_w) c_s] + \mathbf{u}_T \nabla \cdot ((K + (1 - K) f_w (S_w, c_p, c_s)) c_s) = 0.$$

The introduction of surfactant causes the relative permeability curves to change very suddenly. This change is modelled as an instantaneous transition when the average surfactant concentration exceeds $c_{sft}^* = 10^{-4}$, the

concentration value at which the lowest IFT value (IFT_{OWS}) is reached (see Figure 6.2). The average surfactant concentration is defined as

$$c_{sft} = c_s(S_w + KS_o) = c_s [S_w + K(1 - S_w)] = c_s [K + (1 - K)S_w]. \quad (6.7)$$

Where $c_{sft}(x, t) \leq c_{sft}^*$ the set of relative permeabilities without surfactant, denoted by $\{k_{rw}, k_{ro}\}$, is applied and where $c_{sft}(x, t) > c_{sft}^*$ the set of relative permeabilities with surfactant, denoted by $\{k_{rw}^{sft}, k_{ro}^{sft}\}$, is applied.

In terms of the fractional flow function $f_w(S_w, c_p, c_s)$ this instantaneous transition can be written as

$$f_w(S_w, c_p, c_s) = \begin{cases} f_w(S_w, c_p) & \text{if } c_{sft}(x, t) \leq c_{sft}^* \\ f_w^{sft}(S_w, c_p) & \text{if } c_{sft}(x, t) > c_{sft}^* \end{cases}$$

where f_w and f_w^{sft} are the fractional flow functions for the oil-water-polymer and the oil-water-polymer-surfactant systems given by

$$f_w(S_w, c_p) = \frac{k_{rw}(S_w)}{k_{rw}(S_w) + \frac{\mu_w(c_p)}{\mu_o} k_{ro}(S_w)}, \quad f_w^{sft}(S_w, c_p) = \frac{k_{rw}^{sft}(S_w)}{k_{rw}^{sft}(S_w) + \frac{\mu_w(c_p)}{\mu_o} k_{ro}^{sft}(S_w)}.$$

with $\mu_w(c_p)$ given by equation (5.6). For the relative permeabilities shown in Figure 6.1 and a constant polymer concentration $\bar{c}_p > 0$ the corresponding fractional flow functions are shown in Figure 6.3.

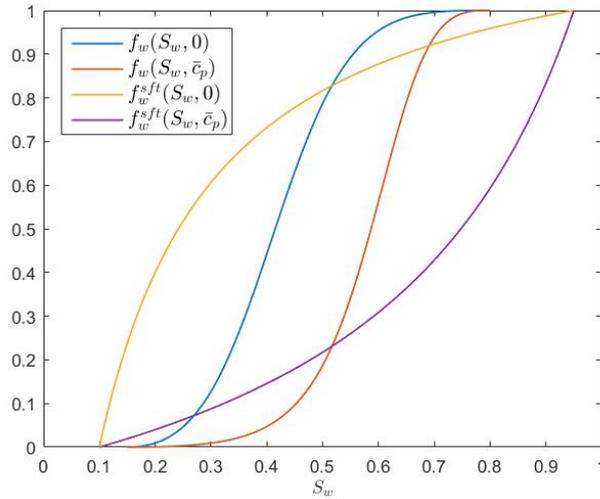


Figure 6.3: Fractional flow functions corresponding to the relative permeability curves shown in Figure 6.1 for an oil-water system ($f_w(S_w, 0)$), an oil-water-polymer system ($f_w(S_w, \bar{c}_p)$), an oil-water-surfactant system ($f_w^{sft}(S_w, 0)$) and an oil-water-polymer-surfactant system ($f_w^{sft}(S_w, \bar{c}_p)$).

6.2. Analytical solution

In this section the analytical solution of some very simple one dimensional SP flooding cases will be derived. In 1D equations (6.4), (6.5) and (6.6) turn into

$$\begin{aligned} \phi \frac{\partial}{\partial t}(S_w) + u_T \frac{\partial}{\partial x}(f_w(S_w, c_p, c_s)) &= 0 \\ \phi \frac{\partial}{\partial t}(S_w c_p) + u_T \frac{\partial}{\partial x}(f_w(S_w, c_p, c_s) c_p) &= 0 \\ \phi \frac{\partial}{\partial t}([K + (1 - K)S_w] c_s) + u_T \frac{\partial}{\partial x}([K + (1 - K)f_w(S_w, c_p, c_s)] c_s) &= 0. \end{aligned} \quad (6.8)$$

Analytical solutions for S_w , c_p and c_s on $x \in \mathbb{R}$ are derived for SP flooding in both secondary and tertiary mode for $K = 0$. For $K > 0$ only the solution of a simplified tertiary flooding case is considered.

If $K = 0$, i.e. if surfactant is only present in the water phase, the last equation of (6.8) simplifies to

$$\phi \frac{\partial}{\partial t} (S_w c_s) + u_T \frac{\partial}{\partial x} (f_w(S_w, c_p, c_s) c_s) = 0 \quad (6.9)$$

which is equal to the polymer transport equation with $c_p = c_s$. For SP flooding in secondary mode similar initial data is imposed on S_w as was done for the polymer flooding. However, since we inject water with surfactant, which lowers the residual oil saturation, the initial data is now given by

$$S_w(x, 0) = \begin{cases} S_w^{inj} = 1 - S_{or}^{sft} = 0.95 & x < 0 \\ S_w^0 & x \geq 0. \end{cases} \quad (6.10)$$

On c_p and c_s the following similar Riemann initial data is imposed:

$$c_p(x, 0) = \begin{cases} \bar{c}_p & x < 0 \\ 0 & x \geq 0 \end{cases} \quad c_s(x, 0) = \begin{cases} \bar{c}_s & x < 0 \\ 0 & x \geq 0, \end{cases} \quad (6.11)$$

with \bar{c}_p and $\bar{c}_s \gg c_{sft}^*$ constant. Again we assume $u_T > 0$, such that the solution for $x > 0$ represents a reservoir on $x > 0$ where from $t = 0$ water with a constant polymer and surfactant concentration (\bar{c}_p, \bar{c}_s) is injected at $x = 0$. The initial polymer and surfactant concentration profile travel through the reservoir with the same speed. The characteristic velocities of both concentrations are equal to

$$\left(\frac{dX}{dt} \right)_{c_p} = \left(\frac{dX}{dt} \right)_{c_s} = \frac{u_T f_w(S_w, c_p, c_s)}{\phi S_w} = \frac{u_T f_w^{sft}(S_w, \bar{c}_p)}{\phi S_w}.$$

Since the polymer and surfactant concentration fronts align, two saturation shocks arise in the SP flooding solution, same as in the polymer flooding solution. One shock where water with polymer and surfactant contacts water without both chemicals and one shock where the water saturation jumps to its initial value.

The difference with polymer flooding is that the flux function for water with polymer and surfactant, $f_w^{sft}(S_w, \bar{c}_p)$, is a convex function (see Figure 6.3). This means that the saturation S^{*1} value left from the first shock is simply the injection saturation $S_w^{inj} = 1 - S_{or}^{sft}$ and the velocity of the shock, i.e. the velocity of the polymer-surfactant front, is given by the characteristic velocity of polymer and surfactant at this saturation:

$$\sigma = \frac{u_T f_w^{sft}(S_w^{inj}, \bar{c}_p)}{\phi S_w^{inj}} = \frac{u_T}{\phi (1 - S_{or}^{sft})}. \quad (6.12)$$

The saturation right after the shock S^{*2} can be found from the jump conditions across the shock:

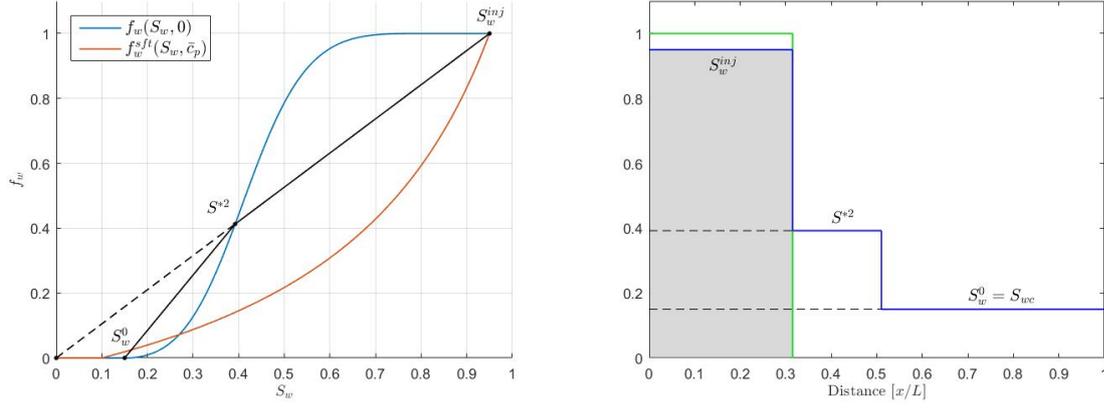
$$\frac{f_w^{sft}(S_w^{inj}, \bar{c}_p)}{S_w^{inj}} = \frac{f_w^{sft}(S_w^{inj}, \bar{c}_p) - f_w(S^{*2}, 0)}{S_w^{inj} - S^{*2}}$$

$$\frac{1}{1 - S_{or}^{sft}} = \frac{1 - f_w(S^{*2}, 0)}{1 - S_{or}^{sft} - S^{*2}}.$$

From S^{*2} the water saturation jumps to the initial water saturation $S_w^0 = S_{wc}$. The resulting analytic solution together with the schematic illustration on how to find the shock saturations is shown in Figure 6.4. Note that the normalized polymer concentration $c_p(x, t)/\bar{c}_p$ and the normalized surfactant concentration $c_s(x, t)/\bar{c}_s$ are the same in this case and therefore represented as one solution. For a more detailed derivation of this analytical solution, see Appendix A.

For SP flooding in tertiary mode, the injection of polymer and surfactant starts after some period T of waterflooding. For simplicity it is assumed that the reservoir is fully water saturated after this period, i.e. $S_w(x, T) = S_w^0 = 1 - S_{or}$. The only difference with the analytical solution of SP flooding in secondary mode is that the water saturation jumps from S^{*2} to $1 - S_{or}$ instead of S_{wc} , see Figure 6.5.

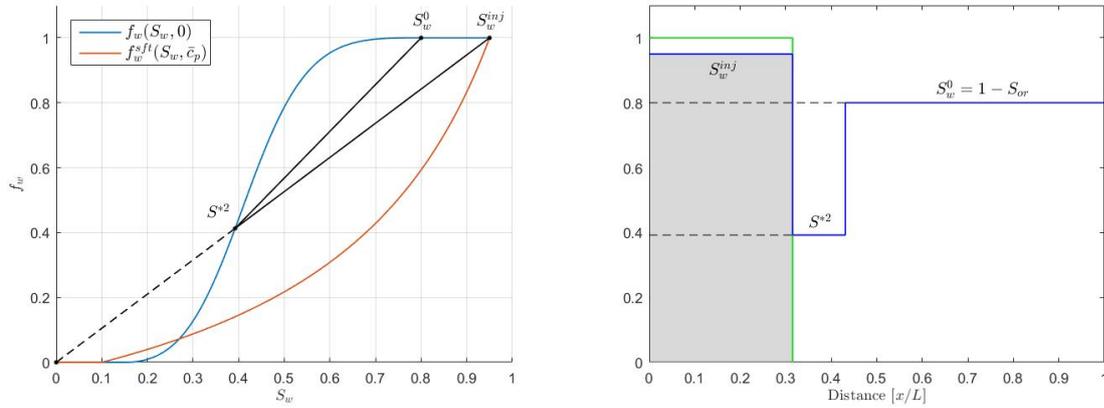
A third case is considered where polymer injection starts at $t = 0$ and surfactant injection after some time T . For simplicity it is again assumed that the reservoir is fully water saturated after this period $S_w(x, T) = 1 - S_{or}$ and the extra assumption is made that the polymer concentration in the reservoir is uniform, i.e.



(a) Schematic illustration of the construction of the SP flooding solution in secondary mode for $K = 0$.

(b) Solution of $S_w(x, t)$, $c_p(x, t)$, $c_s(x, t)$ at a certain time $t > 0$. The water saturation $S_w(x, t)$ is shown in blue and the normalized polymer/surfactant concentration is shown in green.

Figure 6.4: Analytical solution of SP flooding in secondary mode with $K = 0$ for certain time $t > 0$.



(a) Schematic illustration of the construction of the SP flooding solution in tertiary mode for $K = 0$.

(b) Solution of $S_w(x, t)$, $c_p(x, t)$, $c_s(x, t)$ at a certain time $t > 0$. The water saturation $S_w(x, t)$ is shown in blue and the normalized polymer/surfactant concentration is shown in green.

Figure 6.5: Analytical solution of SP flooding in tertiary mode with $K = 0$ for certain time $t > T$.

$c_p(x, T) = \bar{c}_p$. This changes the jump conditions across the first shock. The saturation S^{*2} right after the shock can now be found by solving

$$\frac{f_w^{Sft}(S_w^{inj}, \bar{c}_p)}{S_w^{inj}} = \frac{f_w^{Sft}(S_w^{inj}, \bar{c}_p) - f_w(S^{*2}, \bar{c}_p)}{S_w^{inj} - S^{*2}}.$$

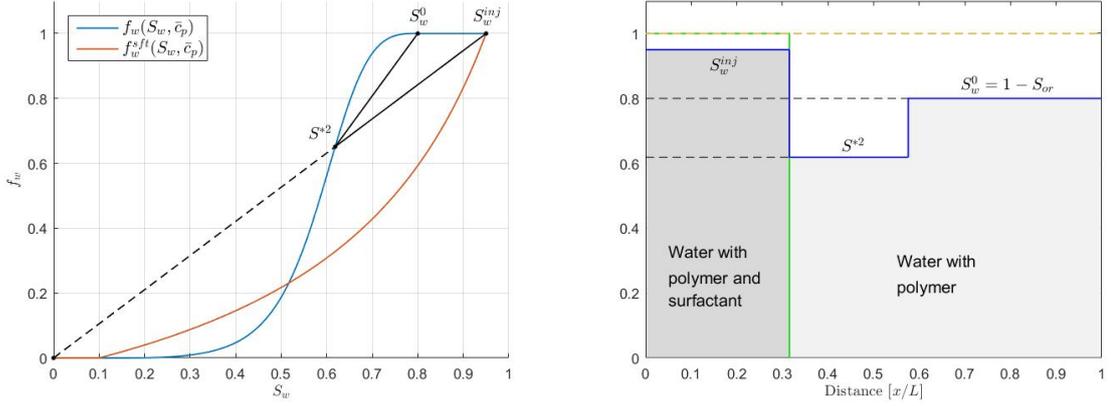
The resulting analytic solution is shown in Figure 6.6.

For $K > 0$ the polymer and surfactant no longer travel with the same speed. The characteristic velocity of the polymer is given by

$$v_p = \left(\frac{dX}{dt} \right)_{c_p} = \frac{u_T}{\phi} \frac{f_w(S_w, c_p, c_s)}{S_w}$$

while the characteristic velocity of the surfactant is given by

$$v_s = \left(\frac{dX}{dt} \right)_{c_s} = \frac{u_T}{\phi} \frac{K + (1-K)f_w(S_w, c_p, c_s)}{K + (1-K)S_w}.$$



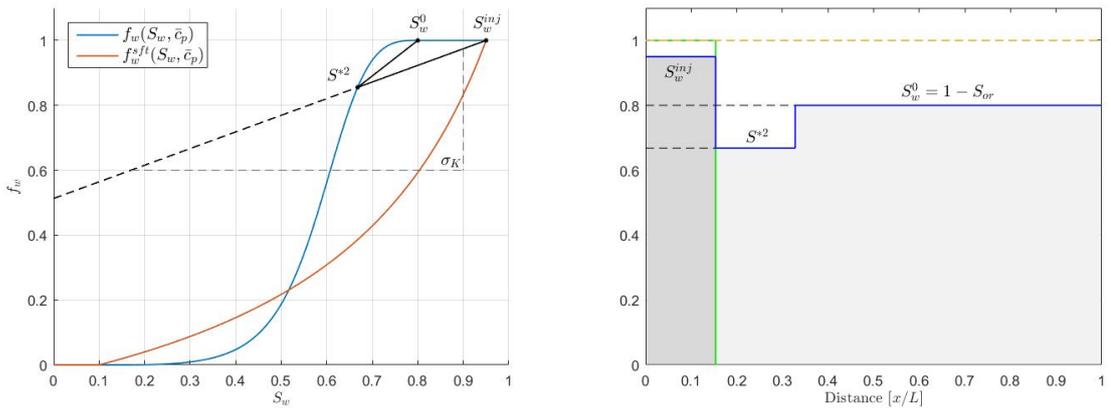
(a) Schematic illustration of the construction of the solution of SP flooding case 3 for $K = 0$. (b) Solution of $S_w(x, t)$, $c_p(x, t)$, $c_s(x, t)$ for certain time $t > 0$. The water saturation $S_w(x, t)$ is shown in blue, the normalized polymer concentration $c_p(x, t)/\bar{c}_p$ is shown in yellow and the normalized surfactant concentration $c_s(x, t)/\bar{c}_s$ is shown in green.

Figure 6.6: Analytical solution of SP flooding case 3 with $K = 0$ at a certain time $t > T$.

This means that if the initial data (6.11) are imposed on c_p, c_s , the polymer and surfactant concentration fronts no longer align for $t > 0$. This complicates the construction of the analytical solution of SP flooding in secondary and tertiary mode. Therefore we only consider the third polymer flooding case, where surfactant injection starts after a certain time T and S_w, c_p are assumed uniform over the reservoir at that time. Again two saturation shocks form. The first shock coincides with the discontinuity in $c_s(x, t)$ and travels with the shock velocity

$$\sigma_K = \frac{u_T}{\phi} \frac{K + (1-K)f_w(S_w^{inj}, c_p, c_s)}{K + (1-K)S_w} = \frac{u_T}{\phi} \frac{1}{K + (1-K)(1 - S_{or}^{sft})} = \frac{u_T}{\phi(1 - S_{or}^{sft} + K S_{or}^{sft})}.$$

Comparing this to the shock velocity for $K = 0$, see equation (6.12), we see that $\sigma_K < \sigma$ for all $K > 0$. Partitioning thus delays the transport of surfactant. The saturation value right from the first shock can be obtained in a similar way as was done for $K = 0$. The resulting analytic solution for $K = 20$ together with the schematic illustration on how to find the shock saturations is shown in Figure 6.7.



(a) Schematic illustration of the construction of the solution of SP flooding case 3 for $K = 20$. (b) Solution of $S_w(x, t)$, $c_p(x, t)$, $c_s(x, t)$ at a certain time $t > 0$. The water saturation $S_w(x, t)$ is shown in blue, the normalized polymer concentration $c_p(x, t)/\bar{c}_p$ is shown in yellow and the normalized surfactant concentration $c_s(x, t)/\bar{c}_s$ is shown in green.

Figure 6.7: Analytical solution of SP flooding case 3 with $K = 20$ for certain time $t > T$.

6.3. Numerical solution

For more complicated problems than the ones considered in the previous section, the SP flooding must be solved numerically. As in section 5.3, the coupled system of equations (6.3) is solved using a sequential approach. This approach is simply an extended version of the approach used for polymer flooding. A single time step from t^n to t^{n+1} in the SP flooding sequential approach consists of the following steps

1) Solve the flow equations: compute S_w^{n+1}, S_o^{n+1} using c_p^n, c_s^n via the fully implicit nonlinear discretization scheme

$$\begin{cases} S_{o,i}^{n+1} = S_{o,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_o \left(S_{o,i}^{n+1}, c_{p,i}^n, c_{s,i}^n \right) - u_o \left(S_{o,i-1}^{n+1}, c_{p,i-1}^n, c_{s,i-1}^n \right) \right] \\ S_{w,i}^{n+1} = S_{w,i}^n - \frac{\Delta t}{\phi \Delta x} \left[u_w \left(S_{w,i}^{n+1}, c_{p,i}^n, c_{s,i}^n \right) - u_w \left(S_{w,i-1}^{n+1}, c_{p,i-1}^n, c_{s,i-1}^n \right) \right] \end{cases} \quad (6.13)$$

2) Solve the polymer and surfactant transport equation: compute c_p^{n+1} and c_s^{n+1} using S_w^{n+1} and S_o^{n+1}

$$\begin{aligned} (S_w c_p)_i^{n+1} &= (S_w c_p)_i^n - \frac{\Delta t}{\phi \Delta x} \left[F_{i+1/2} \left(S_w^{n+1}, c_p^n, c_s^n \right) - F_{i-1/2} \left(S_w^{n+1}, c_p^n, c_s^n \right) \right] \\ [(S_w + K S_o) c_s]_i^{n+1} &= [(S_w + K S_o) c_s]_i^n - \frac{\Delta t}{\phi \Delta x} \left[G_{i+1/2} \left(S_w^{n+1}, S_o^{n+1}, c_p^n, c_s^n \right) - G_{i-1/2} \left(S_w^{n+1}, S_o^{n+1}, c_p^n, c_s^n \right) \right]. \end{aligned}$$

Explicit high-resolution fluxes are chosen for both $F_{i\pm 1/2}$ and $G_{i\pm 1/2}$:

$$\begin{aligned} F_{i+1/2} &= F_{i+1/2}^n = u_{w,i}^{n+1} c_{p,i}^n + \Phi(\theta_{i+1/2}^n) \frac{1}{2} u_{w,i}^{n+1} \left(1 - u_{w,i}^{n+1} \frac{\Delta t}{\Delta x} \right) \left[c_{p,i+1}^n - c_{p,i}^n \right] \\ G_{i+1/2} &= G_{i+1/2}^n = u_{s,i}^{n+1} c_{s,i}^n + \Phi(\theta_{i+1/2}^n) \frac{1}{2} u_{s,i}^{n+1} \left(1 - u_{s,i}^{n+1} \frac{\Delta t}{\Delta x} \right) \left[c_{s,i+1}^n - c_{s,i}^n \right] \end{aligned}$$

where

$$u_{w,i}^{n+1} = u_w \left(S_{w,i}^{n+1}, c_{p,i}^n, c_{s,i}^n \right), \quad u_{s,i}^{n+1} = u_w \left(S_{w,i}^{n+1}, c_{p,i}^n, c_{s,i}^n \right) + K u_o \left(S_{o,i}^{n+1}, c_{p,i}^n, c_{s,i}^n \right).$$

Numerical solutions for the SP flooding situations from Figures 6.4, 6.5, 6.6 and 6.7 are shown in Figure 6.8. Since the focus in this chapter is on the transport of surfactant, the normalized polymer concentration is not shown in the results. For SP flooding in secondary and tertiary mode the normalized polymer concentration $c_p(x, t)/\bar{c}_p$ is equal to the normalized surfactant concentration. For SP flooding case 3, the polymer concentration is uniform over the reservoir, so the normalized polymer concentration is equal to one.

All saturation solutions in Figure 6.8 show a dip below the analytical solution directly after the surfactant front. This dip changes as the solution travels through the reservoir. To further investigate this behaviour we look at the solutions over time at the end of the reservoir which is located at $x = L$, see Figure 6.9. In all cases, small oscillations in time in both the oil and water saturation are visible just before the surfactant front. This behaviour is nonphysical. Since the relative permeabilities and therefore the fractional flow functions switch discontinuously around the surfactant front, the hypothesis is that this discontinuous transition causes the nonphysical behaviour in the solutions. In the next section this hypothesis is tested and analysed one of the cases from Figure 6.9.

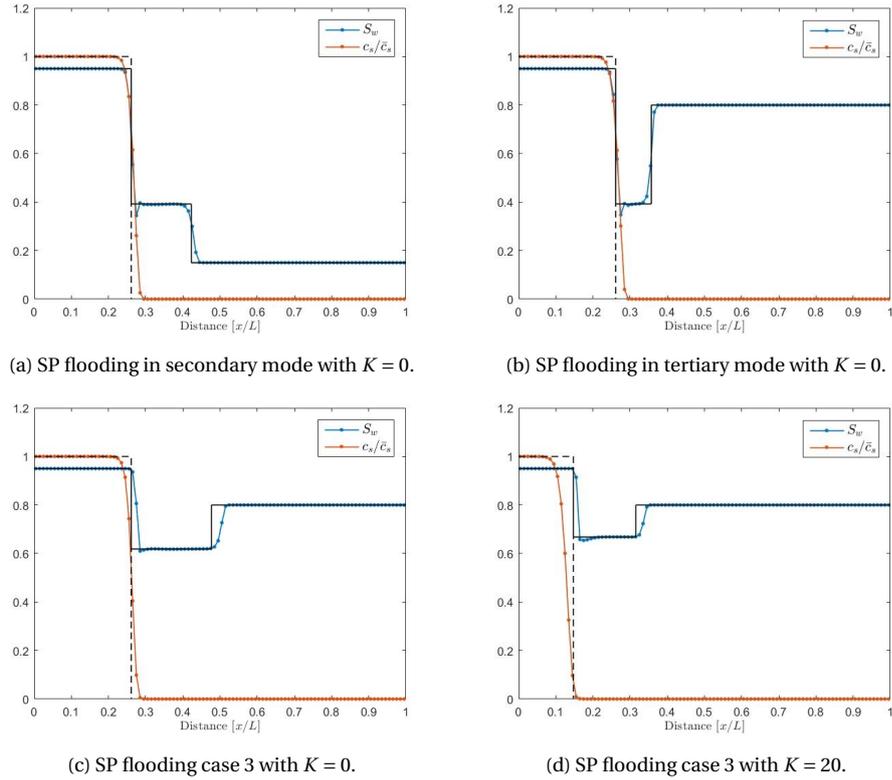


Figure 6.8: Numerical solutions of $S_w(x, t)$ and $c_s(x, t)/\bar{c}_s$ at a certain time $t > 0$ for different SP flooding cases. The analytical solution of $S_w(x, t)$ is given by the solid black line and the analytical solution of $c_s(x, t)/\bar{c}_s$ is given by the dashed black line.

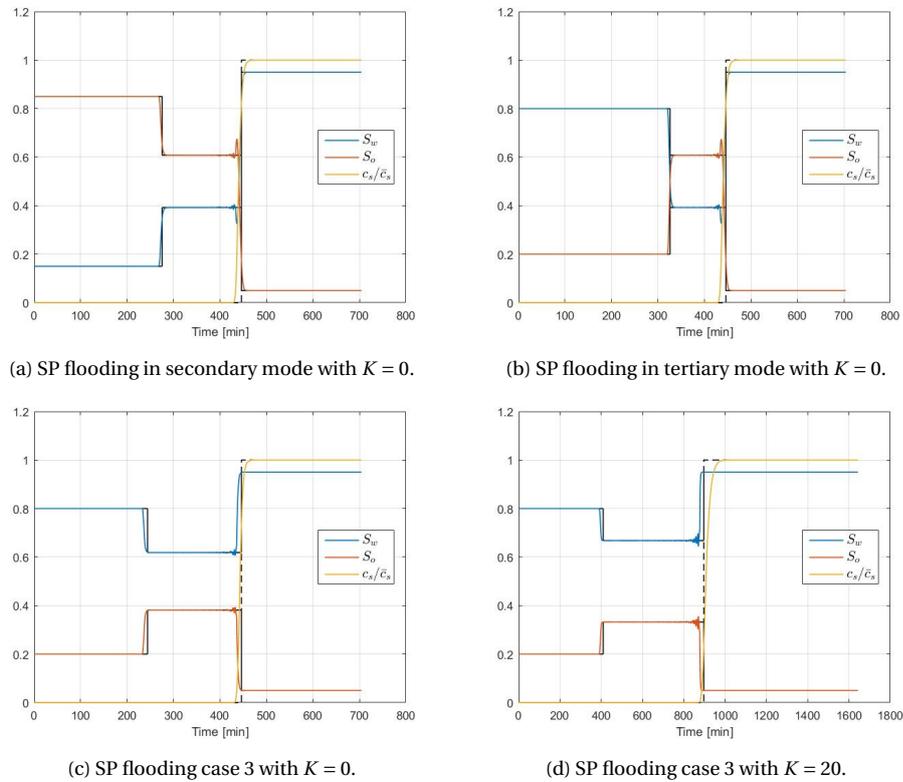


Figure 6.9: Numerical solutions of $S_w(x, t)$, $S_o(x, t)$ and $c_s(x, t)/\bar{c}_s$ over time at the end of the reservoir for different SP flooding cases showing small oscillations in the oil and water saturations over time.

6.4. Analysis oscillations

Consider the third SP flooding case with $K = 0$. To analyse the small oscillations shown in Figure 6.9c, we look at the behaviour of both S_w and the flux function $f_w(S_w, c_s, c_p)$ at a fixed position over time. We vary the location of this position, the grid size and the time step size. Based on the results from this analysis and the previous section, the following observations can be made regarding the oscillations:

- The oscillations occur in time, not in space.
- The oscillations do not disappear for $\Delta t \downarrow 0$.
- The period T of the oscillations depends on the grid size, i.e. $T = \Delta x/v_s$ where v_s is the velocity of the surfactant.
- The amplitude of the oscillations increases slightly in the x -direction.

The third observation follows from the results shown in Figure 6.10, where the grid size is varied while the location and time step size are kept fixed.

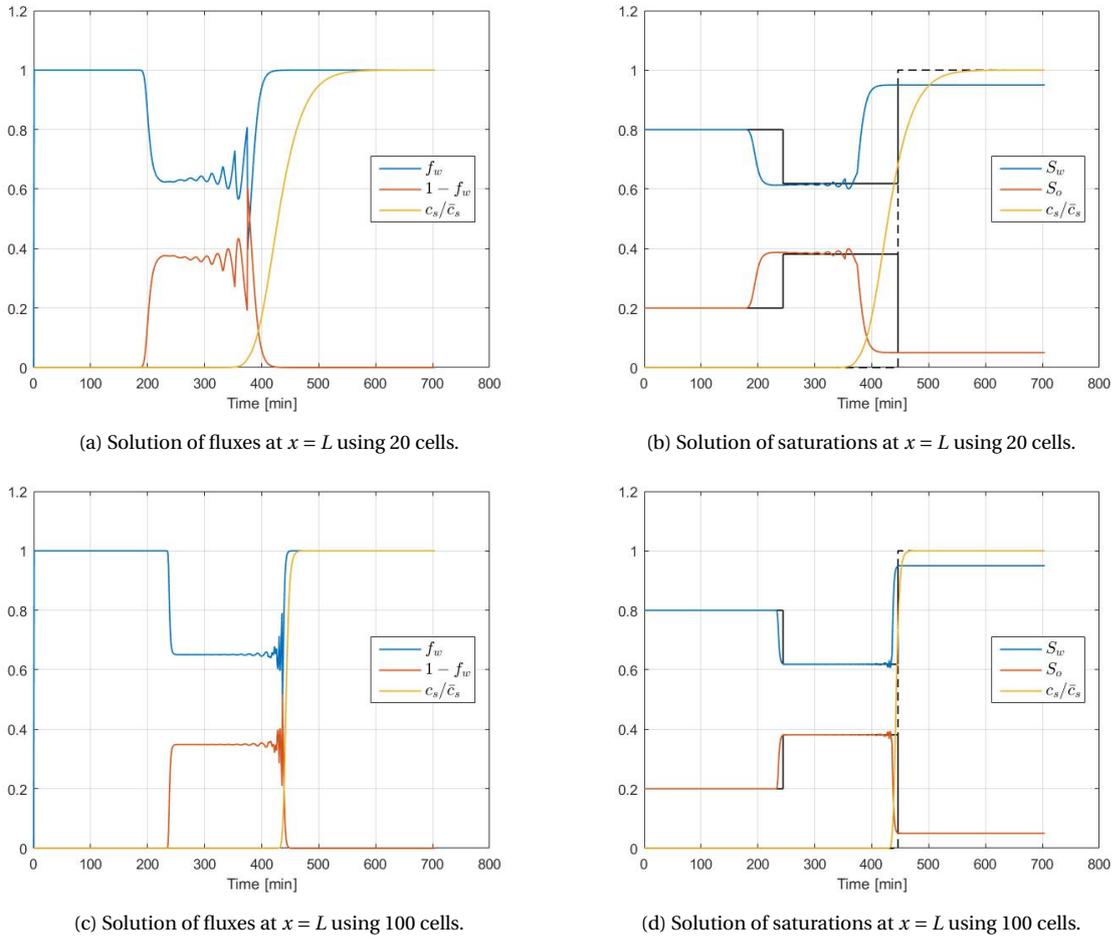


Figure 6.10: Numerical solutions of S_w , $f_w(S_w, c_s, c_p)$ and $c_s(x, t)/\bar{c}_s$ over time at the end of the reservoir for SP flooding case 3.

In Figure 6.10a and 6.10b, which show the solutions over time on the coarse grid, the oscillations are much more visible than in Figures 6.10c and 6.10d, which show the solutions over time on the fine grid. The numerical solution on the coarse grid is therefore selected for closer analysis in order to determine what the cause of the oscillations is.

At a certain time t^n the water saturation is approximated by the discrete saturation profile $S_{w,i}^n$ and the surfactant concentration is approximated by the discrete concentration profile $c_{s,i}^n$, for $i = 1 \dots N$. Since $K = 0$, the average surfactant concentration given in equation (6.7) is approximated by $c_{sft,i}^n = c_{s,i}^n S_{w,i}^n$. In the cells

where $c_{sft,i}^n \leq c_{sft}^*$ the fractional flow function without surfactant is applied and in the cells where $c_{sft,i}^n > c_{sft}^*$ the fractional flow function with surfactant is applied. Denote the first cell where $c_{sft,i}^n \leq c_{sft}^*$ by $I + 1$, then the discrete fractional flow function (i.e. flux function) at time t^n is given by

$$f_{w,i}^n = \begin{cases} f_w(S_{w,i}^n, \bar{c}_p) & \text{for } i \leq I \\ f_w^{sft}(S_{w,i}^n, \bar{c}_p) & \text{for } i > I, \end{cases}$$

with $f_w(S_w, \bar{c}_p)$ and $f_w^{sft}(S_w, \bar{c}_p)$ as in Figure 6.11. The fractional flow function thus switches discontinuously from cell I to cell $I + 1$ as demonstrated in Figure 6.12, where C_i^n is defined as

$$C_i^n = \begin{cases} 1 & \text{where } c_{sft,i}^n \leq c_{sft}^* \\ 0 & \text{where } c_{sft,i}^n > c_{sft}^*. \end{cases}$$

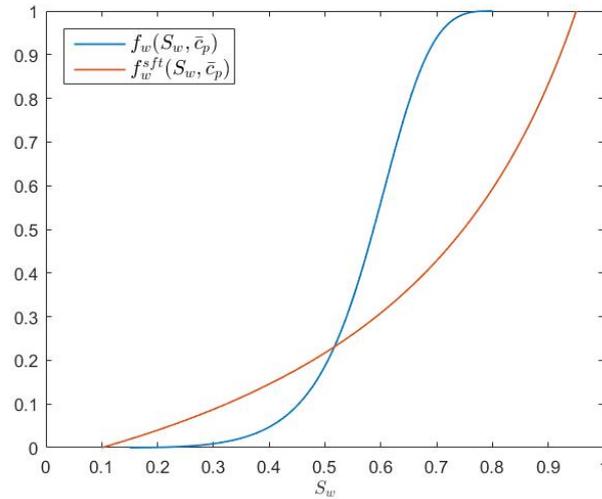


Figure 6.11: Fractional flow functions for an oil-water-polymer system ($f_w(S_w, \bar{c}_p)$) and an oil-water-polymer-surfactant system ($f_w^{sft}(S_w, \bar{c}_p)$).

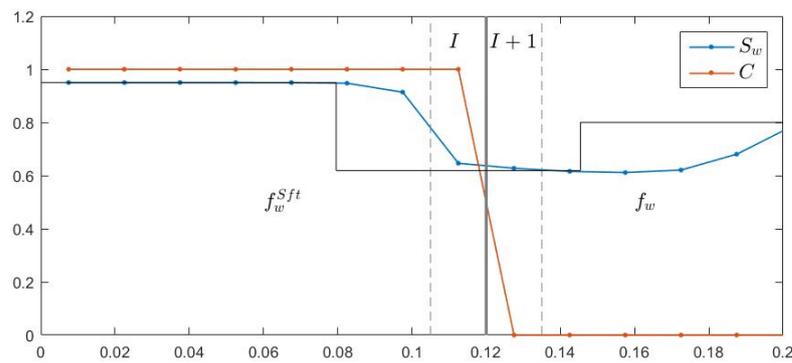


Figure 6.12: Solution profile at time t^n with a discontinuous switch in fractional flow function from cell I to cell $I + 1$.

Now time steps are taken small, i.e. $\Delta t v_s \ll \Delta x$, such that a number of steps is required before the discontinuity shifts to the next grid cell, i.e. before $C_{I+1} = 1$. According to the sequential solution scheme, the saturation at time t^{n+1} follows from the fully implicit scheme (6.13). However, the absence of the fractional flow functions and the implicitness of this scheme makes it difficult to analyse what happens to the solution $S_{w,i}^{n+1}$ based on $f_{w,i}^n$. For this analysis we therefore assume that the water saturation is modelled by the

fractional flow function (6.4), which is discretized using a first order explicit scheme:

$$S_{w,i}^{n+1} = S_{w,i}^n - \frac{u_T}{\phi} \frac{\Delta t}{\Delta x} [f_{w,i}^n - f_{w,i-1}^n].$$

The saturation in grid block $I + 1$ at time t^{n+1} is then given by

$$\begin{aligned} S_{w,I+1}^{n+1} &= S_{w,I+1}^n - \frac{u_T}{\phi} \frac{\Delta t}{\Delta x} [f_{w,I+1}^n - f_{w,I}^n] \\ &= S_{w,I+1}^n - \frac{u}{\phi} \frac{\Delta t}{\Delta x} [f_w(S_{w,I+1}^n, \bar{c}_p) - f_w^{Sf t}(S_{w,I}^n, \bar{c}_p)] \\ &= S_{w,I+1}^n - \frac{u}{\phi} \frac{\Delta t}{\Delta x} F(S_{w,I}^n, S_{w,I+1}^n). \end{aligned} \quad (6.14)$$

Depending on the sign of F , $S_{w,I+1}^{n+1}$ goes either up or down. A contour plot of F for varying $S_{w,I}$ and $S_{w,I+1}$ is shown in Figure 6.13. For the situation of Figure 6.12, $(S_{w,I}^n, S_{w,I+1}^n)$ is located left from the red line, i.e. in the

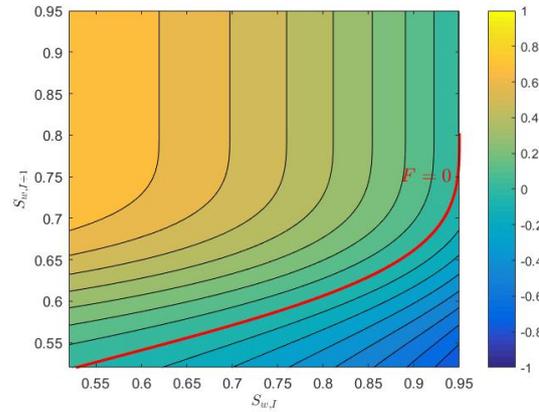


Figure 6.13: Contour plot of $F(S_{w,I}, S_{w,I+1})$.

region where $F > 0$, which means that $S_{w,I+1}^{n+1} < S_{w,I+1}^n$ according to (6.14). The saturation in cell $I + 1$ keeps declining over time until $(S_{w,I}, S_{w,I+1})$ is such that $F > 0$, after which $S_{w,I+1}$ will increase again. This process of decline and increase occurs within the time it takes for the surfactant front to travel through one grid block (i.e. the time it takes for C_{I+1} to go from zero to one), causing an oscillation to arise in $S_{w,I+1}$ over time. This behaviour is illustrated by the solution path of $(S_{w,I}, S_{w,I+1})$ from t^n till $t^{n+\tau}$ shown in Figure 6.14a. At $t^{n+\tau+1}$ the average surfactant concentration in cell $I + 1$ is high enough for C_{I+1} to turn one. This means that after $t^{n+\tau}$ the discontinuity in the flux function moves one grid block and the whole process repeats itself. If we refine the grid then almost the same solution path is followed but in less steps, see Figure 6.14b. This is in accordance with the observation made earlier that the period of the oscillations is proportional to Δx .

Now we have explained the oscillation in saturation in the cell adjacent to the surfactant front. However, long before the surfactant front arrives at the end of the reservoir we already see oscillations appearing in the solution, see Figure 6.10. Cause for this lies in the numerical scheme. For the analysis above we assumed that the flow follows an explicit scheme, while actually the equation for the water saturation is solved implicitly. If an implicit scheme is used, then the solution in the cell adjacent to the front influences the solution in the subsequent cells. For example, if the first order implicit fractional flow discretization is applied, the saturation in cell $I + 2$ at time t^{n+1} is given by

$$\frac{u}{\phi} \frac{\Delta t}{\Delta x} f_w(S_{w,I+2}^{n+1}, \bar{c}_p) + S_{w,I+2}^{n+1} = S_{w,I+2}^n + \frac{u}{\phi} \frac{\Delta t}{\Delta x} f_w(S_{w,I+1}^{n+1}, \bar{c}_p).$$

Fluctuations in $S_{w,I+1}^{n+1}$ thus result in fluctuations in $S_{w,I+2}^{n+1}$.

From the analysis in this section it can be concluded that oscillations in the numerical solution are caused by large differences between fluxes left and right from the surfactant front. These large differences are caused

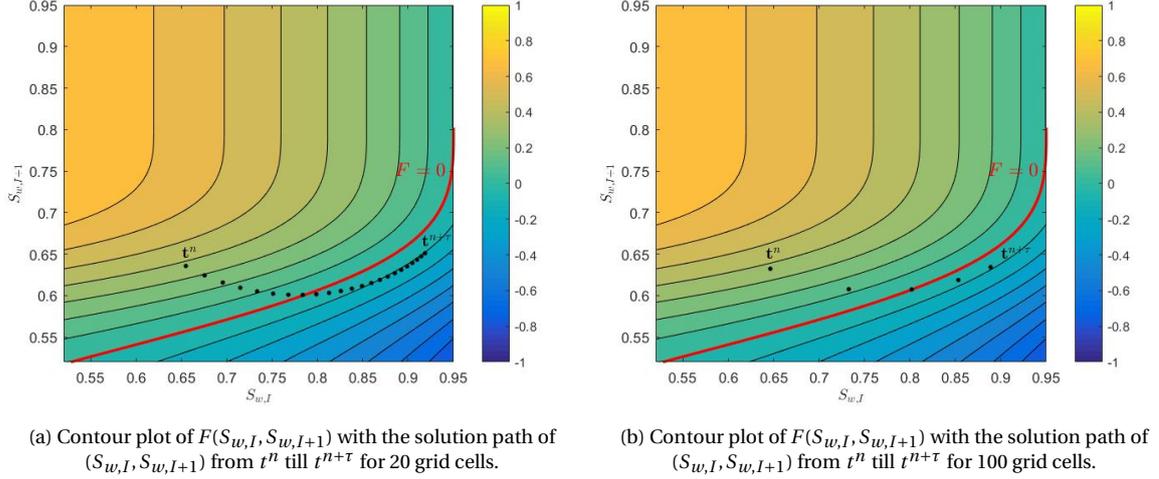


Figure 6.14: Contour plot of F with solution profiles for a coarse grid (left) and a fine grid (right) showing how the oscillations arise in the water saturation over time and that the period of the oscillations depends on Δx .

by the discontinuous transition from the fractional flow curve with surfactant f_w^{sft} to the fractional flow curve without surfactant f_w from one grid block to the next. A more smooth transition could decrease these differences and diminish the oscillations. To do so, an interpolation between the fractional flow curves based on the surfactant concentration c_s is proposed in the next section. However, no guarantee can be given that the interpolation removes the oscillations completely. Also note that we are now looking into a specific SP flooding case with fractional flow functions as in Figure 6.11. In situations where the fractional flow functions are closer to each other, oscillations are less likely to arise and will have a smaller amplitude.

6.4.1. Interpolation

Instead of instantaneously switching between relative permeability curves when $c_{sft}(x, t)$ exceeds $c_{sft}^* = 10^{-4}$, an interpolation function $m(c_{sft})$ is introduced which attains values between 0 and 1 for average surfactant concentrations between 0 and c_{sft}^* . This function is used to interpolate between the two sets of relative permeability functions in the following way:

$$k_{rw}(S_w, c_s) = [1 - m(c_{sft})] \cdot k_{rw}(S_w) + m(c_{sft}) \cdot k_{rw}^{sft}(S_w)$$

$$k_{ro}(S_w, c_s) = [1 - m(c_{sft})] \cdot k_{ro}(S_w) + m(c_{sft}) \cdot k_{ro}^{sft}(S_w).$$

Since $k_{rw}(S_w)$ and $k_{ro}(S_w)$ are not defined for $S_{wc}^{sft} \leq S_w < S_{wc}$ and $1 - S_{or} < S_w \leq 1 - S_{or}^{sft}$, the relative permeability functions are extended as in Figure 6.15 to be able to apply this interpolation.

It is also possible to directly interpolate between the fractional flow functions f_w and f_w^{sft} . However, since our flow simulator solves equations (6.1) and not the fractional flow equations, the fractional flow functions are not directly used in the solution scheme and this approach can't easily be applied. We therefore stick to interpolation between the sets of relative permeabilities.

For the interpolation function $m(c_{sft})$ any function that attains values between 0 and 1 can be selected. Three popular choices for interpolation functions are shown in Figure 6.16. The effect of each of these functions on the resulting fractional flow function for different values of c_{sft} is shown in Figure 6.17. As can be seen in this figure, the linear interpolation results in a slower transition towards f_w than the sine and tangent hyperbolic shaped interpolations.

For the case considered in this section, the linear interpolation function turns out to perform better than the other two interpolation functions. Numerical results obtained by applying the linear interpolation are shown in Figure 6.18. Although the oscillations in the fluxes have not completely disappeared, they are strongly reduced compared to the results for the instantaneous switch shown in Figure 6.8.

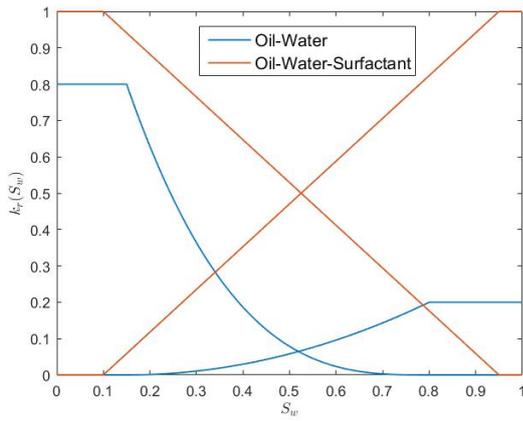


Figure 6.15: Extended relative permeability functions for the oil-water and oil-water-surfactant system to be used for interpolation.

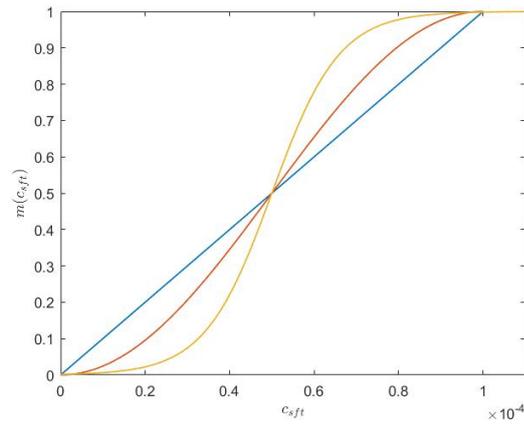
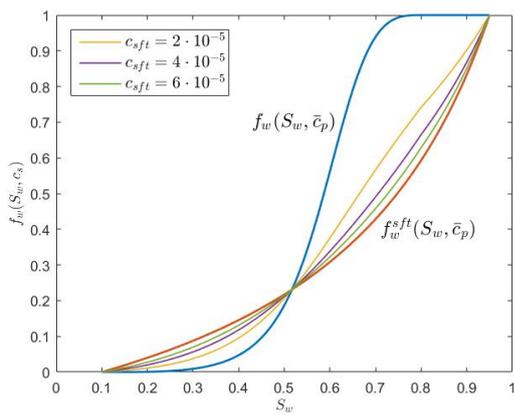
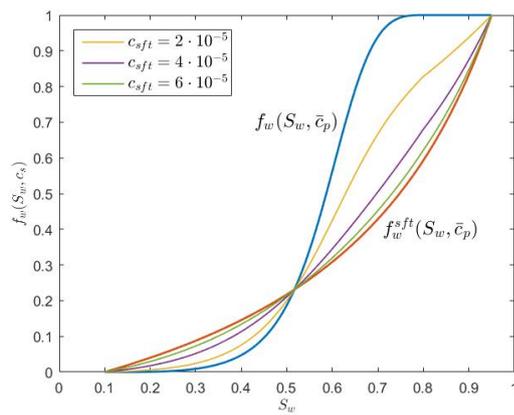


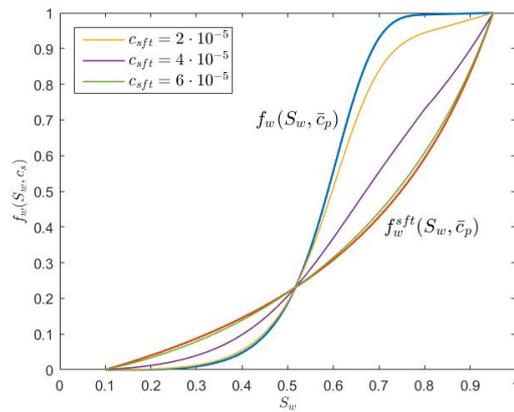
Figure 6.16: Three popular interpolation functions $m(c_{sft})$: a linear function (blue), a sine shaped function (red) and a tangent hyperbolic shaped function (yellow).



(a) Linear $m(c_{sft})$.



(b) Sine shaped $m(c_{sft})$.



(c) Tangent hyperbolic shaped $m(c_{sft})$.

Figure 6.17: Effect of three different interpolation functions $m(c_{sft})$ on the fractional flow function.

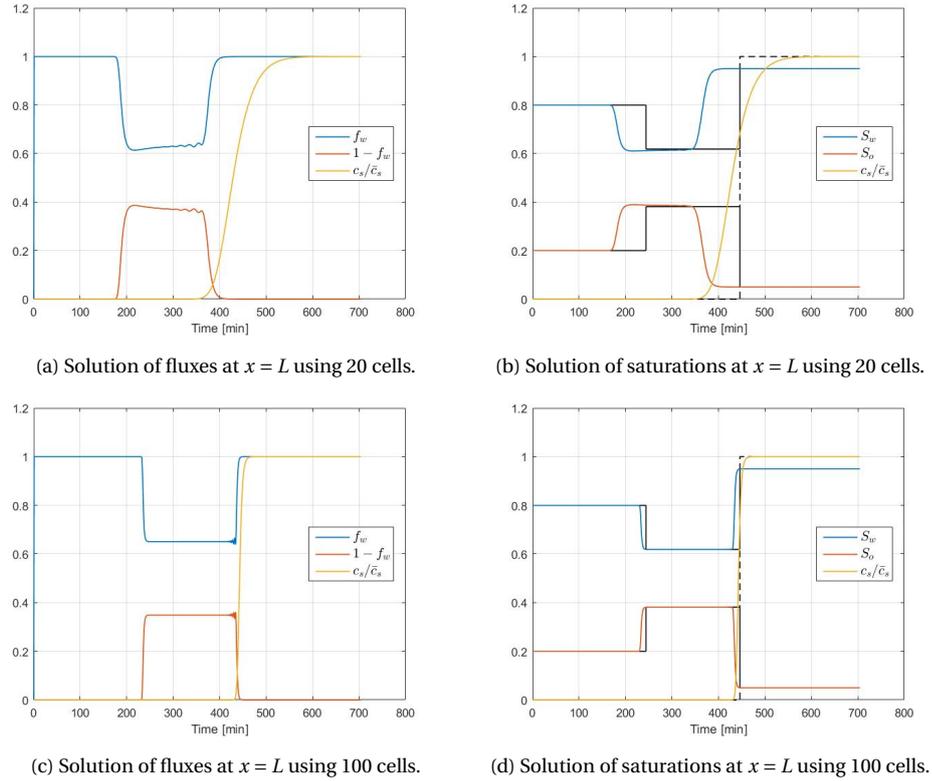


Figure 6.18: Numerical solutions of S_w , $f_w(S_w, c_s, c_p)$ and $c_s(x, t)/\bar{c}_s$ obtained through linear interpolation between the relative permeabilities.

6.4.2. Illustration on typical SP flooding sequence

The findings from the previous sections are now illustrated on a more realistic example, a typical SP flooding sequence. This sequence starts with a preflush, where only water is injected, followed by a slug containing polymer and surfactant. The slug is followed by an aqueous polymer solution and the sequence ends with chase water. Injection schemes are commonly given in terms of pore volumes injected. One pore volume, ϕV , is defined as the total accessible volume in a reservoir with V the total volume of the reservoir. In terms of pore volumes injected, the injection scheme of the SP flooding sequence is given in Table 6.1.

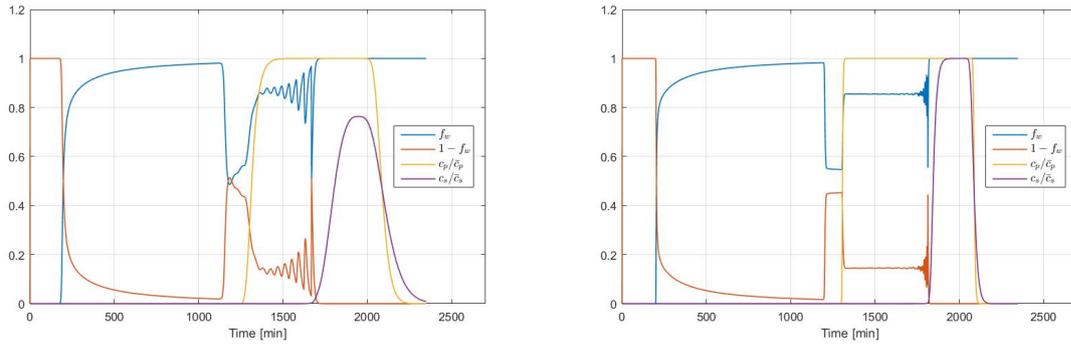
Injection scheme of a typical SP flooding sequence		
PV_{WF}	2.0	PV of water injected
PV_{SP}	0.5	PV of SP injected
PV_{pPF}	1.0	PV of polymer post SP injected
PV_{pWF}	1.0	PV of water post flood

Table 6.1: Injection scheme of a typical SP flooding sequence in terms of injected pore volumes (PV).

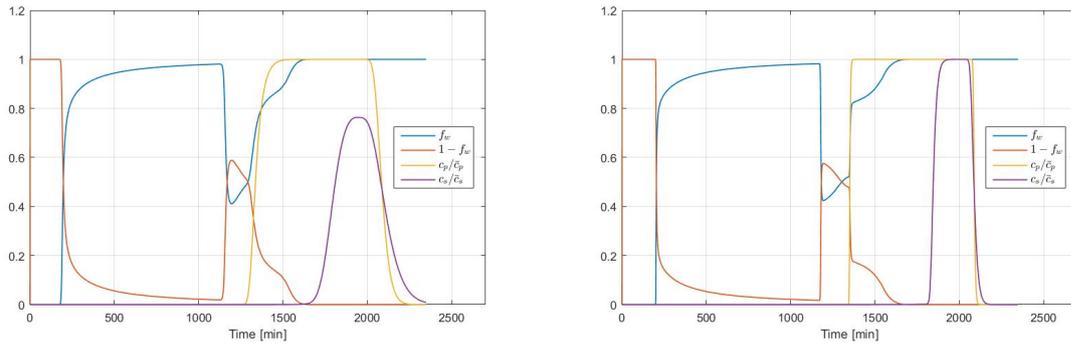
The surfactant partitions over the oil and water phase with $K = 20$ and we the relative permeabilities switch discontinuously at $c_{sft}(x, t) = c_{sft}^*$. Again we assume a finite one dimensional reservoir at $x \in [0, L]$. The numerical solutions at the end of the reservoir of the oil and water fluxes, normalized polymer and normalized surfactant concentration in water are shown in Figure 6.19a for two different grid sizes. Similar oscillations are observed as in the simplified SP flooding case. Applying a linear interpolation instead of a discontinuous transition in relative permeabilities resolves these oscillations but also changes the structure of the solution very much, see Figure 6.19b. The linear interpolation results in a faster emergence of the effect of surfactant, causing an earlier increase in the oil production. On the other hand, applying a sine shaped interpolation results in a solution more like the solution of Figure 6.19a, but does not completely resolve the oscillations in the solution, see Figure 6.19c.

6.4.3. Conclusion

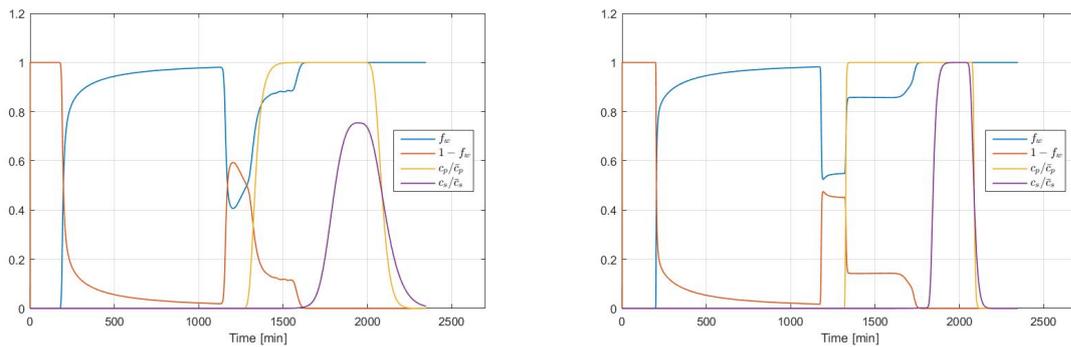
From the analysis in this section we can conclude that the oscillations in the SP flooding solutions are caused by the discontinuous switch in relative permeabilities at the surfactant front. These oscillations can be resolved or at least diminished by applying some form of interpolation. The interpolation function must be carefully selected, as it can have a large impact on the resulting solution profile. To be able to accurately predict the effect of SP flooding on the oil production, we must make sure that the resulting solution profile matches the physical behaviour we wish to describe. Selection of a suited interpolation function for the problem at hand can at this moment only be done based on the outcome of the simulation. This approach is not very practical when dealing with real field cases, which take a considerable amount of time to simulate. It would be an improvement if the selection of the interpolation function could be done prior to simulation.



(a) Solution obtained through instantaneous switch in relative permeabilities using 20 cells (left) and 100 cells (right).



(b) Solution obtained through a linear interpolation between relative permeabilities using 20 cells (left) and 100 cells (right).



(c) Solution obtained through a sine shaped interpolation between relative permeabilities using 20 cells (left) and 100 cells (right).

Figure 6.19: Numerical solutions at $x = L$ of the oil and water fluxes ($f_w(S_w, c_s, c_p)$ and $1 - f_w(S_w, c_s, c_p)$), the normalized polymer concentration ($c_p(x, t)/\bar{c}_p$) and the normalized surfactant concentration in water ($c_s(x, t)/\bar{c}_s$) for different transitions in relative permeabilities.

7

Conclusions and recommendations

In this chapter the conclusions of this research are presented followed by some recommendations regarding future work.

7.1. Conclusions

The overall goal of this research was to determine the best way to discretise and couple flow and transport equations in terms of robustness, stability, efficiency and accuracy. Furthermore, special attention was paid to the modelling of several physical phenomena associated with polymer and surfactant flooding. In this section the main conclusions are presented.

In Chapter 2 it was shown that the fully implicit method that is commonly used in commercial reservoir simulators to solve the system of coupled nonlinear flow equations produces a very stable scheme at the cost of accuracy and efficiency. The method is computationally very expensive and introduces a lot of numerical smearing in the saturation. By decoupling flow and transport the transport equation can be solved using a more accurate and efficient numerical scheme.

To solve the transport equation, different first order and high resolution schemes were implemented and assessed in section 3.2 and 5.3. The explicit high-resolution method (called the semi-implicit method in case of a passive tracer) performed best in terms of accuracy and efficiency. This method strongly reduces the smearing introduced by first order methods and accurately resolves the sharp fronts in both the tracer concentration and the water saturation. The computational expense of this method is very low compared to the computational expense of the flow solver. However, the method is only conditionally stable, so care has to be taken when selecting the time steps.

In Chapter 4 it was shown that the monotonicity of the numerical transport scheme is strongly dependent on the underlying discrete flow solution. When the same fluxes are used to solve both flow and transport, the resulting numerical scheme for the transport equation will always be monotone. However, when the fluxes differ, which is the case when we apply the explicit high-resolution method to the transport equation, then the numerical scheme can become non-monotonous. To overcome this problem, a partially implicit method was proposed. This method follows the accurate and efficient high-resolution scheme but switches to the monotone first order implicit scheme where necessary to ensure monotonicity of the scheme.

Besides the monotonicity, it was shown that the accuracy of the scheme depends on the underlying flow solution as well. The closer the tracer front is to the oil-water front, where the flow solution is the least accurate, the less the accurate the transport solution.

Overall it can be concluded from Chapters 3, 4 and 5 that it is very useful in terms of accuracy and efficiency to apply a high-resolution method to the transport equations even though the underlying flow is solved using a fully implicit first order method.

As mention in the introduction of this thesis, special attention should be paid to how the influences of active tracers on the flow are modelled. Bad choices can lead to instabilities or cause deviations in the model outcome. In section 5.4 it was shown that the conventional approach for modelling the hydrodynamic acceleration of polymer via a constant velocity enhancement factor results in an ill-posed system and an unstable solution. To obtain a well-posed system, some form of saturation dependency could be used to model the

velocity enhancement instead. However, for the saturation dependent models presented in literature it was neither confirmed nor denied whether their solutions correspond to the actual physical results that we wish to mimic. Furthermore, the numerical models are not guaranteed to correspond to the physical models as nonphysical numerical effects can occur in the solution. It therefore remains an open question which physical and numerical model is best suited for modelling the hydrodynamic acceleration.

Furthermore, we saw in Chapter 6 that the introduction of surfactant rapidly lowers the interfacial tension between oil and water, causing the set of relative permeability curves for the system to change rather sudden in the presence of surfactant. When this transition in relative permeabilities is modelled by a discontinuous switch, oscillations arise in the SP flooding solution. These oscillations can be resolved or at least diminished by applying some form of interpolation between the relative permeabilities. The interpolation function must be carefully selected, as it can have a large impact on the resulting solution profile. To be able to accurately predict the effect of SP flooding on the oil production, the resulting solution profile must match the physical behaviour we wish to describe.

From these two cases we can conclude that when modelling influences of the tracer(s) on the flow, the model must be such that the resulting system remains well-posed and the introduction of discontinuities in the model must be avoided when possible.

7.2. Further remarks and recommendations

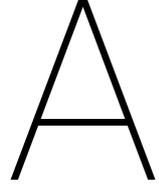
In this research the transport equation was assumed to only contain an advective term. Dispersion, diffusion and adsorption terms that are typically present in transport equations, were not included. Since dispersion/diffusion tends to smoothen the sharp interfaces that arise in the advective transport, numerical problems disappear often when (a small amount of) dispersion/diffusion is introduced into the model. For example the oscillations that arise in the SP flooding solution due to the sudden change in relative permeabilities tend to disappear upon introduction of a dispersion term. However, the modelling of this diffusion/dispersion term brings numerical difficulties of its own.

Generally, diffusion terms are not solved using an explicit method, since the necessary condition for stability of this method places a very severe constraint on the time step. Instead an implicit method is preferable, such as the standard Crank–Nicolson method [13]. This method for the diffusion term can most easily be combined with the high-resolution method for the advection term by using a fractional step method. Whether this is indeed the best approach to model the diffusion/dispersion term in the transport equation of tracers in enhanced oil recovery has to be assessed in further research.

Furthermore, only a very simplified version of the transport of a partitioning tracer was considered in this research, where the tracer concentration in one phase is simply a multiple of the tracer concentration in the other phase. In reality, the process is much more involved and further research is necessary to determine the best way to model and discretize the transport equation of a partitioning tracer.

Also, as mentioned in the conclusion, the question which physical and numerical model is best suited for modelling the hydrodynamic acceleration of polymer remains open after this research. All we can say is that the conventional approach of a constant velocity enhancement factor should not be used, since it results in an ill-posed system. To answer this question, the well-posed models available in literature have to be further evaluated. This evaluation should be two-fold. Firstly, it must be made sure that the numerical solutions correspond to the physical models and that no nonphysical effects are present in the numerical solution. Secondly, the physical model results have to be validated against experimental results to confirm that the correct physical behaviour is being modelled. If none of the existing models turns out to be qualified to model the hydrodynamic acceleration, a new model should be developed.

Finally, in this thesis it was shown that the oscillations arising in the SP flooding solutions due to a sudden change in relative permeabilities can be diminished by applying some form of interpolation. However, this gives no guarantee that the oscillations disappear completely. Besides that, the selection of a suited interpolation function is now based on the simulation results and can differ per situation. Since simulations of real field cases are quite time consuming, the evaluation of different interpolation functions after simulation is not a desirable approach. Ideally, a model or method should be developed that guarantees an oscillation free solution based on information prior to simulation. Further research is necessary to assess whether such a method can be developed.



Deriving analytical solutions

A.1. Passive tracer

In this appendix we will derive an analytical solution to the passive tracer transport problem presented in section 3.1. Although the passive tracer transport equation can be solved separately from the flow equation, the two are now coupled to obtain a system of two conservation laws. For simplicity we write $S = S_w$ and we obtain

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S) \right) = 0 \quad (\text{A.1})$$

$$\frac{\partial (Sc)}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S) c \right) = 0 \quad (\text{A.2})$$

This system can be written as $q_t + f(q)_x = 0$ with

$$q = \begin{bmatrix} S \\ Sc \end{bmatrix} \quad f(q) = \begin{bmatrix} \frac{u_T}{\phi} f_w(q_1) \\ \frac{u_T}{\phi} f_w(q_1) \frac{q_2}{q_1} \end{bmatrix}.$$

Across any shock the Rankine-Hugoniot jump condition must be satisfied:

$$\sigma = \frac{f(q_L) - f(q_R)}{q_L - q_R}, \quad (\text{A.3})$$

where q_L is the state directly left from the shock and q_R the state directly right from the shock. In order to find the postshock value q^* , the following two equations must thus be satisfied simultaneously:

$$\sigma = \frac{u_T}{\phi} \frac{f_w(q_1^*) - f_w(q_{1,R})}{q_1^* - q_{1,R}} \quad (\text{A.4})$$

$$\sigma = \frac{u_T}{\phi} \frac{f_w(q_1^*) \frac{q_{2,R}^*}{q_{1,R}^*} - f_w(q_{1,R}) \frac{q_{2,R}}{q_{1,R}}}{q_2^* - q_{2,R}}. \quad (\text{A.5})$$

Since the shock solution for $q_1 = S_w$ is already known from the analytical flow solution, only the shock in q_2 is considered. This shock is located at the point where the concentration c jumps from \bar{c} to 0, i.e. $c^* = \bar{c}$ and $c_R = 0$. Substituting this into (A.5) results in shock velocity

$$\sigma = \frac{u_T}{\phi} \frac{f_w(S^*)c^* - f_w(S_R)c_R}{S^*c^* - S_Rc_R} = \frac{u_T}{\phi} \frac{f_w(S^*)\bar{c}}{S^*\bar{c}} = \frac{u_T}{\phi} \frac{f_w(S^*)}{S^*}.$$

On the other hand equation (A.4) can be written as

$$\sigma = \frac{u_T}{\phi} \frac{f_w(S^*) - f_w(S_R)}{S^* - S_R} = \frac{u_T}{\phi} \frac{df_w}{dS}(S^*),$$

assuming that there is no discontinuity in S and therefore that $S^* \approx S_R$. Combining both expressions results in

$$\frac{f_w(S^*)}{S^*} = \frac{df_w}{dS}(S^*).$$

A.2. Polymer

In this appendix we will derive an analytical solution to the polymer transport problem presented in section 5.2. As in Appendix A.1, the transport and flow equation are coupled and we obtain the following system of conservation laws:

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S, c_p) \right) = 0 \quad (\text{A.6})$$

$$\frac{\partial(S c_p)}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S, c_p) c_p \right) = 0, \quad (\text{A.7})$$

where $S = S_w$. This system can be written as $q_t + f(q)_x = 0$ with

$$q = \begin{bmatrix} S \\ S c_p \end{bmatrix} \quad f(q) = \begin{bmatrix} \frac{u_T}{\phi} f_w \left(q_1, \frac{q_2}{q_1} \right) \\ \frac{u_T}{\phi} f_w \left(q_1, \frac{q_2}{q_1} \right) \frac{q_2}{q_1} \end{bmatrix}.$$

The solution of $q_1(x, t)$ ($= S_w(x, t)$) contains two shocks. The first shock is located at the point where the polymer concentration c_p ($= q_2/q_1$) jumps from \bar{c}_p to 0. This shock in q_1 coincides with a discontinuity in q_2 ($= S_w(x, t) c_p(x, t)$), separating the water with polymer from the water without polymer. Across this shock the Rankine-Hugoniot jump condition (A.3) must be satisfied, which leads to the following two equations for the shock velocity σ :

$$\begin{aligned} \sigma &= \frac{u_T}{\phi} \frac{f_w \left(q_{1,L}, \frac{q_{2,L}}{q_{1,L}} \right) - f_w \left(q_{1,R}, \frac{q_{2,R}}{q_{1,R}} \right)}{q_{1,L} - q_{1,R}} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p) - f_w(S_R, 0)}{S_L - S_R} \\ \sigma &= \frac{u_T}{\phi} \frac{f_w \left(q_{1,L}, \frac{q_{2,L}}{q_{1,L}} \right) \frac{q_{2,L}}{q_{1,L}} - f_w \left(q_{1,R}, \frac{q_{2,R}}{q_{1,R}} \right) \frac{q_{2,R}}{q_{1,R}}}{q_{2,L} - q_{2,R}} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p) \bar{c}_p}{S_L \bar{c}_p} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p)}{S_L}. \end{aligned}$$

These velocities must be the same, which results in the following relation between the saturation states left and right from the shock

$$\frac{f_w(S_L, \bar{c}_p) - f_w(S_R, 0)}{S_L - S_R} = \frac{f_w(S_L, \bar{c}_p)}{S_L}$$

Furthermore, since we are dealing with a contact discontinuity between $S_w c_p$ and S_w , the shock velocity must be equal to the characteristic water velocity at saturation value S_L . Left from the shock, the water saturation equation is given by

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S, \bar{c}_p) \right) = 0$$

which, since \bar{c}_p is constant, can be written as

$$\frac{\partial S}{\partial t} + \left(\frac{u_T}{\phi} \frac{df_w(S, \bar{c}_p)}{dS} \right) \frac{\partial S}{\partial x} = 0.$$

The characteristic velocity at saturation S_L is thus equal to

$$\left(\frac{dX}{dt} \right)_{S_L} = \frac{u_T}{\phi} \frac{df_w(S, \bar{c}_p)}{dS} (S_L) = \frac{u_T}{\phi} \frac{df_{wp}}{dS} (S_L).$$

By equating this velocity with the shock velocity σ , the saturation S_L can be found:

$$\begin{aligned} \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p)}{S_L} &= \frac{u_T}{\phi} \frac{f_{wp}(S_L)}{S_L} = \frac{u_T}{\phi} \frac{df_{wp}}{dS} (S_L) \\ \frac{f_{wp}(S_L)}{S_L} &= \frac{df_{wp}}{dS} (S_L). \end{aligned}$$

We can thus obtain the values for $S_w(x, t)$ left and right from the shock via

$$\frac{df_{wp}}{dS} (S_L) = \frac{f_{wp}(S_L)}{S_L} = \frac{f_{wp}(S_L) - f_w(S_R)}{S_L - S_R}.$$

A.3. Surfactant

In this appendix we will derive an analytical solution to the surfactant-polymer transport problem presented in section 6.2. The system of conservation laws of Appendix A.2 is extended with the equation for surfactant transport:

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S, c_p, c_s) \right) = 0 \quad (\text{A.8})$$

$$\frac{\partial(S c_p)}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S, c_p, c_s) c_p \right) = 0 \quad (\text{A.9})$$

$$\phi \frac{\partial(S c_s)}{\partial t} + \frac{\partial}{\partial x} \left(\frac{u_T}{\phi} f_w(S_w, c_p, c_s) c_s \right) = 0. \quad (\text{A.10})$$

This system can be written as $q_t + f(q)_x = 0$ with

$$q = \begin{bmatrix} S \\ S c_p \\ S c_s \end{bmatrix} \quad f(q) = \begin{bmatrix} \frac{u_T}{\phi} f_w \left(q_1, \frac{q_2}{q_1}, \frac{q_3}{q_1} \right) \\ \frac{u_T}{\phi} f_w \left(q_1, \frac{q_2}{q_1}, \frac{q_3}{q_1} \right) \frac{q_2}{q_1} \\ \frac{u_T}{\phi} f_w \left(q_1, \frac{q_2}{q_1}, \frac{q_3}{q_1} \right) \frac{q_3}{q_1} \end{bmatrix}.$$

The solution of $q_1(x, t)$ ($S_w(x, t)$) contains two shocks. The first shock is located at the point where the polymer and surfactant concentration (c_p, c_s) ($= (q_2/q_1, q_3/q_1)$) jump from (\bar{c}_p, \bar{c}_s) to $(0, 0)$. This shock in q_1 coincides with a discontinuity in q_2 and q_3 , separating the water with polymer and surfactant from the water without polymer and surfactant. Across this shock the Rankine-Hugoniot jump condition (A.3) must be satisfied, which leads to the following three equations for the shock velocity σ :

$$\begin{aligned} \sigma &= \frac{u_T}{\phi} \frac{f_w \left(q_{1,L}, \frac{q_{2,L}}{q_{1,L}}, \frac{q_{3,L}}{q_{1,L}} \right) - f_w \left(q_{1,R}, \frac{q_{2,R}}{q_{1,R}}, \frac{q_{3,R}}{q_{1,R}} \right)}{q_{1,L} - q_{1,R}} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p, \bar{c}_s) - f_w(S_R, 0)}{S_L - S_R} \\ \sigma &= \frac{u_T}{\phi} \frac{f_w \left(q_{1,L}, \frac{q_{2,L}}{q_{1,L}}, \frac{q_{3,L}}{q_{1,L}} \right) \frac{q_{2,L}}{q_{1,L}} - f_w \left(q_{1,R}, \frac{q_{2,R}}{q_{1,R}}, \frac{q_{3,R}}{q_{1,R}} \right) \frac{q_{2,R}}{q_{1,R}}}{q_{2,L} - q_{2,R}} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p, \bar{c}_s) \bar{c}_p}{S_L \bar{c}_p} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p, \bar{c}_s)}{S_L} \\ \sigma &= \frac{u_T}{\phi} \frac{f_w \left(q_{1,L}, \frac{q_{2,L}}{q_{1,L}}, \frac{q_{3,L}}{q_{1,L}} \right) \frac{q_{3,L}}{q_{1,L}} - f_w \left(q_{1,R}, \frac{q_{2,R}}{q_{1,R}}, \frac{q_{3,R}}{q_{1,R}} \right) \frac{q_{3,R}}{q_{1,R}}}{q_{3,L} - q_{3,R}} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p, \bar{c}_s) \bar{c}_s}{S_L \bar{c}_s} = \frac{u_T}{\phi} \frac{f_w(S_L, \bar{c}_p, \bar{c}_s)}{S_L} \end{aligned}$$

These velocities must be the same, which results in the following relation between the saturation states left and right from the shock

$$\frac{f_w(S_L, \bar{c}_p, \bar{c}_s) - f_w(S_R, 0)}{S_L - S_R} = \frac{f_w(S_L, \bar{c}_p, \bar{c}_s)}{S_L}.$$

This velocity must also equal the velocity of the discontinuity in q_2 and q_3 . Since the flux function $f_w(S_L, \bar{c}_p, \bar{c}_s)$ is convex, the discontinuity in $S_w c_p$ and $S_w c_s$ is simply a jump from the injection values $(S_w^{inj} \bar{c}_p)$, $(S_w^{inj} \bar{c}_s)$ to the initial state zero. The velocity of this discontinuity is given by the jump condition

$$\sigma = \frac{f_w(S_w^{inj}, \bar{c}_p, \bar{c}_s) \bar{c}_p - 0}{S_w^{inj} \bar{c}_p - 0} = \frac{f_w(S_w^{inj}, \bar{c}_p, \bar{c}_s)}{S_w^{inj}}$$

which simply means that $S_L = S_w^{inj}$. The saturation right from the shock can now easily be found by

$$\frac{f_w(S_w^{inj}, \bar{c}_p, \bar{c}_s) - f_w(S_R, 0)}{S_w^{inj} - S_R} = \frac{f_w(S_w^{inj}, \bar{c}_p, \bar{c}_s)}{S_w^{inj}}.$$

B

Monotonicity analysis in 2D

Consider the two-dimensional problem (3.17) on $(x, y) \in \{x, y \geq 0\}$ with $u_x, u_y > 0$ and where from $t = 0$ water with a constant tracer concentration \bar{c} is injected at $(x, y) = (0, 0)$. Assume that the discrete concentration profile in two dimensions at time t_n is given by the piecewise constant data

$$c(x_i, y_j, t_n) = c_{ij}^n = \begin{cases} \bar{c} & \text{if } (i, j) \in \{i \leq I \wedge j \leq J\} \\ 0 & \text{if } (i, j) \in \{i > I \vee j > J\}. \end{cases} \quad (\text{B.1})$$

The semi-implicit first order method, given by

$$S_{ij}^{n+1} c_{ij}^{n+1} = S_{ij}^n c_{ij}^n - \frac{\Delta t}{\phi \Delta x} \left[c_{ij}^n u_{x,i}^{n+1} - c_{i-1,j}^n u_{x,i-1}^{n+1} \right] - \frac{\Delta t}{\phi \Delta y} \left[c_{ij}^n u_{y,j}^{n+1} - c_{i,j-1}^n u_{y,j-1}^{n+1} \right],$$

is applied to (B.1) to obtain the solution at time t_{n+1} , c_{ij}^{n+1} . This method is monotonicity-preserving if this solution satisfies $c_{ij}^{n+1} \geq \{c_{i+1,j}^{n+1}, c_{i,j+1}^{n+1}\}$ for all i, j .

In cells $i, j \in \{1 \leq i \leq I, 1 \leq j \leq J\}$ the solution becomes

$$\begin{aligned} S_{ij}^{n+1} c_{ij}^{n+1} &= S_{ij}^n \bar{c} - \frac{\Delta t}{\phi \Delta x} \left[u_{x,i}^{n+1} - u_{x,i-1}^{n+1} \right] \bar{c} - \frac{\Delta t}{\phi \Delta y} \left[u_{y,j}^{n+1} - u_{y,j-1}^{n+1} \right] \bar{c} \\ &= S_{ij}^{n+1} \bar{c} \end{aligned}$$

where the last step follows from the fully implicit two-dimensional flow scheme (2.18), which can be simplified to

$$S_{ij}^{n+1} = S_{ij}^n - \frac{\Delta t}{\phi \Delta x} \left[u_{x,i}^{n+1} - u_{x,i-1}^{n+1} \right] - \frac{\Delta t}{\phi \Delta y} \left[u_{y,j}^{n+1} - u_{y,j-1}^{n+1} \right]. \quad (\text{B.2})$$

We thus obtain $c_{ij}^{n+1} = \bar{c} = c_{ij}^n$ for cells (i, j) with $1 \leq i \leq I, 1 \leq j \leq J$.

In cell $(I+1, j)$ with $1 \leq j \leq J$ the solution is given by

$$\begin{aligned} S_{I+1,j}^{n+1} c_{I+1,j}^{n+1} &= S_{I+1,j}^n c_{I+1,j}^n - \frac{\Delta t}{\phi \Delta x} \left[c_{I+1,j}^n u_{x,I+1}^{n+1} - c_{I,j}^n u_{x,I}^{n+1} \right] \bar{c} - \frac{\Delta t}{\phi \Delta y} \left[c_{I+1,j}^n u_{y,j}^{n+1} - c_{I+1,j-1}^n u_{y,j-1}^{n+1} \right] \\ &= \frac{\Delta t}{\phi \Delta x} u_{x,I}^{n+1} \bar{c} \end{aligned}$$

since $c_{I+1,j}^n = c_{I+1,j-1}^n = 0$, and thus

$$c_{I+1,j}^{n+1} = \frac{\Delta t}{\phi \Delta x} \frac{u_{x,I}^{n+1}}{S_{I+1,j}^{n+1}} \bar{c}.$$

Rewriting $\frac{\Delta t}{\phi \Delta x} u_{x,I}^{n+1}$ with the use of (B.2) we obtain

$$\frac{\Delta t}{\phi \Delta x} u_{x,I}^{n+1} = S_{I+1,j}^{n+1} - S_{I+1,j}^n + \frac{\Delta t}{\phi \Delta x} u_{x,I+1}^{n+1} + \frac{\Delta t}{\phi \Delta y} \left(u_{y,j}^{n+1} - u_{y,j-1}^{n+1} \right).$$

Substituting this results in

$$c_{I+1,j}^{n+1} = \left[1 - \frac{S_{I+1,j}^n}{S_{I+1,j}^{n+1}} + \frac{\Delta t}{\phi \Delta x} \frac{u_{x,I+1}^{n+1}}{S_{I+1,j}^{n+1}} + \frac{\Delta t}{\phi \Delta y} \left(\frac{u_{y,j}^{n+1} - u_{y,j-1}^{n+1}}{S_{I+1,j}^{n+1}} \right) \right] \bar{c}.$$

For the scheme to be monotonicity-preserving this solution must satisfy $c_{I+1,j}^{n+1} \leq c_{Ij}^{n+1} = \bar{c}$. So, assuming that $S_{I+1,j}^{n+1} > 0$, it must hold that

$$S_{I+1,j}^n - \frac{\Delta t}{\phi \Delta x} u_{x,I+1}^{n+1} - \frac{\Delta t}{\phi \Delta y} (u_{y,j}^{n+1} - u_{y,j-1}^{n+1}) \geq 0.$$

This means that the scheme becomes non-monotone in cells i, j where

$$S_{ij}^n - \frac{\Delta t}{\phi} \left[\frac{u_{x,i}^{n+1}}{\Delta x} + \frac{u_{y,j}^{n+1} - u_{y,j-1}^{n+1}}{\Delta y} \right] < 0.$$

Applying a similar approach to the solution in cell $(i, J+1)$ with $1 \leq i \leq I$, results in the monotonicity criterion

$$S_{i,J+1}^n - \frac{\Delta t}{\phi \Delta x} (u_{x,i}^{n+1} - u_{x,i-1}^{n+1}) - \frac{\Delta t}{\phi \Delta y} u_{y,J+1}^{n+1} \geq 0,$$

which means that the scheme also becomes non-monotone in cells i, j where

$$S_{ij}^n - \frac{\Delta t}{\phi} \left[\frac{u_{x,i}^{n+1} - u_{x,i-1}^{n+1}}{\Delta x} + \frac{u_{y,j}^{n+1}}{\Delta y} \right] < 0.$$

Bibliography

- [1] C Agca, GA Pope, and K Sepehrnoori. Modelling and analysis of tracer flow in oil reservoirs. *Journal of Petroleum Science and Engineering*, 4(1):3–19, 1990.
- [2] Kai Bao, Knut-Andreas Lie, Olav Møyner, and Ming Liu. Fully implicit simulation of polymer flooding with mrst. In *ECMOR XV-15th European Conference on the Mathematics of Oil Recovery*, 2016.
- [3] GA Bartelds, J Bruining, and J Molenaar. The modeling of velocity enhancement in polymer flooding. *Transport in Porous Media*, 26(1):75–88, 1997.
- [4] Gepke Alies Bartelds. The influence of inaccessible and excluded pore volume on polymer flooding. 1998.
- [5] Martin Blunt and Barry Rubin. Implicit flux limiting schemes for petroleum reservoir simulation. *Journal of Computational Physics*, 102(1):194–210, 1992.
- [6] Christos Chatzichristos, Jan Sagen, Olaf Huseby, Jiri Muller, et al. Advanced numerical modelling for tracer flow. *IAHS Publication(International Association of Hydrological Sciences)*, (262):17–23, 2000.
- [7] Zhangxin Chen, Guanren Huan, and Yuanle Ma. *Computational methods for multiphase flows in porous media*. SIAM, 2006.
- [8] KH Coats. Reservoir simulation. *Petroleum Engineering Handbook*, pages 48–1, 1987.
- [9] Sindre T Hilden, Halvor Møll Nilsen, and Xavier Raynaud. Study of the well-posedness of models for the inaccessible pore volume in polymer flooding. *Transport in Porous Media*, 114(1):65–86, 2016.
- [10] Kristian Jørgensen. Implementation of a surfactant model in mrst with basis in schlumberger’s eclipse, 2013.
- [11] Stein Krogstad, Knut-Andreas Lie, Olav Møyner, Halvor Møll Nilsen, Xavier Raynaud, Bård Skaflestad, et al. Mrst-ad—an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In *SPE reservoir simulation symposium*. Society of Petroleum Engineers, 2015.
- [12] Larry W Lake, RT Johns, WR Rossen, and Gary Pope. *Fundamentals of enhanced oil recovery*. Society of Petroleum Engineers, 2014.
- [13] Randall J LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.
- [14] KA Lie, TS Mykkeltvedt, and X Raynaud. Fully implicit higher-order schemes applied to polymer flooding. In *ECMOR XV-15th European Conference on the Mathematics of Oil Recovery*, 2016.
- [15] J Liu, GA Pope, K Sepehrnoori, et al. A high-resolution, fully implicit method for enhanced oil recovery simulation. In *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers, 1995.
- [16] AT Morel and R Bodenmann. Stability analysis for the method of transport. Eidgenössische Technische Hochschule [ETH] Zürich. Seminar für Angewandte Mathematik, 1996.
- [17] Gary A Pope et al. The application of fractional flow theory to enhanced oil recovery. *Society of Petroleum Engineers Journal*, 20(03):191–205, 1980.
- [18] Robert D Richtmyer and KW Morton. Difference methods for initial-value problems. 1967.
- [19] J Sagen, B Cvetkovic, E Brendsdal, G Halvorsen, YL You, T Bjørnstad, et al. Reservoir chemical-thermal simulation with tracers. In *European Petroleum Conference*. Society of Petroleum Engineers, 1996.

-
- [20] A Settari and K Aziz. Treatment of nonlinear terms in the numerical solution of partial differential equations for multiphase flow in porous media. *International Journal of Multiphase Flow*, 1(6):817–844, 1975.
- [21] Kenneth S Sorbie. *Polymer-improved oil recovery*. Springer Science & Business Media, 2013.
- [22] Robert C Wattenbarger, Khalid Aziz, FM Orr Jr, et al. High-throughput tvd-based simulation of tracer flow. *SPE Journal*, 2(03):254–267, 1997.