NTNU
Norwegian University of
Science and Technology

Mathematical modeling and numerical study of viscous fingering

Ferenc Székely

Master of Science in Mathematical Sciences
Submission date: May 2018
Supervisor: Xavier Raynaud, IMF
Co-supervisor: Halvor Nilsen, Sintef

Norwegian University of Science and Technology
Department of Mathematical Sciences

Table of Contents

Summary

The main goal of the thesis is to simulate and analyze non-linear partial differential equations effeciently in Matlab. Immiscible displacement of oil by water in porous media is simulated and analyzed in Matlab. We visualized different scenarios in two dimensions by using the MRST toolbox. Multiple initial states were used on various mesh sizes and we determined optimal grid sizes to converge to the correct number of fingers. We examined the stability of viscous fingering theoretically and numerically with the help of automatic differentiation. We showed general results about the instability of the base state and determined the relationship between the growth rate and the number of fingers. We conducted a theoretical introduction of dimension splitting which could provide even faster and better results for the stability.

Introduction

Fluid flows in porous media have been a great interest for the last 70 years due to oil recovery. After primary production the pressure drops and a significant part of the oil remains in the ground. One of the most natural ways to mine the valuable resource from an underground reservoir is to drill two wells into the ground. We inject a liquid (e.g. water) or a gas into the system through one well and hopefully the other liquid (usually oil) comes out through the production well because of the pressure difference. Unfortunately, in practice we find that water finds his way through the oil and nothing but water comes out. The problem is that the breakthrough is happening way too fast. The phenomena is known as viscous fingering and it is attributable mainly to the viscosity differences between the fluids. The water phase (part) is moving faster and displaces the oil phase.

In this thesis we would like to focus on multiphase immiscible fluid flows where the phases do not interact on a molecular level (i.e. not free to move) and there is surface tension between the phases.

In the next few chapters we attempt to give a brief description of the governing physical laws of the system. We try to cover all the necessary details, but some concepts are not fully described and we advise the reader to look them up in the references for deeper understanding.

Then we derive the governing equations using these previously introduced concepts. This boils down to the pressure equation and the saturation equation.

Before we go into the details, we investigate the general behavior of the system. We classify the partial differential equations and form some expectations before solving them. This could give a good overview and a frame of reference if the deeper analysis is getting complicated.

In the continuation we focus on the discretization of the equations and how the mathematical model was built in order to efficiently use the numerical schemes to solve it. We superficially discuss how the finite volume framework applied to the cells.

Then we introduce how the MRST toolbox is used to simulate the viscous fingering through some elementary examples. We focus on giving the full picture without hiding the tiny details such that the reader can recreate our results with the help of this document and the codes.

Finally, we carry out a stability analysis for the base state, and we investigate the relationship between the growth rate and the number of fingers.

Chapter 1

Physical and mathematical description of the fingering

1.1 Physical properties of the phenomena

1.1.1 Representative elementary volume

One of the important concepts is Representative Elementary Volume (REV). We don't go into the details here how to choose an appropriate scale, we only introduce the idea of it.

The goal is to choose some minimum volume, which represents very well the physical properties of the reservoir. Ideally, all the petrophysical properties of the flow are constant on a well chosen length scale.

For example, porosity is the ratio of the pore spaces and the rocks in the medium $\Phi = \frac{V_v}{V_v + V_r}$ and it varies with our choice of the volume. In Figure 1.1 we can see how the porosity is changing by choosing different representations of the medium. Our goal is to work on a scale where the average of the porosity is representative for the porosity of the reservoir

1.1.2 Formulation of a new continuum model

In a Darcy-continuum description a fluid quantity at a point $\overrightarrow{x}$ represents an average over the REV. In the case of immiscible multiphase flow the elements are not mixing and there is an infinitely thin interface between them. At a point $\overrightarrow{x}$ it is either totally filled with rock, oil or water. This can be well seen in Figure 1.2.

In the Darcy representations the oil-water concentration (later saturation) will show a continuous picture. So the transmission from one to another phase is continuous like in Figure 1.3.

[6]

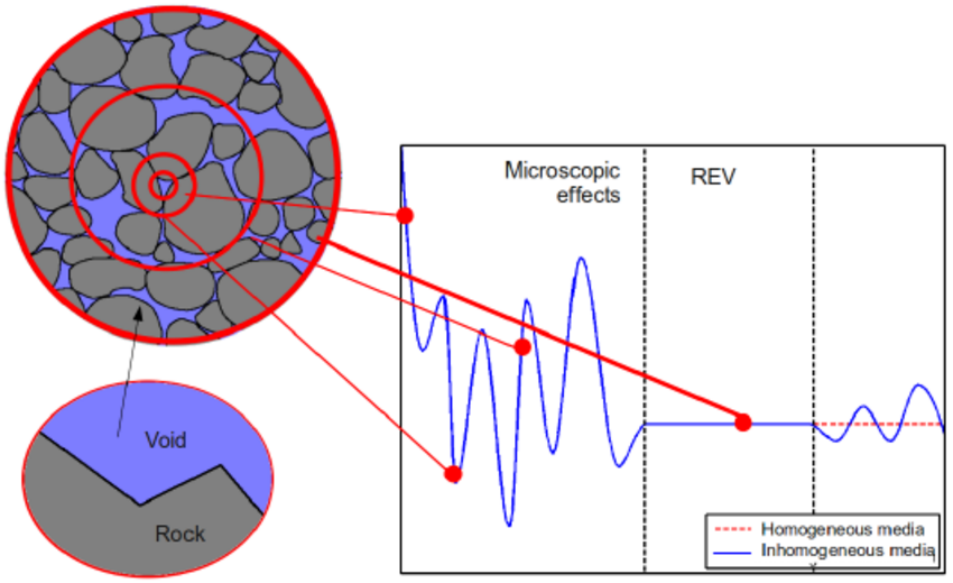


Figure 1.1: Example of REV

1.1.3 Darcys law

Henry Philibert Gaspard Darcy was a French hydraulic engineer who was the pioneer in the field of the one phase fluid flow in porous media. He designed a water tank where he made experiments and he discovered the following relationship between the flow rate and the pressure difference which attributed here to the hydraulic pressure. We tend to follow the notations from [6].

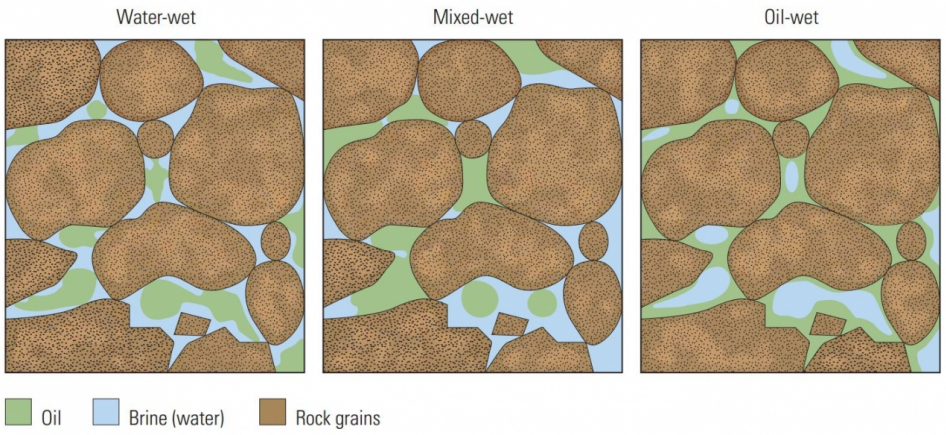
$$\frac{Q}{A} = \kappa \frac{h_t - h_b}{L}. \tag{1.1}$$

In equation (1.1) κ is the hydraulic conductivity, L is the length and h is the height. The quantity $\frac{Q}{A}$ is the Darcy flux, which represents the volume per fluid per total area and time and has dimensions $[m/s]$.

Sometimes it is referred as velocity, but actually it is not in the microscopic sense. Since some parts of the cross sectional area are blocked by (in this case) sand (later by rocks), it represents the macroscopic velocity which calculated on the REV discussed in the previous Section 1.1.1. But we will use Darcy velocity and by that we refer to the concept described above. In slightly different notation the law for one phase flow:

$$\overrightarrow{v} = -\frac{K}{\mu}(\nabla p - g\rho\nabla z). \tag{1.2}$$

Equation (1.2) expresses simply conversation of momentum and it can be derived from the Navier-Stokes equations.



[10]

Figure 1.2: Multiphase flow in Porous media

1.1.4 Saturation

In Section 1.1.2 we defined a continuum model where more than one phase can occupy a point as we could see in Figure 1.3. This leads to some reasonable assumptions and we introduce a new quantities.

One of this is the saturation S_α which defines the portion occupied by phase α. Furthermore, we assume that the fluids fill completely the void space between the rocks. This gives us the relation between saturation:

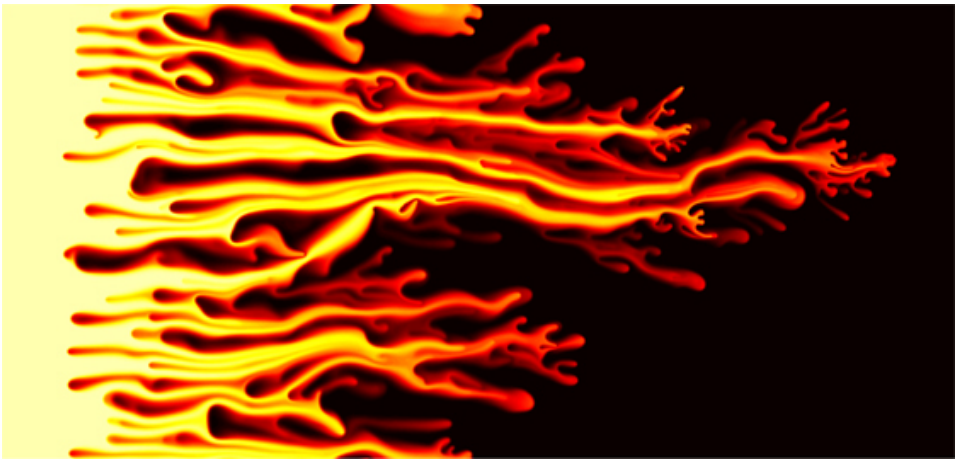
$$\sum_\alpha S_\alpha = 1. \tag{1.3}$$

In theory any phase saturation can vary from 0 to 1, but in reality system when water reaches the $irreducible\ water\ saturation$, it becomes immobile. It will be denoted with S_wr. We will talk about these residual values more in the upcoming sections.

1.1.5 Wettability

As we mentioned before immiscible fluids are separated by a well-defined, infinitely thin interface. This interface has a surface tension which measures the forces one must apply in order to change the shape of it this interface. Naturally. the system tries to minimize this surface energy. This results in the interface of a droplet of one phase contained within another phase to assume a spherical shape. The interface tension will keep phases apart, irregardless of the size of the droplet.

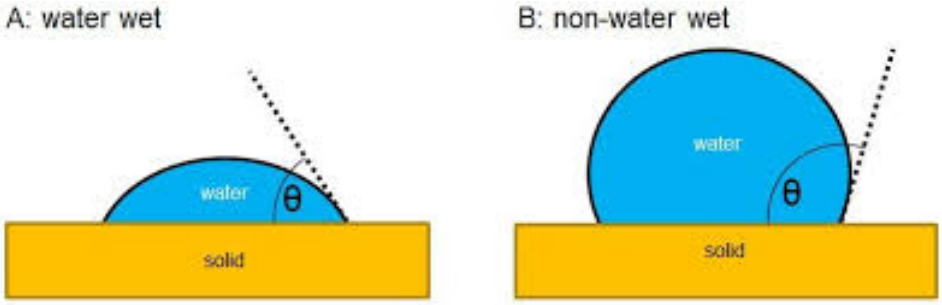
On the other hand the flow will be strongly affected by how the phases attach to the interface of the solid material (e.g. rock). The ability of a liquid to main contact with a solid interface is called wettability and it is determined by adhesive forces when the liquid and solid are in contact.



[11]

Figure 1.3: Saturation profile in two phase flow

These intermolecular forces will cause water droplets to spread across the mineral surface. In the case when both phases are present in a pore, one is more attracted to mineral particles and we call then system water wet or non-wet. In Figure we can see this behavior. Similarly we are going to call one of the phases wetting phase, while the other will be the non-wetting phase.



[13]

Figure 1.4: Water or non-water wet systems

The relationship between the contact angle θ the interface energies can be written by using Young equation:

$$\cos\theta = \frac{\sigma_{os} - \sigma_{ws}}{\sigma_{ow}}. \tag{1.4}$$

Where the subscript s and σ refers to the solid and the surface energy, respectively. Here we note the first case on Figure 1.4 is more often in nature and it the porous media is called hydrophilic or water wet.

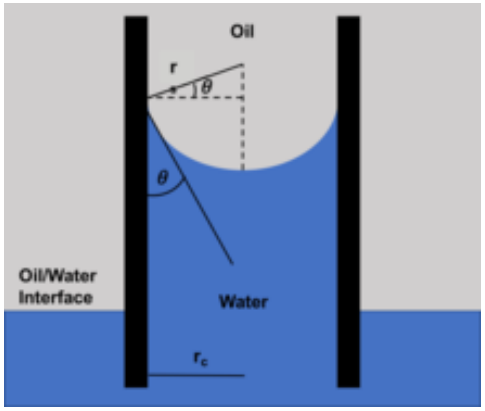
1.1.6 Capillary Pressure

In straight consequence of the surface tension the pressure will be different in the phases. This difference is called the capillary pressure:

$$p_c = p_n - p_w, \tag{1.5}$$

where in equation (1.5) the capillary pressure is always positive because the pressure in the non-wetting phase is greater than the pressure in the wetting phase. According to this we can write p_{cow} (or p_{cwo}) in most of the cases (in oil-wet systems).

Therefore, in narrow spaces (capillary tubes) liquids can move against gravity forces when the capillary pressure overcomes the other external forces. On Figure 1.5 we can see a water-wet system with $\theta < \pi/2$ and the oil-wet system can be easily imagined with $\theta \geq \pi/2$.



[1]

Figure 1.5: Behavior in capillary tubes

The intuition behind this is that whether the force between the fluid particles (cohesion forces) or the forces between the liquid and the solid interface (adhesion forces) are bigger. In the first case which can be seen on Figure 1.5 we say that a concave meniscus is formed. On the other hand, when adhesion is smaller than cohesion a convex meniscus is formed. In the equilibrium state of air-water capillary tube equation when the forces are balancing each other we can write the following equation for the upward forces (capillary force).

$$2\pi r(\sigma_{as} - \sigma_{ls}) = 2\pi r\sigma\cos\theta. \tag{1.6}$$

So we get the capillary force in the form of

$$p_c = \frac{2\pi\sigma\cos\theta}{r}, \tag{1.7}$$

after dividing by $2\pi r$. The downward forces are derived from Archimedes' law. In equilibrium

$$p_c = \Delta\rho g h. \tag{1.8}$$

The void space will supposedly contain a lot of narrow pores which could be considered as capillary tubes of different radius. In (1.7) one can observe that the capillary pressure increases with a decreasing tube radius, which means it will play an important role in establishing the fluid distribution in the reservoir.

Now oil enters into a water filled porous medium from downward. Then the buoyancy force must exceed a minimum capillary pressure in order to enter to the void space. The capillary pressure that is required to force an oil droplet into the rock is called the entry pressure. We can imagine that in the medium there are many capillary tubes with different radius and according to equation (1.7)] it will be the tube with the smallest capillary pressure (i.e. the wildest tube). As the pressure increases between the buoyant oil and the water, the oil will be able to penetrate smaller and smaller pore throats and thus the water saturation drops there. This leads to a relationship between the capillary pressure and the saturation.

$$p_{cnw} = p_n - p_w = P_c(S_w). \tag{1.9}$$

This relationship can be seen in Figure 1.6. However the the shape of the curve is strongly dependent of the pore sizes. If most of the scales are similar sizes, they will be quickly filled with the non wetting fluid when the entry pressure is exceeded. Hence the saturation drops rapidly with the increasing capillary pressure. Logically, if there is a higher variation in size among the spores, the decrease in saturation will be more gradual.

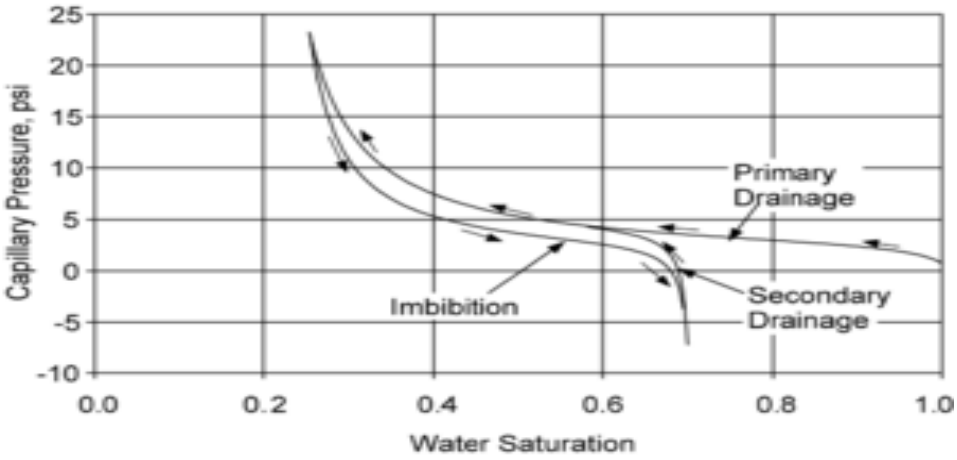
This example was based on the assumption that a non wetting phase is displacing a wetting fluid. This type of displacement is called $drainage$ which refers to the fact that the saturation of the wetting phase is decreasing. The opposite of drainage is called imbibition, occurs when a wetting fluid displaces a non wetting fluid.

So in the second case we inject water to flush out the oil from the reservoir. During the process the water saturation increases with more water injected into the system. However the oil saturation does not drop below a residual or immobile oil saturation (see Figure 1.6). The drainage and imbibition curves usually do not coincide. This is an example of $hysteresis$, when the behavior of the system depends not just only the current but also on the previous history.

It is often practical to express $P_c(S_w)$ as an analytical or tabulated function. In most of the cases it is done by using experimental results. Since each sample will generate a different capillary curve because of the different pore sizes. We normalize the data with the Leverett J-function.

$$J(S_w) = \frac{P_c}{\sigma \cos\theta}\sqrt{\frac{K}{\phi}}. \tag{1.10}$$

Because of the extent of the project exercise we cannot continue this chapter, but we note that there are many cases with different behaviors, thus capillary functions. We built significant parts of this section on [6] and we encourage the reader to look into it if interested in the behavior of different capillary function.



[8]

Figure 1.6: Primary drainage, imbibition and secondary drainage in water wet systems

1.1.7 Relative Permeability

Permeability first appears in one phase models and it is the parameter affects how fast the fluid flows through the medium. It is denoted with K and called the absolution permeability which measures the transmission of the rock.

In theory, this is an intrinsic property, however in reality this quite true. Because on microscale level there are interactions between the rock and the fluid which can modify the permeability. In addition, gases do not have no-slip condition and that results in that gases adhere differently to the interface.

In case of multiphase flow, each phase α will have an effective permeability ${K_\alpha}^\epsilon$ and this has to be smaller than K. The latter comes trivially from the fact that in case of immiscible fluid flows there is an additional capillary pressure, which the fluid should overcome in order to displace the other fluid. Thus the relationship is given by

$$\sum_\alpha {K_\alpha}^\epsilon < K. \tag{1.11}$$

In order to model this properly, we introduce a new property called relative permeability which is defined as

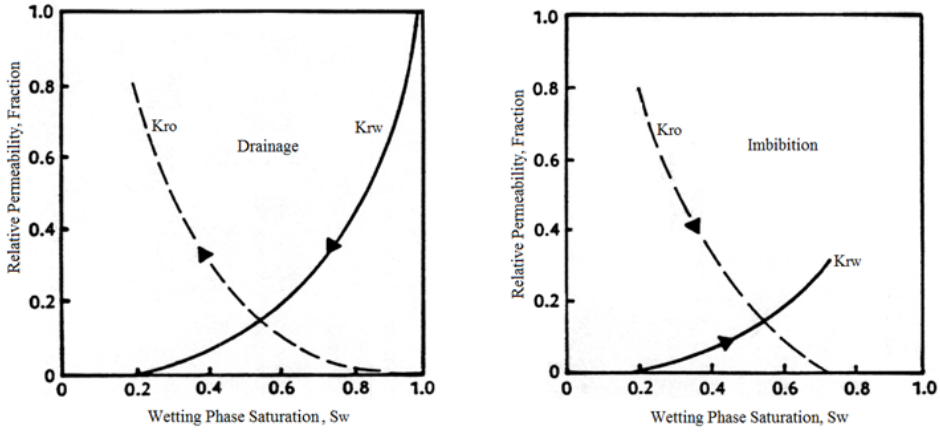
$$k_{r\alpha} = \frac{{K_\alpha}^\epsilon}{K}. \tag{1.12}$$

From (1.11) and (1.12) follows that $\sum_\alpha k_{r\alpha} < 1$. For anisotropic media, the relationship between k and K, however the relations are often defined as

$${K_\alpha}^\epsilon = k_{r\alpha} K. \tag{1.13}$$

Moreover the relative permeabilities will be the function of saturation, $k_{rw} = k_{rw}(S_w)$ and $k_{rn} = k_{rn}(S_w)$, where n and w refers to wetting and non-wetting, respectively. We

already discussed drainage and imbibition, in Figure 1.7 we can see how the relative permeability is changing during the processes.



[9]

Figure 1.7: Relative Permeability

We stop here and we would rather discuss how permeability is expressed in the special cases. The reader can find more information in [6].

1.2 Behavior of the system

So far we gather the necessary components and concepts which will appear in the equation. Naturally, there are very special cases when other petrophysical parameters can play an important role and we introduce them when it becomes necessary.

In this chapter, we are going to formulate the model and analyze its physical behaviour. We followed the outline in [6], but it was extracted, simplified and customized for our purpose.

1.2.1 Mass conservation for components

The phases α satisfy mass conservation, which gives us a system of N equations for N immiscible fluids. One can write for each component,

$$\frac{\partial}{\partial t}(\phi \rho_\alpha S_\alpha) + \nabla \cdot (\rho_\alpha \overrightarrow{v}_\alpha) = \rho_\alpha q_\alpha. \tag{1.14}$$

Each phase can contain different chemical properties, but these are considered as single components since there is no transfer between the phases.

In Section 1.1.3 we derived the relationship between pressure and Darcy flux for one phase flow. We extend that result and write

$$\overrightarrow{v}_\alpha = -\frac{Kk_{r\alpha}}{\mu_\alpha}(\nabla p_\alpha - g\rho_\alpha \nabla z). \tag{1.15}$$

Furthermore, we can simplify by introducing new notation of the phase mobility $\lambda_\alpha = Kk_{r\alpha}/\mu_\alpha$, or the relative phase mobility $\lambda_\alpha = \lambda_\alpha K$.

One of the possible ways to write down our system if we express the saturation as the function of pressures. We assume that the capillary pressure has an unique inverse and we end up with a system of two unknowns, pressure and saturation. However, both phases are present in the equations and this will make the system highly coupled and creates difficulties for numerical solutions.

We would rather continue with a different approach presented below.

1.2.2 Fractional flow formulation for incompressible flow

We choose the water saturation and the oil pressure as primary unknowns and we transform the equations accordingly and we have the trivial relationship from equation (1.3) for the saturation between phases,

$$\begin{aligned} \frac{\partial}{\partial t}(\phi\rho_w S_w) + \nabla\cdot\Big(\frac{\rho_w K k_{rw}}{\mu_w}(\nabla p_n - \nabla P_c(S_w) - \rho_w g\nabla z)\Big) &= \rho_w q_w, \\ \frac{\partial}{\partial t}(\phi\rho_n(1-S_w)) + \nabla\cdot\Big(\frac{\rho_n K k_{rn}}{\mu_n}(\nabla p_n - \rho_n g\nabla z)\Big) &= \rho_n q_n. \end{aligned} \tag{1.16}$$

In case of incompressible flow we have constant densities. This could be a good approximation of the flow is not totally incompressible but the base pressure in the reservoir is very high. Density varies a lot when pressure is relatively small, but as pressure raises the density tends to be constant after a while.

The goal of this formula is to simplify the mass conservation equation into

$$\rho\frac{\partial S_\alpha}{\partial t} + \nabla\cdot\overrightarrow{v}_\alpha = q_\alpha. \tag{1.17}$$

In order to achieve this, we write the total flux

$$\begin{aligned} \overrightarrow{v} &= \overrightarrow{v}_n + \overrightarrow{v}_w = -\lambda_n\nabla p_n - \lambda_w\nabla p_w + (\lambda_n\rho_n + \lambda_w\rho_w)g\nabla z, \\ &= -(\lambda_n+\lambda_w)\nabla p + \lambda_w\nabla p_c + (\lambda_n\rho_n + \lambda_w\rho_w)g\nabla z. \end{aligned} \tag{1.18}$$

We now add the two equations and then use the the relationship between saturation that $S_w + S - n = 1$, and we have the pressure equation

$$\phi\frac{\partial}{\partial t}(S_n + S_w) + \nabla\cdot(\overrightarrow{v}_n + \overrightarrow{v}_w) = \nabla\overrightarrow{v} = q_n + q_w. \tag{1.19}$$

We note the total mobility with $\lambda = \lambda_w + \lambda_n = \lambda K$ and the total source with $q = q_n + q_w$. We substitute into (1.17) and (1.19), then we collect all the variables containing pressure on the left-hand side and the rest on the right-hand side, we obtain

$$-\nabla(\lambda k\nabla p_n) = q - \nabla[\lambda_w\nabla p_c + (\lambda_n\rho_n + \lambda_w\rho_w)g\lambda z)]. \tag{1.20}$$

In other words the diversion of the (Darcy's) flux is equal what is produced through the wells, but we have some additional difficulties. The coefficients are variables and the right hand side is dependent on saturation through the relative mobilities. We call this equation a Poisson-type equation.

To find an equation for the water saturation, we multiply the velocity with the relative mobility of the other phase, subtract the results, and use Darcy's law to obtain

$$\begin{aligned}\lambda_n \overrightarrow{v}_w - \lambda_w \overrightarrow{v}_n &= \lambda \overrightarrow{v}_w + \lambda_w \overrightarrow{v}, \\ &= \lambda_n \lambda_w K(\nabla p_w - \rho_w g \nabla z) + \lambda_w \lambda_n K()\nabla p_n - \rho_n g \nabla z \\ &= \lambda_w \lambda_n K[\nabla p_c + (\rho_w - \rho_n) g \nabla z].\end{aligned} \tag{1.21}$$

We express $\overrightarrow{v}_w$ and use (1.17), we obtain the $saturation\ equation$ or $transport\ equation$:

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot [f_w(\overrightarrow{v} + \lambda_n \triangle \rho g \nabla z)] = q_w - \nabla \cdot (f_w \lambda_n P_c' \nabla S_w) \tag{1.22}$$

Where $\triangle \rho = \rho_w - \rho_n$ and the fractional flow function is defined as $f_w = \frac{\lambda_w}{\lambda_n + \lambda_w}$ (see in Figure 1.8).

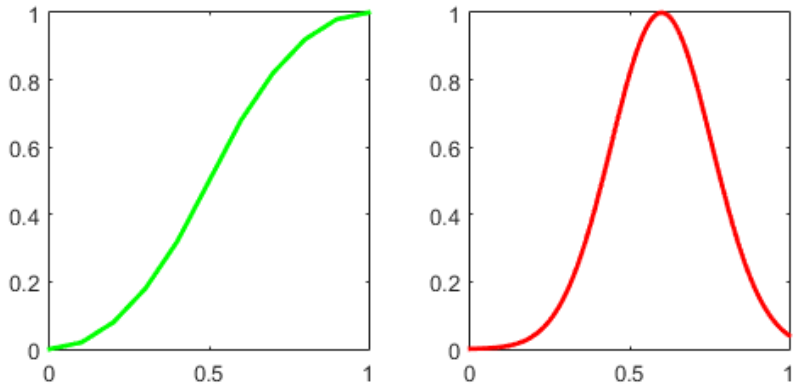


Figure 1.8: Fractional flow function (left) and gravity segregation function (right), while the x-axis shows the water saturation in both figures

1.2.3 Hyperbolic behaviour

As we can see in (1.22) we have three driving forces, viscous advection $f_w \overrightarrow{v}$, gravity segregation $\lambda_n \triangle \rho g \nabla z$ and capillary forces $f_w \lambda_n P_c' \nabla S_w$.

At first sight the equation looks parabolic because of the presence of the second order capillary term. The problem is well-posed because $P_c' < 0$ which makes the overall term positive.

However if we take a closer look at the parabolic term, we can see that λ_n is zero at the endpoints. This means that the second order term vanishes where the saturation is

zero or 1. In Figure 1.3 we can observe that in major part of the domain the saturation is either zero or one. This means the capillary term vanishes and if we look at the equation at reservoir level, the equation becomes parabolic.

Until parabolic equations tend to have an infinite speed of propagation, hyperbolic equations have a finite speed of propagation.

Furthermore, if the viscous term is dominant and we can neglect the gravity and capillary forces, then the flow will be co-current because the $f'_w > 0$. On the other hand the gravity segregation function typically has a bell shape, which means that it could create a counter-current flow.

1.3 Transport equation in 1D and the B-L case

We will use the theory of one dimensional transport equation to find solutions for the Buckley-Leverett case. We will proceed from simple linear cases to more complex problems and finally deal with the formulation of shocks or rarefaction waves.

1.3.1 Linear flux and the method of characteristics

We write the scalar conservation law for one dimension without sinks or sources. The equations and notations are in line with the textbook [6].

$$\frac{\partial u}{\partial t} + \frac{v}{\phi}\frac{\partial f(u)}{\partial x} = 0 \tag{1.23}$$

Where u will be replaced with S, but it is not limited and could be any physical quantity being transported through the medium. First, we see what the constant v is in our case. We know the elliptic pressure equation derived from Darcys law is

$$v'(x) = q, \qquad v(x) = -\lambda(x)p'(x) \tag{1.24}$$

Since we do not have sources and sinks $q = 0$, then v is constant in the domain $[0, \infty)$. Furthermore, we can scale the equation with $t^* = t\frac{v}{q}$ and we obtain

$$\frac{\partial S}{\partial t} + \frac{\partial f(S)}{\partial x} = 0 \tag{1.25}$$

We would like to find the speed of propagation for constant saturation values. We write the formula for the total differential

$$dS = \frac{\partial S}{\partial t}dt + \frac{\partial S}{\partial x}dx \tag{1.26}$$

and set $dS = 0$. We express $\frac{\partial S}{\partial t}$ and write

$$\frac{dx}{dt}\Big|_{dS=0} = \frac{df(S)}{dS} \tag{1.27}$$

Here, we assume that f is a function of S only, then it reads from equation (1.27) that constant saturation propagates through straight lines.

$$x(t) = x_0(t_0) + \frac{df(S)}{dS}(t - t_0) \tag{1.28}$$

Furthermore, we need initial and boundary conditions to obtain a well-posed problem.

In special cases, for example for linear fractional flow function, e.g. $f(S) = aS$, the solution can be given in a closed form as below

$$S(x,t) = \begin{cases} S_{0,x}(x - at), & x \geq at \\ S_{t,0}(at - x), & x < at \end{cases} \tag{1.29}$$

This means simply that the initial and the boundary data and propagates it along the x-axis with speed a. This way of solving the equation is called $themethodofcharacteristics$.

1.3.2 Shocks and rarefaction waves

It was discussed before in Section 1.2.2 that fractional flow function tends to have a typical S shape (See Figure 1.8). In addition, it has an inflection point where the second derivative changes sign and $f'(S)$ has a maximum (in our case, generally it could be a minimum as well).

Some points travel faster than others, when we have a nonlinear function. Essentially, we have two cases which result in two different behaviours. When we are analyzing the B-L case, we have the two cases present in our solution in different parts of the omain.

We are going to analyze separately the cases when the second derivative of the fractional flow function has different signs.

Rankine-Hugoniot shocks

First, we would like to deal with the case when the second derivative of the fractional flow function is strictly negative. Intuitively, we can imagine two points travelling with different speed. Moreover, we know that $f'(a) < f'(b)$ if $a > b$. So the fluid particle, which starts at point a is going to "catch" the fluid particle, which originates from point b. We will end with a multivariate solutions which is clearly not physical (See in Figure 1.9).

To avoid this, we replace the multivariate solution with a discontinuity. However, we can't interpret the solution in the classical sense, since the derivatives do not exist pointwise. We define the weak form and with it a weak solution. We multiply the solution with a test function φ of compact support and integrating the results over time and space

$$\int\int \left[\frac{\partial S}{\partial t} + \frac{\partial f(S)}{\partial x}\right] \varphi(x,t) dx dt = 0. \tag{1.30}$$

We integrate by parts

$$\int\int \left[S\frac{\partial \varphi}{\partial t} + f(S)\frac{\partial \varphi}{\partial x}\right] dx dt = 0. \tag{1.31}$$

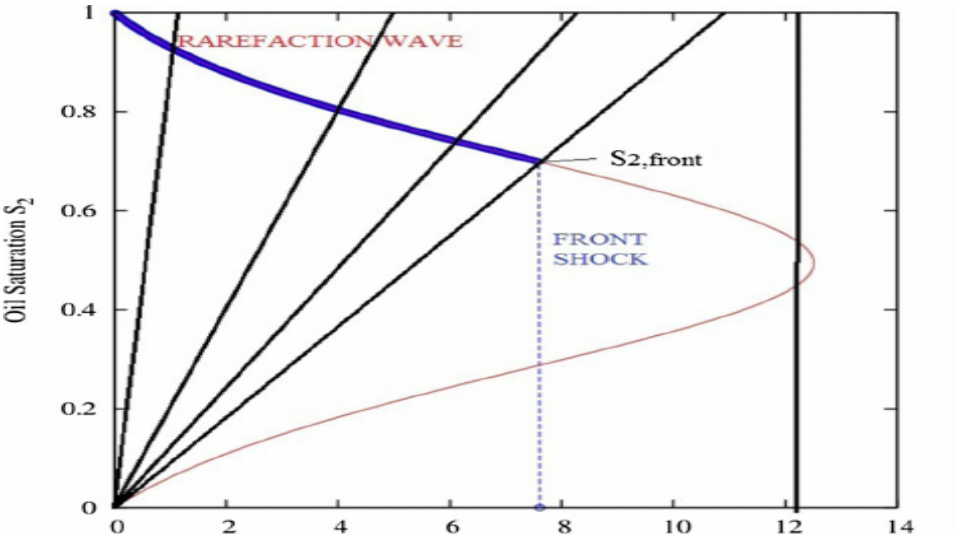
A weak solution is defined as any solution that satisfies (1.31) for all smooth and compactly supported test functions φ. We aim to find a propagating discontinuity in Figure 1.9. We

will denote the front $x_s(t)$ and we choose two points a and b such that $a < x_d(t) < b$. We use the integral form of the saturation equation

$$f(S_1) - f(S_1) = \int_a^b \frac{\partial f(S)}{\partial x} dx = \frac{d}{dt} \int_a^b S(x,t) dx. \tag{1.32}$$

Then we transform the last integral with limits

$$\int_a^b S(x,t) dx = \lim_{\epsilon \to 0} \int_a^{x_s(t)-\epsilon} S(x,t) dx + \lim_{\epsilon \to 0} \int_{x_s(t)+\epsilon}^b S(x,t) dx. \tag{1.33}$$



[12]

Figure 1.9: The solution of the one dimensional Buckley-Leverett case

Leibniz rule can be applied because the integrals and we obtain

$$\frac{d}{dt} \int_a^{x_s(t)-\epsilon} S(x,t) dx = \int_a^{x_s(t)-\epsilon} \frac{\partial S}{\partial t} dx + S(x_s(t) - \epsilon, t) \frac{dx_s(t)}{dt}. \tag{1.34}$$

We let $a \to x_s(t) - \epsilon$ and then we have only the second term. We substitute everything back to equation (1.32), we get the Rankine-Hugoniot shock condition

$$\frac{dx_s(t)}{dt} = \frac{F(S^+) - F(S^-)}{S^+ - S^-}. \tag{1.35}$$

where the values $S^{\pm}$ will denote the saturation values immediately to the left and to the right to the shock. This condition provides a necessary relationship between the right and a left hand states, but it does not guarantee that the solution will be physical. Further investigation is required to find the physically correct solution. The sufficient condition is called the entropy condition. We stop here and continue with the case of rarefaction waves.

Rarefaction waves

The second case when the second derivative of $f(S)$ is strictly negative. Intuitively, we can imagine this that a point a, which was originally ahead of point b, is moving even faster and the distance will be bigger and bigger between them.

If we apply the method of characteristics, we will end up with no solution in parts of our domain. This can be observed in Figure 1.9.

This is actually can be considered as a Riemann problem, i.e., a conservation law with a constant left and right state. We denote these states with S_L and S_R. When $S_L > S_R$, the solution of the problem has the following form

$$S(x,t) = \begin{cases} S_L, & x/t < f_c'(S_L), \\ S_{t,0}(at - x), & f_c'(S_L) < x \leq f_c'(S_R), \\ S_R, & x/t \geq f_c'(S_R). \end{cases} \tag{1.36}$$

Thee solution can be constructed by finding the upper concave envelope $f_c'(S)$ of the fractional flow function over $[S_R, S_L]$ and $f_c'(S)$ the inverse of its derivative.

This case could be analyzed in details and could lead to interesting findings. For example, we could find out how much oil is recovered until the water breaks through.

From now on we would like to focus on the two dimensional case and would like to analyze the appearence of viscous fingers.

Chapter 2

Model Examples

2.1 Two-dimensional models

In this section we describe the two dimensional model by using the concepts already introduced. Furthermore, we formulate the discrete form of the equations and describe some numerical methods which could be used for solving these types of problems.

2.1.1 The appearance of fingering effect

The fundamental reason for the appearance of the viscous effect can understood by examining the flux of liquid containing oil and water.

In two dimensional models we neglect the effect the gravity. So the driving force in the channel is the pressure difference between the left and right end of the tube.

Since oil is more viscous than water, it shows greater resistance against displacement. In the channels where there is less oil, water moves more freely and therefore faster than in other channels. As we can see in Figure 2.1, there are three different compositions of water and oil.

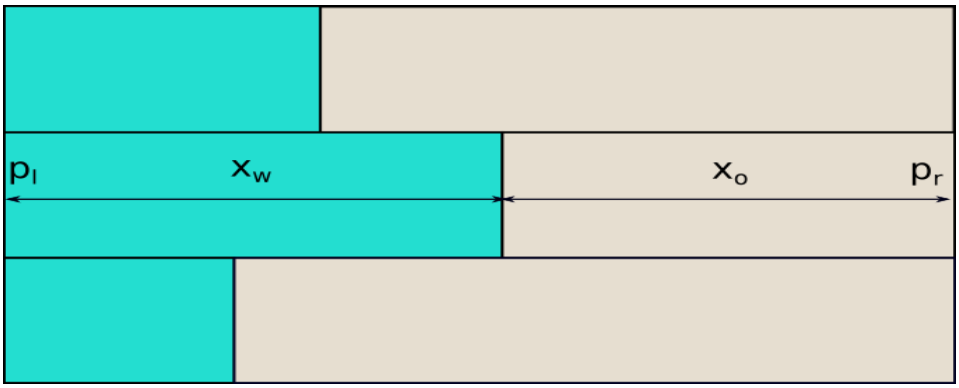


Figure 2.1: Fingers represented by individual channels

This formation will occur because of many reasons. One of them is that we can't flood the oil field perfectly from one side. That leads to small perturbations which will grow over time. On the other hand the porous medium is always highly heterogenous which makes this kind of formation inevitable.

If we use Darcy's law (1.1) and we write down for the phases $\alpha = o, w$, we obtain

$$u = -\frac{K_\alpha}{\mu_\alpha}\Delta p. \tag{2.1}$$

Here we neglected the gravitational forces in the channel due to very small Δz. If we fix the pressure at the end of the channels and substitute into equations for the phase, we find that a completely water filled channel has a larger flux due to the smaller viscosity of the water. We assume that we have started from fully oil filled channel and flooded with water from the left. Since the flux is a continuous function which means it must be increasing with increasing water saturation, because the initial state has a smaller flux than the final state. Thus we get

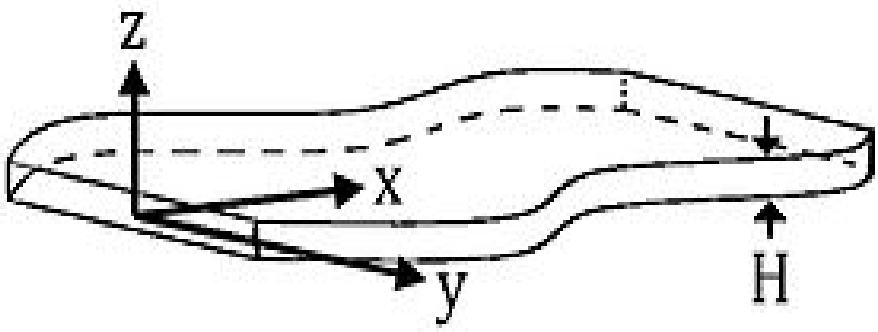
$$u = -\left(\frac{x_w}{\mu_w} + \frac{x_o}{\mu_o}\right) K\Delta p. \tag{2.2}$$

This is a very important mechanism because it means that the longer fingers will grow even faster than the smaller ones. This effect is fortunately controlled by the capillary pressure. Since fingers with very small radius will experience a very large resistance from the oil interface due to capillary forces. This is called the smearing effect and it will be discussed in more details later.

Many different problem formulations were designed to simulate viscous fingering. In the followings we review the fundamental principles of some models.

2.1.2 The Muskat problem

Sometimes porous medium will be represented by a Hele-Shaw cell (See Figure 2.2). The fluid is flowing between two plates where H is the distance between them and $H \to 0$.



[4]

Figure 2.2: Schematic of a two-dimensional Hele-Shaw cell

Two immiscible fluids separated by an infinitely thin interface that is modelled by a moving boundary. We have a constant higher pressure on the right end. We can write the Darcy equations for the phases $i = 1, 2$

$$u_i = -\frac{k}{\mu_i}\nabla p_i \qquad \text{and} \qquad \nabla \cdot u = 0. \tag{2.3}$$

At the moving boundary we have a pressure condition $p_1 = p_2$ and a kinematic boundary condition $u_1 \cdot \overrightarrow{n} = u_2 \cdot \overrightarrow{n} = v_n$, where $\overrightarrow{n}$ is the normal to the boundary and v_n is the normal velocity of the boundary.

The stability analysis of the planar interface has shown that the Muskat-problem is an ill-posed problem, i.e. the solution 'blow up' in finite time. In practice, we neglect some physical phenomenon which regularizes the problem and ensures that it becomes well-posed.

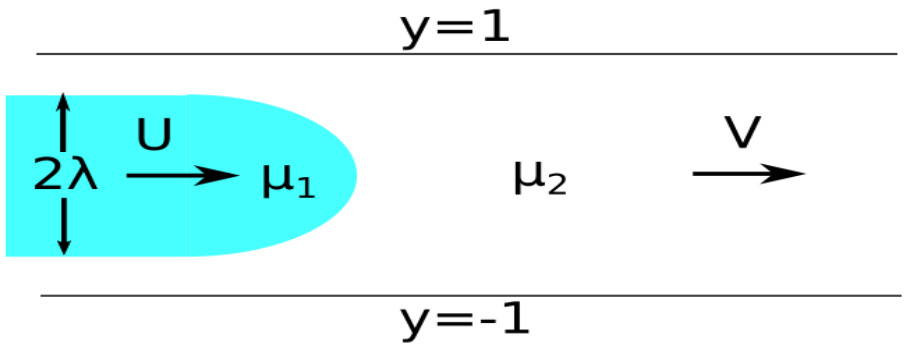


Figure 2.3: The Saffman-Taylor finger

For this unique problem there is a well known analytical solution. One single finger with viscosity μ_1 penetrates the other phase μ_2 in an infinite channel $-\infty < x < \infty$, $-1 < y < 1$ (see Figure 2.3). If at the right end ($x \to -\infty$) occupies a fraction of the channels λ and as $x \to \infty$ the velocity of the displaced fluid is V. The displacing fluid moves with constant velocity U. Then the interface is given by the formula

$$x - Ut = \frac{1-\lambda}{\pi} \log\left(\frac{1 + \cos\left(\frac{\pi y}{\lambda}\right)}{2}\right). \tag{2.4}$$

And the finger moves with velocity

$$U = \frac{MV}{1 + \lambda(M-1)}, \tag{2.5}$$

where $M = \frac{\mu_2}{\mu_1}$. One can observe that we have different values for every $0 < \lambda < 1$ and the speed of the finger can take any value between V and MV.

2.1.3 The Peaceman model

There are other models which are replacing the two phase flow with a one phase flow with a solvent. This implies that the concentration (exactly as the saturation) must be defined everywhere in the reservoir. and we modify the equation,

$$\phi \frac{\partial c}{\partial t} + \nabla \cdot (uc) = \nabla \cdot (D \nabla c) \tag{2.6}$$

. Regardless from the approach we arrive to very similar solutions in the simulations.

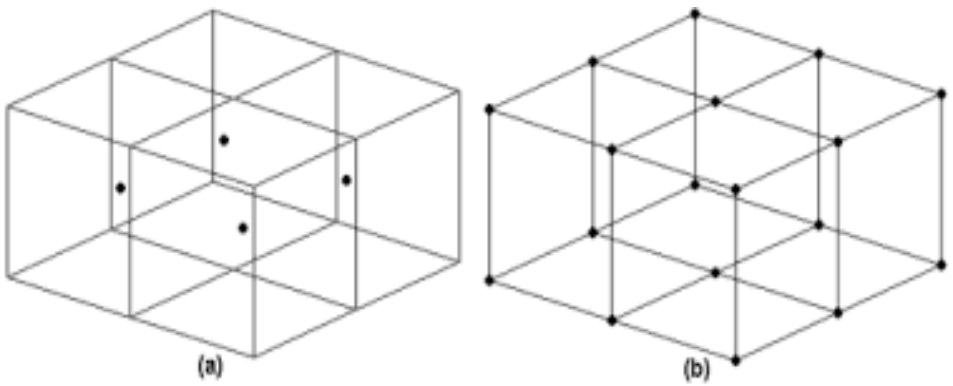
2.2 Discretization of the equation

In this chapter, we introduce the grid structure and the approach to solve these systems numerically. Later in the section we implemented a basic first order scheme and we will present the results here. We adapted here major parts from the textbook [6].

2.2.1 The Grid

A $grid$ is a tessalation of volumetric or planar object which consists of $cells$. The description of a grid can be given by the geometry, i.e. the shapes the of the cells the form the grid and the topology their topology that tells how the cells are connected. In 2D the cell is a closed polygon which is defined by its vertices and edges. In 3D, the object is a polyhedron for which the geometry is defined by vertices,edges and $faces$ (surface enclosed by edges).

Furthermore, we assume that the cells are not overlapping, i.e. every point each point in the grid is in a single cell. Two cells which share the same face are said to be connected, neighbours. Therefore, connections between cells can be described with edges and faces, so the topology of the grid can be defined by connections, sometimes referred as the connectivity of the grid.



[3]

Figure 2.4: Edges and centroids in single cells

When we are implementing grids in modeling, we have two important factors to consider. Optimization between generality and efficiency is not a trivial task at all.

2.2.2 Finite volume discretization

Finite volume discretization is built up on the idea of cell averages. Its big advantage that it is a conservative method regarding cell value averages, but it does not know the exact values at point such as in the center (centroid) of the cell or at the centroid of the faces. These values must be approximated. Lets formalize how we are seeking the solution and we are defining the sliding average below

$$\overline{u}(x,t) = \frac{1}{\Delta x}\int_{x-1/2\Delta x}^{x+1/2\Delta x} u(\xi,t)d\xi. \tag{2.7}$$

We are going to use the notation that $u_i(t) = \overline{u}(x_i,t)$ is the average in the cell i, $\Big\{[x_{i-1/2}, x_{i+1/2}]\Big\}$ [6]. Using this notation and we substitute into (1.23) with v and $\phi = 1$, we obtain a semi discrete-version

$$\frac{du_i}{dt} = \frac{1}{\Delta x_i}[f(u(x_{i-1/2},t) - f(u(x_{i+1/2},t)]. \tag{2.8}$$

Then we integrate over time from t_n to t_{n+1} and we get the following fully discrete equation form,

$$u_{n+1} - u_n = \frac{1}{\Delta x_i}\Big[\int_{t_n}^{t_{n+1}} f(u(x_{i-1/2},t)dt - \int_{t_n}^{t_{n+1}} f(u(x_{i+1/2},t)dt\Big]. \tag{2.9}$$

However, we don't know the values at $x_{1-1/2}$ or $x_{1+1/2}$. In order to be able to calculate the saturation values, we need the definition of the flux on the faces. We have only cell averages for the pressure and using Darcys equation, we can compute the $flux$ at the centroid of the cells. Therefore, we need an approximation on the faces. This will be carried out first with a simple upwind scheme using the value from the upstream end. Later we will get to higher order methods.

$$F^n_{x\pm1/2} \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+}} f(u(x_{i\pm1/2},t)dt. \tag{2.10}$$

Here there are two useful pieces of information. One is that we are using information from the upstream end. The second is that we have a hyperbolic equation which means that the propagation of the speed is finite. Thus the approximation will be using finite number of points to calculate the flux such that $F^n_{x\pm1/2} = F(u^n_{i-m},\dots,u^n_{i+l})$. To prove that the method is conservative we substitute (2.10) into (2.9) and multiply with Δx, then we obtain

$$\sum_{-M}^{N} u_i^{n+1}\Delta x = \sum_{-M}^{N} u_i^n \Delta x - \Delta t^n(F^n_{N+1/2} - F^n_{-M-1/2}). \tag{2.11}$$

If the initial solution $u_0(x)$ has a bounded support, it means that the flux terms will cancel out if we choose M,N sufficiently large. Thus, we end up with a conservative scheme

$$\sum_i u_i^{n+1}\Delta x = \sum_i u_i^n \Delta x = \cdots = \int u_0(x)dx. \tag{2.12}$$

Discrete grad and div operators

So basically we need the pressure gradients (in the cells) and the divergence of the saturation (on the faces) in order to calculate the saturation values. We described the grid by cells $c \in \mathbb{N}^{n_c}$ and by faces $f \in \mathbb{N}^{n_f}$. The divergence from a cell or into a cell (depending on the sign) is the sum of the fluxes through the faces. In other words, it will be a function of c which loops over the faces. Let v is the flux from $N_1(f)$ to $N_2(f)$. Hence, we get the flux for cell c. It can be written as

$$\text{div}(v)[c] = \sum_{f\in N(c)} v[f]\mathbf{1}_{\{c=N_1(f)\}} - \sum_{f\in N(c)} v[f]\mathbf{1}_{\{c=N_2(f)\}}. \tag{2.13}$$

The grad operator is easier to define, since that is only the difference between the neighbouring cells, i.e. it is a function of face f

$$\text{grad}(p)[f] = p[N_2(f)] - p[N_1(f)], \tag{2.14}$$

for any $p \in \mathbb{R}^{n_c}$. Now it is time to introduce the discrete relationship between flux and pressure. We did not discuss the two-point flux approximation (TPFA), but it is an immediate consequence of Darcys equation for the discrete case. We have v as a discrete approximation of the flux on the faces.

$$v[f] \approx \int_{\Gamma_f} \mathbf{K}(x)\nabla p \cdot \overrightarrow{n}_f ds, \tag{2.15}$$

where $\overrightarrow{n}_f$ is the normal to the face f, where the orientation is given by the grid. Writing the discrete version of Darcys law, we obtain

$$v[f] = -\mathbf{T}[f]\text{grad}(p)[f] \approx -\int_{\Gamma_f} \mathbf{K}(x)\nabla p \cdot \overrightarrow{n}_f ds, \tag{2.16}$$

where $\mathbf{T}[f]$ is the transmissibility of the face f. Summarizing the results, we end up with

$$\text{div}(v) = q, \tag{2.17}$$

$$v = -T\text{grad}(p). \tag{2.18}$$

Explicit and implicit time-discretizations

We have two options to solve one time step: explicitly or implicitly. The advantage of explicit methods, such as the forward Euler, are that they tend to be faster, but in practice

they are notoriously unstable. The reason for this is the CFL (Courant–Friedrichs–Lewy) condition, e.g. that the numerical scheme has to include the domain of dependence to be stable, in other words the time step should satisfy the $\Delta t \leq C\Delta x$ equation, where c is constant if $f(u)$ is linear, however, in our case it depends highly on x. We either check the CFL every time step and set Δt accordingly or we have to use another approach. We will choose the second alternative and use an implicit scheme.

2.2.3 Numerical example

In our first simulation, we are going to flood an oil field with water from the left and see how the water saturation changes over time. In order to simplify the model will be two dimensional to avoid gravitational forces. We will go through the codes to connect the previously introduced concepts with their forms implemented in practice.

Grid and material properties of fluids and rock

First, we set a very simple rectangular Cartesional grid and we build the geometry of cells and faces for it.

```
nx = 50;
ny = 50;
G = cartGrid([nx, ny],[1  1]);
G = computeGeometry(G);
```

Listing 2.1: Matlab example

$Cartgrid$ creates the grid and $computeGeometry$ constructs the structure of cells and faces for it. The rock and fluid properties are easily defined by the following lines.

```
rock = makeRock(G, 100*milli*darcy, 0.5);
fluid = initSimpleADIFluid('mu', ...
                           [1, 10, 1]*centi*poise, ...
                           'n', [1 1 1], ...
                           'rho',...
                           [1000, 700, 250]...
                           *kilogram/meter^3);
```

Where the second argument in makeRock defines the permeability and the third is the porosity of the rock. The fluid properties are viscosity, degrees of monomials (describing relative permeability) and density for water,oil and gas. We don't need gas properties for this simulation but it doesn't cause any trouble to leave them in the code.

Choosing the model

This is one of the crucial steps. The model knows basically all the information about the equations. It contains the function $EquationsOilWater.m$ which provides the mass conservation equations for water and oil saturation.

```
model = TwoPhaseOilWaterModel(G, rock, fluid);
```

Initial saturation

We do not want to overcomplicate things with the initial setup either. We break the symmetry in the second column, so the fingering can be more visible.

```
[ii, jj ] = gridLogicalIndices(G);
%first we just set the first column on the left boundary
sW = ones(G.cells.num, 1);
totheright = ii > 1;
sW(totheright) = 0;
%second column every second row is oil and others are water
sec_col = (ii == 2) & (mod(jj,2) == 1) ;
sW(sec_col) = 1;
%similary for mod 7
sec_col = (ii == 2) & (mod(jj,7) == 1) ;
sW(sec_col) = 1;
%create the state variable
s = [sW, 1-sW];
state = initResSol(G,100*barsa,s);
%plot the grid how it looks like
plotCellData(G, state.s(:, 1))
```

Boundary conditions

Setting up the boundary is trickier because boundary conditions will be built into sources. Lets state this formally and get to the coding after. For Ω_i, we write the saturation equation with source term q.

$$\int_{\Omega_i} (s_t + \nabla \cdot (su))dx = \int_{\Omega_i} q(x)dx,$$

$$\frac{d}{dt}\int_{\Omega_i} sdx + \sum_j \int_{\Gamma_{i,j}} (su)dx = \int_{\Omega_i} q(x)dx,$$

$$|\Omega_i|\frac{d}{dt}s_i + \sum_j \int_{\Gamma_{i,j}} (su)dx = \int_{\Omega_i} q(x)dx,$$

$$\frac{|\Omega_i|}{\Delta t}(s_i^{n+1} - s_i^n) + \sum_j \int_{\Gamma_{i,j}} s^{n+1}udx = \int_{\Omega_i} q(x)dx.$$

For internal cells, there is no problem, since we know the flux using upwind scheme and we know we take the saturation from the neighbouring cells. On the other hand, if Ω_0 is a cell on the boundary, the mass conservation for the cell is bit different. Let $u = 1$ and write

$$\frac{|\Omega_0|}{\Delta t}(s_0^{n+1} - s_0^n) - \int_{\Gamma_{-1,0}} s_{-1}^{n+1}dx + \int_{\Gamma_{0,1}} s_0^{n+1}dx = q_i.$$

We need boundary condition. Let us say that $s = 1$ on the left-hand side. Then equation above becomes

$$\frac{|\Omega_0|}{\Delta t}(s_0^{n+1} - s_0^n) + \int_{\Gamma_{0,1}} s_0^{n+1}dx = \left|\Gamma_{-1,0}\right| + q_i.$$

So it is important that on the left boundary how we set the saturation which is going to flow in from outside. In our case the left boundary is more important because we will set the pressure in a way such that that the fluid will move from left to right.

```
bc = [];
bc = pside(bc, G, 'Left', 150*barsa, 'sat', [1 0]);
bc = pside(bc, G, 'Right', 100*barsa,'sat', [0 1]);
```

In the second row we declare s_{-1}^{n+1} to be one (water). As we said before the saturation at the right boundary is not important because nothing will flow in from there, since pressure differences were initialized otherwise.

Solution of the equations

We have a highly nonlinear system with unknown saturation and pressure at the next time step. We are going to use automatic differentiation to solve these equations. The fundamental concept of automatic differentiation is to store the first derivative of every variable with respect to the space variable(in our case these are cells). By defining some rules for the derivatives of functions Matlab can easily compute the Jacobian of the system. Then we can use Newton-Rhaphson-method to solve the equations.

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2.19}$$

It is fast and effective and the Jacobian stores essential information about the behaviour of the system. For more information on automatic differentiation see [7].

Simulation and results

The rest of the code is storing and plotting the results. A realistic time scale is dependent on the grid scale.

```
dt = repmat(0.1*hour, 20, 1);
% set the schedule
schedule = simpleSchedule(dt, 'bc', bc);
[wellSols, states] = simulateScheduleAD(state,model,schedule,'Verbose',
    true);
%% Plot the segregation process
h = figure();
for i = 1:numel(states)
    figure(h); clf;
    % Neat title
    str = ['after ', formatTimeRange(sum(dt(1:i))), ' (Step #', num2str(i)
    , ')'];
    % Plotting
    plotCellData(G, states{i}.s(:, 1))
    title(['Water saturation ', str])
    % Make axis equal to show column structure
    axis equal tight off
    colorbar
    pause(0.2)
end
```

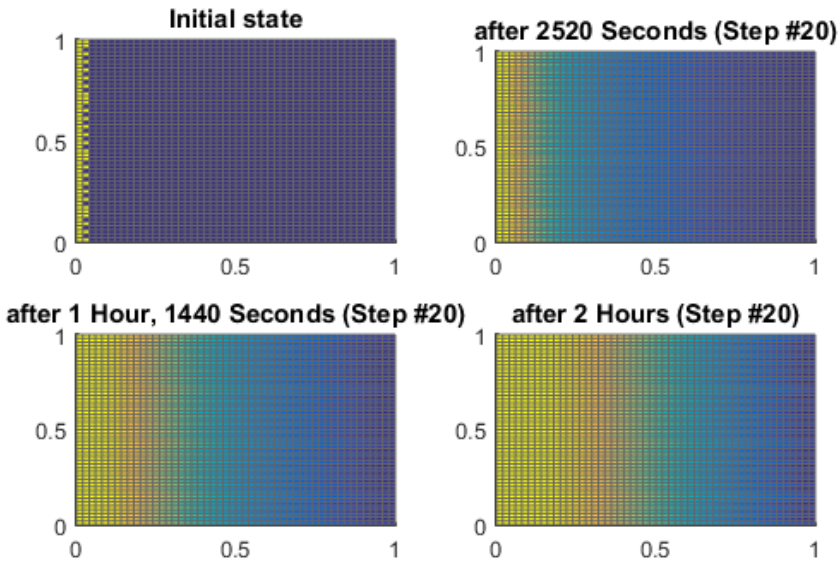


Figure 2.5: Evolution of water saturation

It is clear that the fingers which were bigger at the beginning evolve faster than the rest of the domain. However, we are not able to see really big differences in the evolution of the fingers in Figure 2.5. We are going to use higher order methods to simulate the same experiment and compare the results.

2.3 Shock in the moving frame

In this Section we would like to investigate the fingering subject to small perturbation in a moving coordinate system. In the one dimensional case we have already seen the fractional flow function (See Figure 1.8), which leads to shocks where the function is strictly convex with respect to water saturation S_w and generates rarefaction waves in the concave-region. For now we focus on the shock range, i.e. we stay within the range between 0.1 and 0.2 water saturation to make sure that shock occurs but no rarefaction waves. The moving coordinate will have the speed of the shock such that the fingering will be visible with respect to the shock. Let's derive the system from the two dimensional example, we have gone through before.

In Section 2.4 we provided a numerical solution to the two dimensional fingering for the equation

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot \vec{u}_w = 0, \tag{2.20}$$

where $\vec{u}_w$ is the Darcy flux and we can use equation (1.2), disregarding the gravitational

part. It results in

$$\phi \frac{\partial S_w}{\partial t} - \nabla \cdot (K \nabla P) = 0, \tag{2.21}$$

where K and P are the transmissibility and pressure, respectively. The code used a simple rectangular grid in Cartesian coordinates. However, we can choose a moving coordinate in the horizontal direction,

$$\xi = x + \upsilon t, \qquad Y = y, \qquad T = t.$$

Here we have the shock speed defined at (2.38). Hereafter, we are going to use the notations ξ, y and t. If we substitute into (2.21) with the new coordinates, then we obtain

$$\frac{\partial S_w}{\partial t} - \phi \frac{\partial S_w}{\partial x_1} \frac{\partial x_1}{\partial t} - \nabla \cdot (\lambda_\alpha K \nabla P) = 0. \tag{2.22}$$

We can bring the second term under the divergence operator and then we have a new fractional flow, which we can call real or effective fractional flow function. One can note that $\frac{\partial x_1}{\partial t} = \upsilon$ and we can write

$$\frac{\partial \phi S_w}{\partial t} - \nabla \cdot (\lambda_\alpha K \nabla P + \upsilon s_w \nabla x_1) = 0. \tag{2.23}$$

2.3.1 Discretization of the moving frame

We can discretize the scheme by using TPFA (two-point flux approximation), We define the transmissibilities on the face between the cells $T_{i,j}$ which consist of the half transmissibilities

$$T_{i,j} = \left(\frac{1}{t_{i,j}} + \frac{1}{t_{j,i}} \right)^{-1}, \tag{2.24}$$

where the half-transmissibility $t_{i,j} = \frac{1}{|c_{i,j}|^2} \overrightarrow{n}_{i,j} \cdot \overrightarrow{K} \overrightarrow{c}_{i,j}$ ($\overrightarrow{c}_{i,j}$ is the vector between the cells and the face centroids). So the discrete flux obtains the form of

$$v_{i,j} = T_{i,j}(p_i - p_j) \approx - \int_{\partial C_{i,j}} \overrightarrow{K} \cdot \overrightarrow{n}_{i,j} dA. \tag{2.25}$$

Now, we need to find the flux of the moving component of

$$\int_{\partial C_{i,j}} \phi \nabla x_1 \cdot \overrightarrow{n}_{i,j} dA. \tag{2.26}$$

Similarly, we need to define the pseudo transmissibilities $\tilde{T}_{i,j}$ here. It will have the exact same form as (2.24) but here instead of the $\overrightarrow{K}$ vector we have the porosity,

$$\tilde{t}_{i,j} = \frac{\phi_i}{|c_{i,j}|^2} \overrightarrow{n}_{i,j} \cdot \overrightarrow{c}_{i,j}. \tag{2.27}$$

Then we can write the total flux using the discrete operators introduced in Section 2.2.

$$u_w[f] = \lambda_w^f[f] T[f] grad(p)[f] - \upsilon_s s_w^f[f] \tilde{T}[f] grad(x_1)[f] \tag{2.28}$$

Unfortunately, we still don't know the exact saturations and mobilities at the faces. One can use an upwind scheme, but even for small perturbations it leads to instabilities.

2.3.2 Unstable upwind scheme

The major reason why the upwind scheme can be unstable lies in the different fractional flow functions. Contrary to the function for the steady coordinate system, the effective fractional flow is still convex but not monotone.

First, we had the fractional flow in Figure 1.8 and we can obtain a pure shock if we set

$$f(S) \leq \frac{S - S_R}{S_L - S_R} f(S_L) + \frac{S_L - S}{S_L - S_R} f(S_R). \tag{2.29}$$

The speed is given by the Rankine-Hugoniot condition

$$v_S = \frac{f(S_L) - f(S_R)}{S_L - S_R}. \tag{2.30}$$

We define the effective fractional flow as

$$f_{eff}(S) = f(S) + v_S S, \tag{2.31}$$

which leads to a convex function with a unique minimum (see Figure 2.6).

If we have an initial condition with a single jump between $i-1$ and i cells and the saturation is

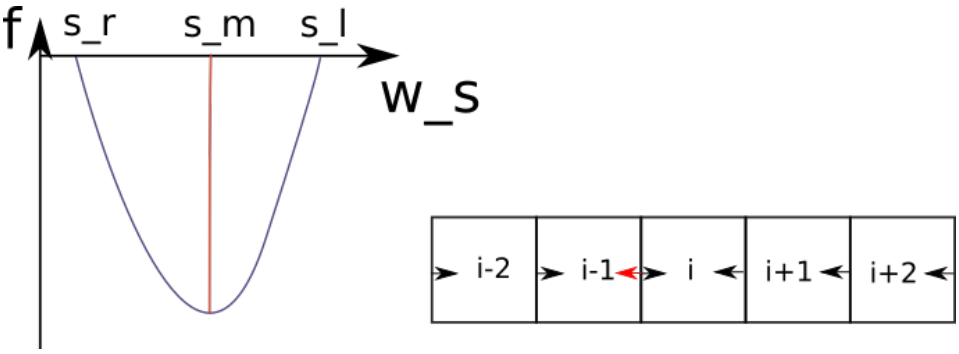


Figure 2.6: Effective fractional flow on the right, flux direction in the cells on the left.

$$S_j = \begin{cases} S_L, & if j < i, \\ S_R, & if j \geq i. \end{cases} \tag{2.32}$$

Then we encounter the problem that on the face $i - 1/2$ we get two contradictory upwind direction. From the fact that $f'_{eff}(S_L) > 0$, we would obtain that the flux is coming from left to right. On the other hand, the cell i is suggesting otherwise because $f'_{eff}(S_R) < 0$. We can assign the upwind direction according the highest absolute value of the derivative. For example, if $\left|f'_{eff}(S_L)\right| > \left|f'_{eff}(S_R)\right|$, then we choose the upwind direction from left to right (see Figure 2.6). Moreover, follows from the definition of the effective fractional flow that

$$f_{eff}(S_L) = f_{eff}(S_R).$$

Actually, this is not a contradiction yet, it simply means that we have a stationary solution.

However, if we consider a small perturbation in the saturation at the left hand side, i.e. we set $S_L = S_L + \epsilon$. This results in that the sum of the flux is positive

$$F_{i-1/2} - F_{i+1/2} = f_{eff}(S_L - \epsilon) - f_{eff}(S_R) > 0.$$

Therefore we have that

$$s_i^n = s_i^{n-1} + \Delta t(F_{i-1/2} - F_{i+1/2}),$$

i.e. s_i^n is increasing in n. This will lead to an unstable scheme.

2.3.3 The HLL fix

Harten, Leer and Lax entropy fix gives an idea how to fix this problem. Until the function is monotone, there is no need to change the upwind scheme. An additional intermediate state will be introduced $\tilde{S}_{i-1/2}$. The Rankine-Hugoniot shocks then reads

$$v_1 = \frac{f(\tilde{S}_{i-1/2}) - f(S_{i-1})}{\tilde{S}_{i-1/2} - S_{i-1}} \qquad \text{and} \qquad v_2 = \frac{f(S_i) - f(\tilde{s}_{i-1/2})}{s_i - \tilde{s}_{i-1/2}}.$$

We get that

$$\tilde{S}_{i-1/2} = \frac{f(S_i) + f(S_{i-1}) - v_2 s_i + v_1 s_{i-1}}{v_1 - v_2}.$$

Furthermore, we define

$$v_1 = f'(S_{i-1}) \qquad , \qquad v_2 = f'(S_i).$$

Then we alter the scheme where $v_1 < 0$ and $v_2 > 0$, i.e.where the function is not mono-

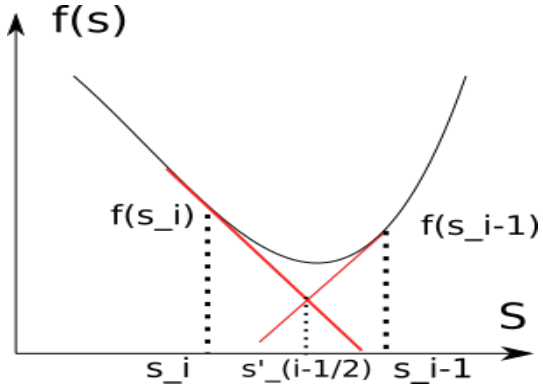


Figure 2.7: The Hll entropy fix

tone (See Figure 2.7). We obtain a scheme which is stable with

$$S_{i-1/2} = \begin{cases} S_{i-1}, & if v_1 > 0, \\ \tilde{S}_{i-1/2} & if v_1 > 0 \quad \text{and} \quad v_2 < 0, \\ S_i, & if v_2 < 0. \end{cases} \tag{2.33}$$

2.3.4 Implementation and results

We would like to present results for incompressible flow. We are going to investigate the differences between the moving frame and the usual Cartesian coordinates. Moreover, we test the fingering with capillary pressure on meshes with high and low resolution.

As we mentioned before in order to get a shock we need to stay in the convex range of the original fractional flow function. The fractional flow is defined as

$$f(S) = \lambda(S_w 0)/(\lambda(S_w) + \lambda(S_o)),$$

and calculated by the mobility functions of the fluids. This is a monotone function with an inflection point (see Figure 2.8). We are interested in the behaviour of the shock which occurs when water saturation is between 0.1 and 0.2. We calculate the shock speed according

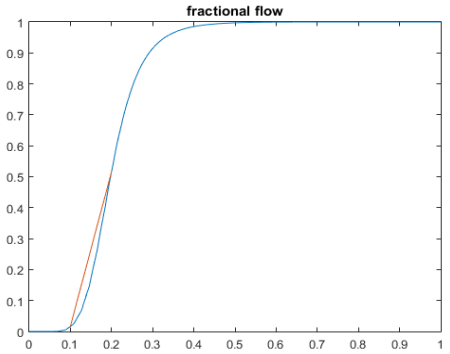


Figure 2.8: The convex range in the fractional flow function

to the formula in equation (2.30).

Boundary conditions

In MRST there are three routines for setting boundary condition. We use `addBC` to impose either flux or pressure conditions on the boundary faces. We can also directly call `pside` or `fluxside` to impose pressure or boundary conditions, respectively.

In the moving frame script we first define saturation and influx rate on the faces. The functions are stored in fluid properties for permeability, viscosity and permeability. The grid structure knows the faces and the cells. Then after putting the elements together it gives the influxes and the outfluxes in the boundary cells.

```
        muW = fluid.muW(p);
        mobW = krW./muW;
        muO = fluid.BOxmuO(p).*fluid.bO(p);
        mobO = krO./muO;
if param.set_outflux
      bc.influxes = [G.faces.areas(faces{1})*param.influx_rate*...
                                        [mobO(1),mobW(1)]/(mobO(1) +...
                                mobW(1)); -G.faces.areas(faces{2})*...
        param.influx_rate*[mobO(2), mobW(2)]/(mobO(2) + mobW(2))];
```

```
else
    bc.influxes = G.faces.areas(faces{1})*param.influx_rate*...
                                                [mobO,mobW]/(mobO + mobW);
end
```

This can be considered as sources in the cells on the boundary. In the script with Cartesian coordinates, we impose boundary/source conditions on the incoming boundary.

```
src = addSource(src,cells,fluxes_in,'sat',sat_in);
```

In both scripts the cell- and faceindeces must be computed manually. Source conditions are easier to work with, however usually the boundary conditions are given.

Initial state

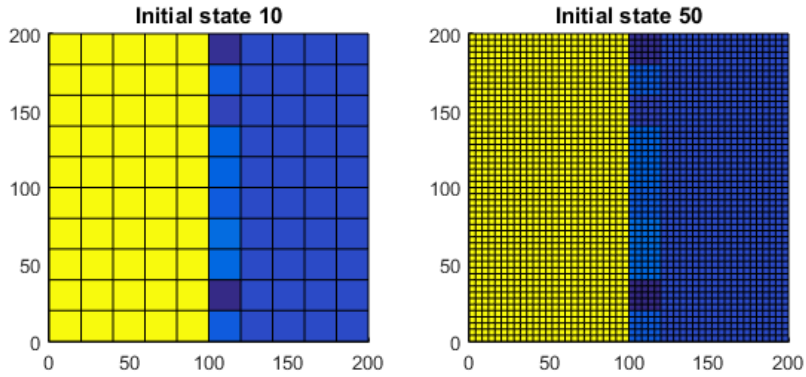


Figure 2.9: Initial states with diffent mesh refinements

To compare objectively different mesh refinements or moving coordinates versus the steady system, we need to start with the same initial conditions for water saturation. Therefore, we generated a random perturbation for the 10x10 grid, which will be used for all the other refinements and systems (See the initial state in Figure).

First, we used this idea, but the only difference that we copied the perturbation pattern for one column only. This is still not capturing the same saturation conditions and makes the comparison unreliable. Other approaches emerged, such as to start with finer meshes and take the average of the perturbation for smaller meshes. The averages can neutralize the perturbation and the comparison does not make sense anymore.

The advantages of the moving frame

In the first example (See figure 2.5) the fingering is noticeable but it does not give back the true nature of the phenomenon. Because the boundary and the initial conditions were different, we simulated the first example with similar conditions. The figures did not show much difference from the presented example, so we compare the moving frame to the above presented figures.

The apparent speed of the frame creates a diffusive effect, thus it masks the fingering. As we can see in Figure 2.10 the appearance of the fingering is very well observable on finer meshes.

Mesh size and capillary pressure

In general, when we are trying to determine the optimal mesh size, it would be ideal that the numerical error would be negligible compared to any characteristic force present in the viscous fingering.

It happens very often that the numerical error which generates the diffusion will suppresses the capillary pressure. We would like to avoid this to happen.

We have already introduced capillary pressure curves in Figure 1.6. In the numerical script we use a simple water saturation capillary pressure table which will be extended to our domain by interpolation. The shape of the function is very similar to the above presented analytical function.

So far the magnitude of the pressure was not important since the driving forces are generated by fluxes and sources. Although, capillary pressure is very sensitive this setup and we have to adjust these quantities accordingly.

Simulations and results

The physical domain in our simulation was a 200x200 oil/water reservoir. Capillary pressure is set to the same magnitude as base pressure. At the oil-water boundary it has a significant effect on the fingering.

In order to show how inappropriate mesh size can can make the capillary pressure's effect disappear, we first used a very big mesh element and used 10 elements over the axes. In Figure 2.10 the simulations in the upper part one can observe that there is literally no different between the two figures.

This must be incorrect and as we can see below that on finer meshes the capillary effect becomes more and more visible. To make sure that the numerical error is negligible we refine the mesh further and analyze the formations.

Shielding, spreading and splitting

George M. Homsy describes the viscous fingering in his article [5] throughout the processes shielding, spreading and splitting. We would like to first show some experimental results and then compare our numerical simulations to these phenomenomas.

We have briefly mentioned the Hele-Shaw cell 2.2 where we have

$$\begin{aligned} \nabla \cdot u &= 0, \\ \nabla p &= -12\mu \overrightarrow{u}/b^2 + \rho g. \end{aligned}$$

This configuration allows us to experiment and to check whether our numerical schemes gives good results.

As we can observe in Figure 2.11 if we have a perturbation (e.g. the water has already penetrated slightly in the oil) then as it was expected the finger is going to spread due to

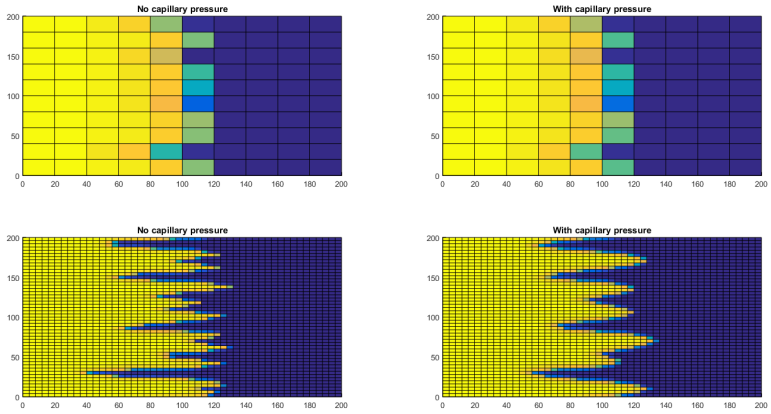


Figure 2.10: Simulations on different mesh sizes with and without capillary pressure

the viscosity differences. When it reached a critical point and the capillary pressure is getting bigger than the pressure driving the liquid (water), it splits and we have two less sharp fingers. The shielding is simply that bigger fingers are spreading faster and fill the space in front of the small fingers.

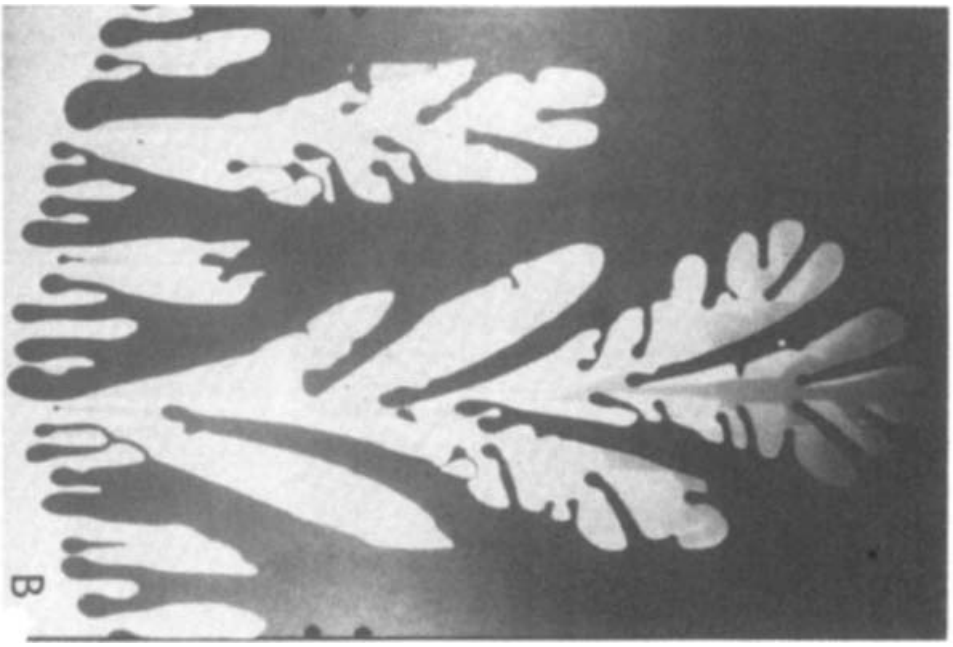
Now let's take a look at the numerical simulation. However, the spreading is difficult to see, but the bigger fingers are certainly moving faster. In addition, we can observe the the effect of the capillary pressure is more and more visible as we refine our mesh. In Figure 2.10 the last plot shows the simulation on a 50x50 grid and if we compare that to Figure 2.12, we can track the splitting.

In the last figure we feel comfortable about that the important properties of the phenomenon are captured because it is apparently converging. On the other hand, on this mesh size (above 150x150) one step takes significantly more time and even the convergence is getting slower and we need to cut the time step at some steps in order to converge.

This suggests us to find more adequate high order methods. Before we do that, we test the convergence for different perturbations.
In the first set of simulations, we generated 10 random numbers on a $[0, 1]$, we subtracted 0.5 from them and we scaled them up to the desired magnitude of the perturbation. Then we just simply copied these values for the finer meshes, keeping the saturation values the same for the physical domain.

In the second method we generated a multivariate normal distribution by a python code. In the script we generated a covariance matrix with a correlation paramater γ. When γ was typically small the correlation was very strong and the perturbation values were small. Then we used slightly larger γ and tested our meshes. See in Figure 2.13.

We can observe as we have the proper mesh size (i.e. the numerical error is negligible) the number of fingers is converging to 10-11 fingers (depends on how we count the very small fingers which are becoming part of a bigger finger). In addition, one can see that the very sharp edges disappear and length of the fingers can't exceed a certain limit and when



[5]

Figure 2.11: Flooding the Hele-Shaw cell

we are watching the progression we get the idea that could be a relationship between the number of fingers and the length. This was discussed in 2.4.

2.4 Growth and wavenumber of the fingering

In [14], Yortsos and Hickernell derive important relations between the rate of growth and the wavenumber of the disturbance. We summarize and provide some additional visualization of their results.

2.4.1 The base state

[14]

The base state is the solution of the model when we inject flow (e.g. water) with a constant pressure into an oil filled reservoir. It is represented by the same continuum model which we described in Section 1.1.2 and every quantities will be understood in terms of REV (see Section 1.1.1).

The governing equations are Darcys equation, conservation of mass and momentum for the phases, capillary pressure and saturation equation. The mass conservation equation in general form is

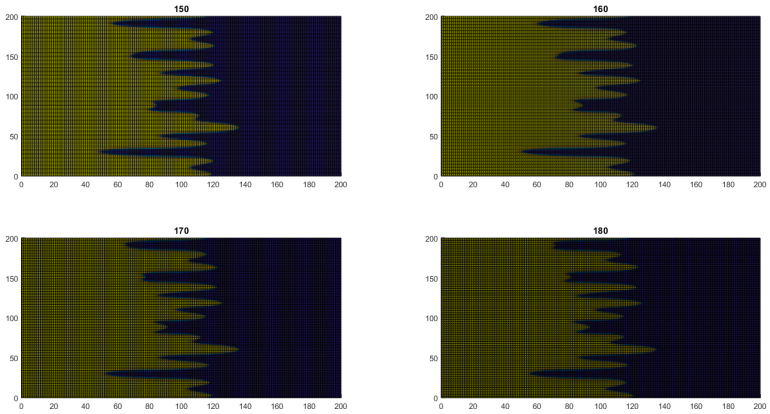


Figure 2.12: Different mesh sizes for the moving frame

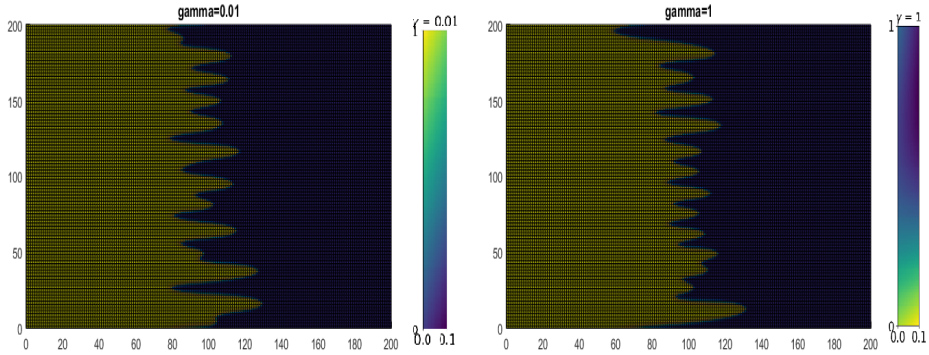


Figure 2.13: Different covariance matrix

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot \overrightarrow{u}_w = 0. \tag{2.34}$$

We are not going to indicate that the flux is a vector function from now on. Moreover, they assume that the capillary function is a function of saturation only and it takes the form of

$$P_c = \gamma_{ow} \cos\theta \frac{\sqrt{\phi}}{\sqrt{k}} J(S_w). \tag{2.35}$$

Capillary pressure, relative permeabilities and fractional flow function were introduced in the previous chapters. We are going to refer to Figures 1.6, 1.7 and 1.8. As we described before the saturation satisfies a parabolic equation,

$$\phi \frac{\partial S_w}{\partial t} + u_T \frac{df_w}{dS_w} \cdot \frac{\partial S_w}{\partial X} = \frac{\partial}{\partial X}\Big(f_w \frac{k k_{ro}}{\eta_o} \cdot \frac{dP_c}{dS_w} \cdot \frac{\partial S_w}{\partial X} \Big). \tag{2.36}$$

Where initial saturation $S_w = S_{wi}$ at $T = 0$, $S_w = S_{wi}$ at $X = 0$ (boundary saturation) and $f_w \frac{kk_{ro}}{\eta_o} \cdot \frac{dP_c}{dS_w} \cdot \frac{\partial S_w}{\partial X} = f_{w0}$.

We analyzed the shape of the fractional flow function before Section 1.2.2 and in Figure 1.8 one can see a typical S shape function for it. However, for a strongly wet system, empirical results suggest that the fractional flow is strongly convex.

If we neglect capillary effects, write the equation

$$\phi \frac{\partial S_w}{\partial T} + u_T \frac{df_w}{dS_W} \frac{\partial S_w}{\partial X} = 0, \tag{2.37}$$

and we assume strong convexity of the fractional flow function, the solution will develop shocks. It is in line with the results that we discussed for the B-L case in 1.3, since the rarefaction part disappears because there is no inflection point and the second derivative is always positive. Similarly, it can happen that only rarefaction waves appear, in the case where the fractional flow is strictly concave.

The authors analyze the results in a moving coordinate system with a speed of

$$v = \frac{u_T}{\phi} \frac{(f_w^{-\infty} - f_w^{\infty})}{(S_w^{-\infty} - S_w^{\infty})}, \tag{2.38}$$

and the moving coordinate $\xi = x - vt$. They introduced the following normalized and scaled variables,

$$\lambda_w(S) = \frac{k_{rw}}{k_{rw}^{-\infty}}, \qquad \lambda_o = \frac{k_{ro}}{k_{ro}^{\infty}}, \qquad M = \frac{k_{rw}^{-\infty}\eta_o}{k_{ro}^{\infty}\eta_w},$$
$$\lambda_T(S) = M\lambda_w + \lambda_o, \qquad \lambda_c(S) = \frac{\sqrt{k}}{\sqrt{\phi}} \cdot \frac{1}{\gamma \cos\theta \cdot \frac{dP_c}{dS_w}} < 0.$$

Using these normalizations, we have that $f_w = \frac{M\lambda_w}{\lambda_T}$. Furthermore, they introduce dimensionless time, length and capillary pressure by multiplying with $\phi L(S_w^{-\infty} - S_w^{\infty})/u_T$,$L$ and $u_T\eta_o Lkk_{ro}^{\infty}$ respectively. The length is defined as $L = \sqrt{\phi k}(S_w^{-\infty} - S_w^{\infty})k_{rw}^{-\infty}/N_{Ca}$ where $N_{Ca} = u_T\eta_w/\gamma_{ow}\cos\theta$ is the local capillary number of the displacement.

We have already discussed that, by solving the equation via the method of characteristics, we obtain a travelling wave solution

$$S(x,t) = S(x - vt). \tag{2.39}$$

So we use this information and let $\xi = x - vt$. Then we integrate (2.36) and use the chain rule that $\frac{dS(\xi)}{dt} = \frac{dS}{d\xi} \cdot \frac{\partial \xi}{\partial t}$ and we define the reduced saturation

$$S = \frac{S_w - S_w^{\infty}}{S_w^{-\infty} - S_w^{\infty}}. \tag{2.40}$$

Due to scaling and normalization, they obtain the following simplifications:

$$\lambda_o\lambda_w\lambda_c\frac{dS}{d\xi} = \lambda_T(vS + f_w^{\infty}) - M\lambda_w, \tag{2.41a}$$

$$\lambda_o\frac{dP}{d\xi} = vS + f_w^{\infty} - 1, \tag{2.41b}$$

$$v = f_w^{-\infty} - f_w^{\infty}. \tag{2.41c}$$

If the solutions exist, the right hand side of (2.41) must be positive, but this is satisfied, because of the identity $\lambda_T(vS + f_w^{\infty}) - M\lambda_w = \lambda_T(vS - (f_w^{\infty} - f_w^{-\infty}))$. In case of miscible flow ($\lambda_c = 0$), the base states admits any solution, provided that the right hand side of (2.41) vanishes identically [14]. Additional results can be found in the article [2].

Phase phane linear analysis for ODE

One possible way to analyze the base state if we convert it to an ODE system. Let's consider

$$\ddot{y} = f(y, \dot{y}). \tag{2.42}$$

We rewrite it as a system and set

$$u = y,$$
$$v = \dot{y}.$$

Then, we get the first-order system ode,

$$\dot{u} = v,$$
$$\dot{v} = f(u, v).$$

The traveling wave equation is a second order scalar ode, with unknown S and parameter ξ.

The saturation in a one-dimensional model is the solution of

$$s_t + a(s_x, s) + b(s)s_{xx} = 0, \tag{2.43}$$

where $a : \mathbb{R}^2 \to \mathbb{R}$ and $b : \mathbb{R} \to \mathbb{R}$, with $b > 0$. The steady state solution is therefore given by

$$a(s_x, s) + b(s)s_{xx} = 0, \tag{2.44}$$

which is a second order scalar ode.
When we discretize, we replace $s(t, x) = 0$ by a $S(t) \in \mathbb{R}^N$ and (2.43) becomes

$$\frac{dS}{dt} = F(S) \tag{2.45}$$

where $F : \mathbb{R}^N \to \mathbb{R}^N$. The steady state of the discretized system (2.45) is the solution of

$$F(S) = 0.$$

In fact, the equation for the base state (2.44) can be integrated once. Indeed, the complete expression is

$$-v\frac{\partial s}{\partial \xi} - \frac{\partial}{\partial \xi}(\lambda_w(s)K(\frac{\partial p}{\partial \xi} - p_c'(s)\frac{\partial s}{\partial \xi})) = 0, \tag{2.46}$$

$$-\frac{\partial}{\partial \xi}(\lambda_T(s)K\frac{\partial p}{\partial \xi} - \lambda_w p_c'(s)K\frac{\partial s}{\partial \xi}) = 0. \tag{2.47}$$

see (2.74a) and (2.76) below. Hence,

$$\lambda_T(s)K\frac{\partial p}{\partial \xi} - \lambda_w(s)p_c'(s)K\frac{\partial s}{\partial \xi} = u_c,$$

for some constant u_c. From (2.46), we get

$$\frac{\partial}{\partial \xi}(vs + \frac{\lambda_w}{\lambda_T}u - \frac{\lambda_w\lambda_o}{\lambda_T}Kp_c'(s)\frac{\partial s}{\partial \xi}) = 0,$$

which we can integrate. We obtain

$$vs + \frac{\lambda_w}{\lambda_T}u - \frac{\lambda_w\lambda_o}{\lambda_T}Kp_c'(s)\frac{\partial s}{\partial \xi} = C$$

for some constant C. This is a first order ODE. The constant C can be determined by using that $\lim_{\xi\to\infty} s(\xi) = s_r$ and $\lim_{\xi\to\infty} s_\xi(\xi) = 0$ so that we end up with

$$vs + \frac{\lambda_w}{\lambda_T}u - \frac{\lambda_w\lambda_o}{\lambda_T}Kp_c'(s)\frac{\partial s}{\partial \xi} = vs_r + \frac{\lambda_w(s_r)}{\lambda_T(s_r)}u.$$

Since we also have $\lim_{\xi\to-\infty} s(\xi) = s_l$ and $\lim_{\xi\to-\infty} s_\xi(\xi) = 0$, it follows that

$$vs_l + \frac{\lambda_w(s_l)}{\lambda_T(s_l)}u = vs_r + \frac{\lambda_w(s_r)}{\lambda_T(s_r)}u$$

and therefore

$$v = \frac{f_w(s_l) - f_w(s_r)}{s_l - s_r}u,$$

which corresponds with to the Rankine-Hugoniot shock condition.

Simulation of the base state

We have simulated the base state without and with capillary pressure in the moving coordinate system in one dimension. We have given a detailed explanation of the moving frame in Section 2.3. In the Descartes-coordinates the difference is only that the wave(shock) is moving with the shock speed v.

As we expected a pure shock appears on the first figure and the capillary smoothens the graphs in the middle where we have the jump in saturation.

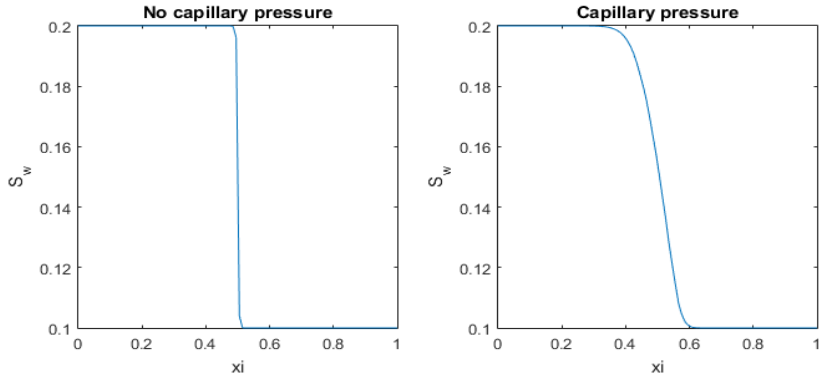


Figure 2.14: On the left we can observe a pure shock, the right hand shows the simulation with capillary pressure (with exaggregated capillary pressure)

$$-v\frac{\partial S_\alpha}{\partial \xi} + \tilde{\nabla}\cdot(\lambda_o \tilde{\nabla} P) = 0 \tag{2.48}$$

The result can be seen in Figure 2.15. If we compare the results with the one dimensional Figure 2.14, we can see that the solution is invariant in the y direction. This is of course expected as the consequence of the symmetric initial state.

We are going to analyze the stability of this state and compare it to the article [14] by Yortsos and Hickernell. First we review the framework in more details.

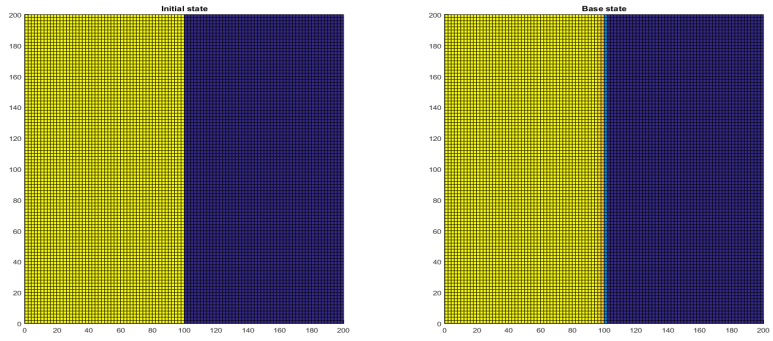


Figure 2.15: The corresponding base state to a symmetrical initial condition in two dimensions

2.4.2 Eigenmode analysis

Standard framework

We have a state variable and a system of partial differential equations which describes the time evolution of that,

$$u_t + Au = 0 \tag{2.49}$$

or

$$u_t t + Au = 0 \tag{2.50}$$

where A is a linear differential operator involving only the spatial derivatives. The heat equations and the acoustic wave equation could be examples respectively, for the first and the second. If we take

$$u_{tt} - \frac{1}{c^2}\Delta p = 0, \tag{2.51}$$

then the eigenmodes denote the eigenvectors of A. For a given eigenvalue λ with eigenmode u_λ, after this we get

$$u(t,x) = e^{-\lambda} u_\lambda(x) \tag{2.52}$$

is a solution to (2.49). Typically, the operator A is self-adjoint, so $\lambda \geq 0$. Then we obtain oscillatory solutions to (2.51) of the form

$$u(t,x) = e^{iwt} u_\lambda(x), \tag{2.53}$$

where $w = \sqrt{\lambda}$. The eigenvalues of the operator depend on the domain.

Discretization in the standard framework

We discretize the equation in space and $u(t,x) \in \mathbb{R}$ becomes $u(t) \in \mathbb{R}^N$. A typical discretization of (2.49) is

$$Mu_t + Au = 0. \tag{2.54}$$

Note here that A in (2.49) is a continuous operator, while A in (2.54) corresponds the discretization of the previous. The matrix M is often called the mass matrix. For a given mass matrix M, λ is a generalized eigenvalue if there exists a $u \neq 0$ such that

$$Au = \lambda M u. \tag{2.55}$$

The vector u is called a generalized eigenvector. Let u_λ be a generalized eigenvector of A with respect to M and we denote by λ the corresponding eigenvalue. Then, let us insert $u = f(t)u_\lambda$ in (2.54), we get

$$(f'(t) + \lambda f(t))Mu_\lambda = 0, \tag{2.56}$$

which implies that $u = e^{-\lambda}u_\lambda$ is a solution. Here it is crucial that the M must be invertible, and it is even better if it is a symmetric positive definite matrix.

Before we go any further and analyze the eigenmodes of the different formulations, we introduce the concept of a small perturbation and linearize it around the base state.

2.4.3 The eigenvalue problem and the dispersion relation

Probably the most important result of the article is the relationship between the rate of growth and the wavenumber. We have to underline here that this subsection is mostly based on [14] with extraction and of the parts which were crucial were our latter analyses.

First they write the equations in a dimensionless notation:

$$\begin{aligned} \frac{\partial S}{\partial t} - v\frac{\partial S}{\partial \xi} + \tilde{\nabla}\cdot(\lambda_o \tilde{\nabla} P) = 0, \\ \tilde{\nabla}\cdot(\lambda_T \tilde{\nabla} P - \lambda_w \lambda_c \tilde{\nabla} S)\frac{\partial P}{\partial \xi} = 0, \\ \tilde{\nabla} = \Big(\frac{\partial}{\partial \xi}, \frac{\partial}{\partial y}\Big). \end{aligned} \tag{2.57}$$

The linear stability of the state in the transverse direction is being analyzed. The solution of the base state is denoted $(\bar{S}, \bar{P})$. So we seek the solution in the form of

$$\begin{aligned} S(x,y,t) = \bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y + \omega t}, \\ P(x,y,t) = \bar{P}(\xi) + \epsilon p(\xi) e^{i\alpha y + \omega t}, \end{aligned} \tag{2.58}$$

where $\epsilon p << \bar{P}$ and $\epsilon s << \bar{S}$. We write back this into the equations and we linearize around the base state. This results in an eigenvalue problem for p, s with a single variable ξ. Substitution into the equation yields

$$\begin{aligned} \frac{\partial}{\partial t}(\bar{S} + \epsilon s(\xi) e^{i\alpha y+\omega t}) - v\frac{\partial}{\partial \xi}(\bar{S} + \epsilon s(\xi) e^{i\alpha y+\omega t}) + \\ \tilde{\nabla}\cdot(\lambda_o(\bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y+\omega t})\tilde{\nabla}(\bar{P}(\xi) + \epsilon p(\xi) e^{i\alpha y+\omega t})) = 0, \\ \tilde{\nabla}\cdot(\lambda_T(\bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y+\omega t})\tilde{\nabla}(\bar{P}(\xi) + \epsilon p(\xi) e^{i\alpha y+\omega t}) \\ -\lambda_w(\bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y+\omega t})\lambda_c(\bar{S}(\xi) \\ +\epsilon s(\xi) e^{i\alpha y+\omega t})\tilde{\nabla}(\bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y+\omega t})) = 0. \end{aligned} \tag{2.59}$$

We linearize around the base state with

$$\lambda_w(\bar{S}(\xi) + \epsilon s(\xi) e^{i\alpha y+\omega t}) = \lambda_w(\bar{S}) + \epsilon \lambda_w^{'}(\bar{S}) p(\xi) e^{i\alpha y+\omega t} \tag{2.60}$$

After calculations, we obtain

$$\begin{aligned} \omega\epsilon e^{i\alpha y+\omega t} s(\xi) - v\frac{d\bar{S}}{d\xi} + v\epsilon e^{i\alpha y+\omega t}\frac{ds}{d\xi} + \frac{\partial}{\partial \xi}(\lambda_o(\bar{S})\frac{d\bar{P}}{d\xi}) + \\ \epsilon e^{i\alpha y+\omega t}\Big(\frac{\partial}{\partial \xi}(\lambda_o^{'}(\bar{S}) s\frac{d\bar{P}}{d\xi}) + \frac{\partial}{\partial \xi}(\lambda_o(\bar{S})\frac{dp}{d\xi}) - \alpha^2 p\Big) = 0, \\ \frac{\partial}{\partial \xi}(\lambda_T(\bar{S})\frac{d\bar{P}}{d\xi}) + \epsilon e^{i\alpha y+\omega t}\Big(\frac{\partial}{\partial \xi}(\lambda_T^{'}(\bar{S}) s\frac{d\bar{P}}{d\xi}) + \frac{\partial}{\partial \xi}(\lambda_T(\bar{S})\frac{dp}{d\xi}) - \alpha^2 p(\xi)\Big) \\ \frac{\partial}{\partial \xi}(\lambda_w(\bar{S})\lambda_c(\bar{S})\frac{d\bar{P}}{d\xi}) + \epsilon e^{i\alpha y+\omega t}\Big(\frac{\partial}{\partial \xi}(\lambda_w^{'}(\bar{S}) s\lambda_c(\bar{S})\frac{d\bar{S}}{d\xi}) \\ + \frac{\partial}{\partial \xi}(\lambda_w(\bar{S})\lambda_c^{'}(\bar{S}) s\frac{d\bar{S}}{d\xi}) + \frac{\partial}{\partial \xi}(\lambda_w(\bar{S})\lambda_c(\bar{S})\frac{ds}{d\xi}) - s\alpha^2\lambda_w\lambda_c\Big) = 0. \end{aligned} \tag{2.61}$$

Since the base state $\bar{S}, \bar{P}$ is a particular solution of the equation, we have only

$$\begin{aligned}
\omega s + v\frac{ds}{d\xi} + \frac{\partial}{\partial\xi}(\lambda_o^{'}(\bar{S})s\frac{d\bar{P}}{d\xi}) + \frac{\partial}{\partial\xi}(\lambda_o(\bar{S})\frac{dp}{d\xi}) - \alpha^2 p = 0,\\
\frac{\partial}{\partial\xi}(\lambda_T^{'}(\bar{S})s\frac{d\bar{P}}{d\xi}) + \frac{\partial}{\partial\xi}(\lambda_T(\bar{S})\frac{dp}{d\xi}) - \alpha^2 p\\
+\frac{\partial}{\partial\xi}(\lambda_w^{'}(\bar{S})s\lambda_c(\bar{S})\frac{d\bar{S}}{d\xi})\\
+\frac{\partial}{\partial\xi}(\lambda_w(\bar{S})\lambda_c^{'}(\bar{S})s\frac{d\bar{S}}{d\xi}) + \frac{\partial}{\partial\xi}(\lambda_w(\bar{S})\lambda_c(\bar{S})\frac{ds}{d\xi}) - \alpha^2\lambda_w(\bar{S})\lambda_c(\bar{S})s = 0.
\end{aligned} \tag{2.62}$$

where s and p are only a function of ξ. To simplify the problem into a second order system, Yortsos and Hickernell introduced new functions,

$$\begin{aligned}
\Phi = s/\Big(\frac{d\bar{S}}{d\xi}\Big),\\
\Psi = p - (v\bar{S} + f_w^{\infty} - 1)s/\Big[\lambda_o\frac{d\bar{S}}{d\xi}\Big].
\end{aligned} \tag{2.63}$$

After using the base equations from (2.41) the equations yield

$$\begin{aligned}
\frac{d}{d\xi}\Big(\lambda_o\frac{d\Psi}{d\xi}\Big) + \frac{d}{d\xi}\Big(\zeta\frac{d\Phi}{d\xi}\Big) + \frac{\omega}{v}\frac{d\zeta}{d\xi}\Phi - \alpha^2\lambda_o\Psi - \alpha^2\zeta\Phi = 0,\\
\frac{d}{d\xi}\Big(\alpha_T\frac{d\Psi}{d\xi}\Big) - \frac{d^2\Phi}{d\xi^2} - \alpha^2\lambda_T\Psi + \alpha^2\Phi = 0,\\
\zeta = vS + f_w^{\infty} - 1,
\end{aligned} \tag{2.64}$$

where the effects of hysteresis and anisotropy are absent. The system can be simplified into a single fourth order equation with

$$\begin{aligned}
\mu = \zeta\lambda_T + \lambda_o \geq 0,\\
\frac{dW}{d\xi} = \lambda_T\Psi + \Phi,
\end{aligned} \tag{2.65}$$

definitions. Then we can write Ψ and Φ as functions of W only,

$$\begin{aligned}
\Phi = \frac{W'' - \alpha^2 W}{\lambda_T^{'}}\\
\Phi = \frac{\lambda_T}{\lambda_T^{'}}[W'' - \alpha^2 W] - W'.
\end{aligned} \tag{2.66}$$

Then we substitute (2.66) into (2.64) and we obtain

$$\begin{aligned}
\frac{d}{d\xi}\Big[\mu\frac{d}{d\xi}\Big(\frac{W'' - \alpha^2 W}{\lambda_T^{'}}\Big)\Big] - \alpha^2\frac{\mu}{\lambda_T^{'}}\big[W'' - \alpha^2 W\big]\\
= -\frac{\omega}{v}\frac{d\zeta}{d\xi}\Big[\frac{\lambda_T}{\lambda_T^{'}}(W'' - \alpha^2 W) - W' - \frac{\alpha^2 v}{\omega}W\Big].
\end{aligned} \tag{2.67}$$

However, in order to derive bounds on ω, they stick to the previous formulation with Ψ and Φ. After some rearrangements of (2.66) and (2.64)

$$\frac{d}{d\xi}\Big[\frac{\mu}{\lambda_T}\frac{d\Phi}{d\xi}\Big] + \alpha^2\frac{d}{d\xi}\Big[\frac{\mu}{\lambda_T}\Big]W - \frac{\alpha^2\mu}{\lambda_T}\Phi + \frac{\omega}{\upsilon}\frac{d\zeta}{d\xi}\Phi - \alpha^2\frac{d\zeta}{d\xi}W = 0, \tag{2.68a}$$

$$\frac{d}{d\xi}\Big[\frac{1}{\lambda_T}\frac{dW}{d\xi}\Big] - \frac{\alpha^2 W}{\lambda_T} - \frac{\lambda_T^{'}}{\lambda_T}\Phi = 0. \tag{2.68b}$$

From (2.68a) ω can be expressed:

$$\frac{\omega}{\upsilon} = \Big\{\frac{\mu}{\lambda_T}\Phi^{''} - \Big(\frac{\mu}{\lambda_T}\Big)^{'}\Phi^{'} + \alpha^2\frac{\mu}{\lambda_T}\Phi + \alpha^2\Big[\zeta^{'} - \Big(\frac{\mu}{\lambda_T}\Big)\Big]W\Big\}\zeta^{'}\Phi. \tag{2.69}$$

Let ξ_{max} is the place where $|\Phi|$ attains its maximum value $\|\Phi\|_\infty$. It can be said without loss of generality that $\Phi(\xi_{max}) > 0$, because if not, then we can replace Φ with $-\Phi$, since the equations are linear. Then, $\Phi'(\xi_{max}) =$, $\Phi''(\xi_{max}) < 0$ and (2.69) yields

$$\omega = \Big(\frac{\upsilon\mu}{\zeta'\lambda_T}\Big)\Big|_{\xi=\xi_{max}}\Big[\frac{-\Phi''(\xi_{max})}{\|\Phi\|_\infty} + \alpha^2\Big] + \upsilon\alpha^2\Big[1 - \frac{\mu/\lambda_T}{\zeta'}\Big]\Big|_{\xi=\xi_{max}}\frac{W(\xi_{max})}{\|\Phi\|_\infty}. \tag{2.70}$$

Let A_1, A_2 and A_3 positive numbers such that

$$\begin{aligned} \frac{\upsilon\mu}{\lambda_T^{'}\zeta} &\le A_1, \\ \Big|\upsilon\Big(1 - \frac{1}{\zeta'}\Big(\frac{\mu}{\lambda_T}\Big)^{'}\Big)\Big| &\le A_2, \\ \frac{|W|}{\|\Phi\|_\infty} &\le A_3. \end{aligned} \tag{2.71}$$

After further calculations, we obtain the relationship in figure (2.16). This is very important that the ω can't exceed the maximum value $\frac{A_2^2A_3^2}{4A_1}$ and it attains it the point $\frac{A_2A_3}{2A_1}$.

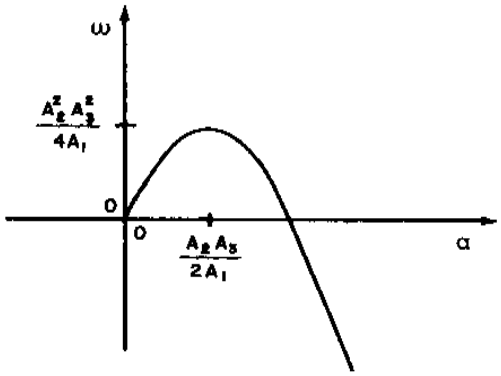
2.4.4 Eigenmodes in different formulations

In the previous subsection 2.4.3 one could see that it is a tedious work to linearize the system analytically. This can be easily obtained from our numerical structure because the saturations and pressure values are stored as *ADI variables*. Thus, the Jacobian of the base state is known numerically. Although it is not straightforward to find the eigenvalues of the system and to prove the instability of the base state.

In our numerical simulations, we have used

$$\frac{\partial s_\alpha}{\partial t} - v\frac{\partial s_\alpha}{\partial \xi} - \nabla\cdot(\lambda(s_\alpha)K\nabla p_\alpha = 0, \tag{2.72}$$

for $\alpha = \{w, o\}$, and



[14]

Figure 2.16: Schematic of upper bound for the rate of growth

$$s_w + s_o = 1 \quad p_o - p_w = p(s_w). \tag{2.73}$$

In this form there are four unknowns s_α and p_α and four equations. This is reduced to the two variables s_w and p_o, which we denote s and p, respectively. The governing equations are then

$$\frac{\partial s}{\partial t} - v\frac{\partial s}{\partial \xi} - \nabla \cdot (\lambda_w(s)K(\nabla p - p_c{}'(s)\nabla s)) = 0, \tag{2.74a}$$

$$-\frac{\partial s}{\partial t} + v\frac{\partial s}{\partial \xi} - \nabla \cdot (\lambda_o(s)K\nabla p) = 0. \tag{2.74b}$$

After the linearization, we obtain

$$\begin{pmatrix} 1 & 0 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} s \\ p \end{pmatrix}_t + B \begin{pmatrix} s \\ p \end{pmatrix} = 0. \tag{2.75}$$

Then we have

$$M = \begin{pmatrix} 1 & 0 \\ -1 & 0, \end{pmatrix}$$

,i.e. the matrix is singular, which means that we can not use the standard framework because it is not invertible. In order to solve this problem, we sum up the equations in (2.74) and obtain the so-called pressure equation,

$$\nabla \cdot (\lambda_T(s)K(\nabla p - p_c{}'(s)\nabla s)) = 0. \tag{2.76}$$

Here we linearize around the base state again, i.e. we are using the Jacobians of the equations which is stored. This would require the same steps as in the previous subsection and we would arrive to something similar as in (2.62) in the saturation or fractional flow formulation. So the pressure equation is taking the form

$$A_{ps}s + A_{pp}p = 0, \tag{2.77}$$

and for the two linear differential operators A_{ps} and A_{pp} the system is

$$\begin{aligned} s_t + A_{ss}s + A_{sp}p &= 0, \\ A_{ps}s + A_{pp}p &= 0. \end{aligned} \tag{2.78}$$

where again A_{ss} and A_{sp} are two linear operators.
It turns out that the operator A_{pp} is invertible and we can express the pressure as

$$p = -A_{pp}^{-1}A_{ps}s. \tag{2.79}$$

Plugging this into the linearized version of (2.74a), we get the equation

$$s_t + (A_{ss} - -A_{pp}^{-1}A_{ps})s = 0, \tag{2.80}$$

and we are back to the standard framework, see 2.4.2, we have

$$s_t + \hat{A}s = 0 \tag{2.81}$$

for

$$\hat{A} = A_{ss} - A_{sp}A_{pp}^{-1}A_{ps}. \tag{2.82}$$

We have to compute the eigenvalues of $\hat{A}$,

$$\hat{A}s = \lambda s. \tag{2.83}$$

In this notation, the solution of the system will be

$$\begin{array}{rcccc} s_1 = & C_1 \cdot e^{-\lambda_1} \cdot v_{11} + & \dots & +C_N \cdot e^{-\lambda_N} \cdot v_{N1}, \\ \vdots = & & \ddots & \\ s_N = & C_1 \cdot e^{-\lambda_1} \cdot v_{1N} + & \dots & +C_N \cdot e^{-\lambda_N} \cdot v_{NN}, \end{array}$$

where

$$v_i = \begin{Bmatrix} v_{i1} \\ \vdots \\ v_{iN} \end{Bmatrix}$$

is the i-th eigenvectors of $\tilde{A}$, λ_i is the corresponding eigenvalue and C_i are the constants which can be determined from the initial and the boundary conditions.

In order to prove the instability of the base state, there must be at least one negative (real part) eigenvalue. Since this will be the dominating solutions as time evolves, we need the corresponding eigenvector to find the frequency, i.e. the number of fingers.

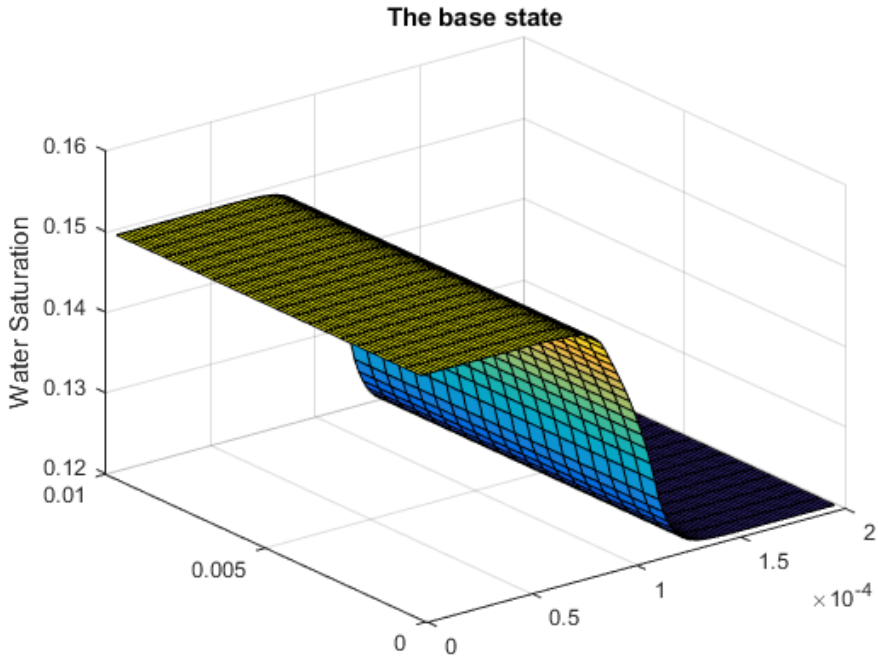


Figure 2.17: 3D representation of the base state

Codes and results

The first problem which we faced was to create the base state. We converged into some smooth state where the shape was dependent on the size of the capillary pressure. However the state has not converged to 0.2 saturation on the left side, instead we get between 1.95 and 1.9. This could be fixed by setting an extremely large physical domain, e.g. 10000 meters.

Although the process is fairly straightforward and we have all the necessary information stored in the Jacobian matrices we face annoying computational difficulties. We would like to use a big mesh size in order to work with sufficiently small matrices. On the other hand, if we don't use an appropriate mesh size, the numerical error is too big and could eventually lead to inaccurate simulations (discussed in Section 2.3.4).

On the other hand, numerical inaccuracy can ironically give us a hint on the behavior of the system. If we use a 150x150 mesh with physical properties 200m (could be any number which is not a precise multiple of the mesh) in both directions, then Matlab rounds the rational numbers which creates a very small inaccuracy in the division. Even this small perturbation is enough to create fingers. This is certainly not a precise proof of the instability of the system but it is a good practical example for that.

Therefore first we have chosen 100x100 mesh size in order to avoid these pitfalls and to get sufficiently fine mesh to reach convergence and capture the true nature of the fingering.

This has taken still quite long time and the results were not conclusive. Then we went down to 50x50 grid and we obtained results more in line with the theoretical results. See the eigenvalues and the eigenvector in Figure 2.18

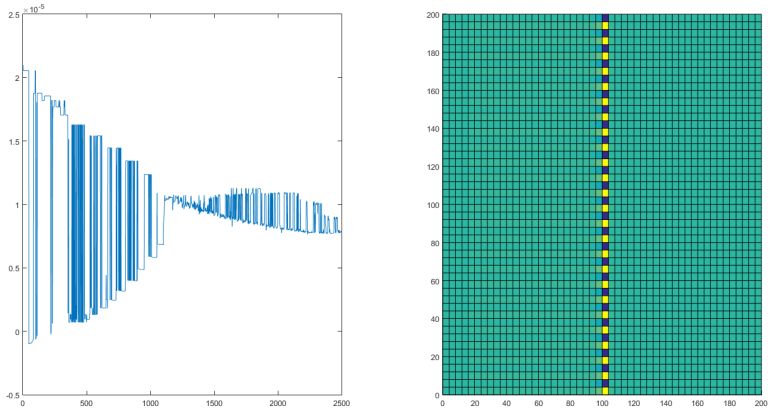


Figure 2.18: Eigenvalues and the corresponding eigenvector to the minimum

As one can see we have negative eigenvalues (actually one would be enough) and we have the corresponding eigenvector which shows which cells are going to grow faster, e.i. they will dominate as time goes forward. Certainly this is valid only in a small environment of the base state where the first derivatives (Jacobian) dominate. However, after long time it won't grow unboundedly because of the capillary pressure.

In figure 2.19 we plotted the number of the fingers (x-axis) in different eigenmodes and their growth rate (y-axis). This is very similar with 2.16 but the largest eigenvalue should not have the largest number of fingers.
We improved this version by using averages of the saturation on the faces. However, this posed new difficulties in the convergence. Then first we simulated the base state with upwind to get a smooth function and after that we ran the simulation again with averages on the faces. This gave much better results. For better visualization purposes, we used a 3-dimensional plot to show the dominating eigenmode (see in Figure 2.20).
If we examine the number of fingers now, we finally get the expected results in Figure 2.21. We multiplied the negative eigenvalues with -1 in order to get a similar plot as 2.16.

Then in the latest version of the code we applied some additional steps. We recommend to everyone that every version should be tested with many incoming parameters both for the physical properties (such as capillary pressure and physical length) and the grid parameters (cell numbers). It can often happen that some major effects can be visible in certain setups and not in others. This requires quite a lot of experimenting.

We first compute the base state in one dimension. Even this seemingly easy process requires several steps. From the symmetrical initial state the base state is computed by using hll fix and upwind scheme on the faces. After this preliminary state was achieved. We change the scheme to face averages from upwind. We apply this order because the

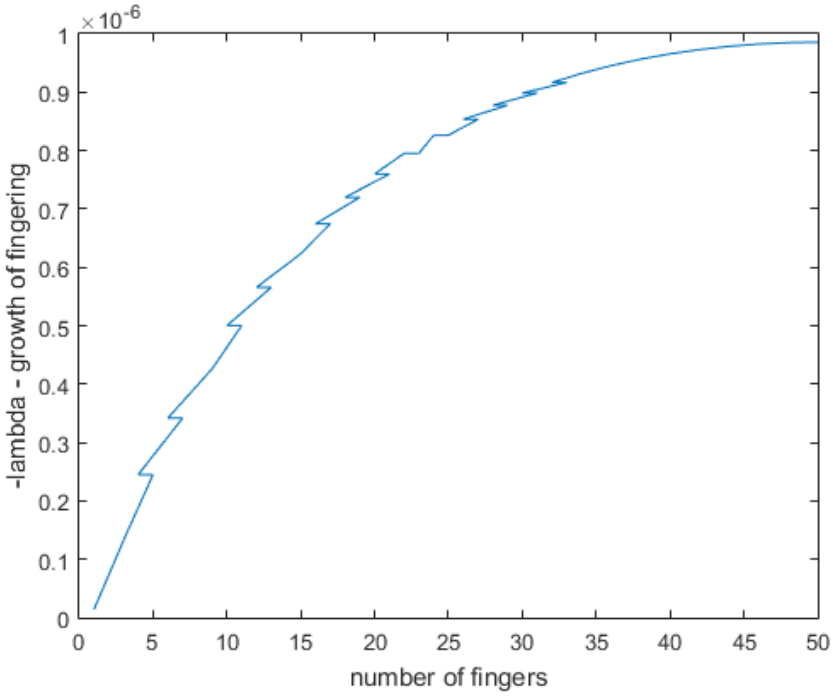


Figure 2.19: Number of fingers in the different modes

initial state is not smooth enough and face averages don't converge to a solution.

We have calculated the source terms at the end of the one dimensional solution. That should be constant on all the faces, since this is a steady state and there is mass conservation in the cells. Then we simply extend this one dimensional solution to two dimensions, see in Figure 2.17. We run the two dimensional simulation with this state and the precalculated source terms for some quite long time in order to check whether the base state is stable.

Finally, we compute the eigenvalues and plot the first eigenmode, i.e. the corresponding eigenvector to the biggest eigenvalue pair (convex conjugates). We obtain convex eigenvalue pairs as we expected before. We summarized the eigenmodes of the two largest convex conjugates and the result can be seen in Figure 2.22.

Capillary pressure stabilizes the fingering

Capillary pressure has a very important effect as we described in previous sections. If we choose an extremely large capillary multiplier, then even the base state becomes and stable equilibrium point (the smallest eigenvalue will be still positive).

After sufficiently long time and on a fine mesh we can't exceed a maximum length and a number of fingers. It is not necessarily that we end up in an equilibrium state which is stable, because it could lead some periodical behavior where the wavenumber and the

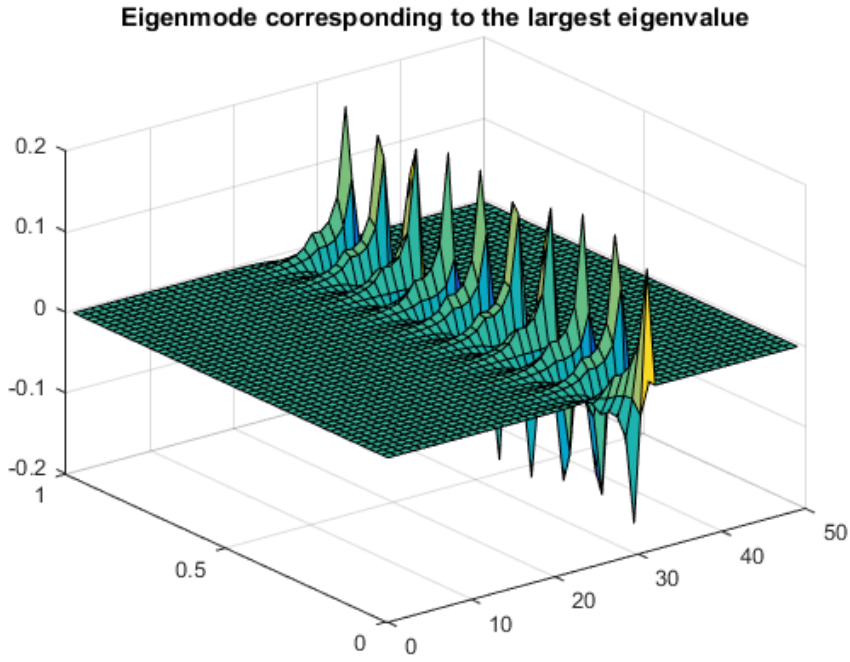


Figure 2.20: Number of fingers in different modes

length are limited in any state.

2.4.5 Dimension splitting

Because of the computational difficulties we tried to find different methods to deal with the problem. Dimension splitting is very similar process to separation of variables where we use functions dependent on one space variable only. Similarly in dimension splitting we make a decomposition of the operator A by space dimensions x and y. Let $u = (s, p)^T$. The eigenvalue problem takes the form of finding λ such that there exists $u \neq 0$ and

$$\mathcal{A}u = \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix} u, \tag{2.84}$$

where

$$\mathcal{A} = \begin{pmatrix} \mathcal{A}_{ss} & \mathcal{A}_{sp} \\ \mathcal{A}_{ps} & \mathcal{A}_{pp} \end{pmatrix}.$$

The operator $\mathcal{A}$ can be split in two $\mathcal{A} = \mathcal{A}_\xi + A_y \Delta_y$ where $\mathcal{A}_\xi$ contains all the derivatives with respect to ξ, $\mathcal{A}_y$ functions of ξ and

$$\Delta_y = \begin{pmatrix} \frac{\partial^2}{\partial y^2} & 0 \\ 0 & \frac{\partial^2}{\partial y^2} \end{pmatrix}.$$

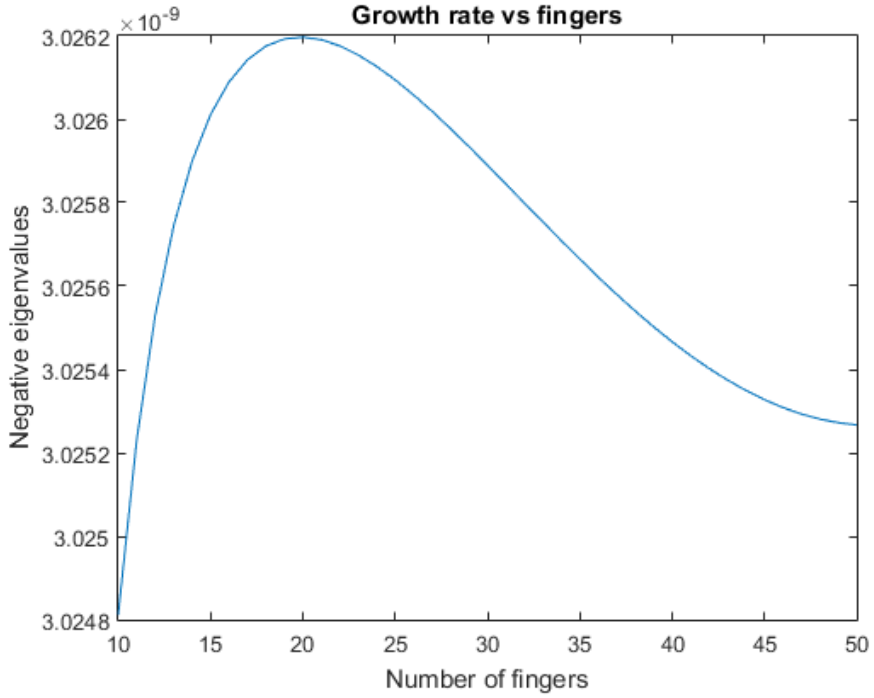


Figure 2.21: The dominating eigenmode

We can rewrite (2.84) as

$$\mathcal{A}_\xi u + \mathcal{A}_y \Delta_y u = \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix} u. \tag{2.85}$$

The operator $\mathcal{A}_y$ has the form

$$\mathcal{A}_y = k \begin{pmatrix} p_s'(s) & -\lambda_w(s) \\ p_s'(s) & -\lambda_T(s) \end{pmatrix}$$

The operators Δ_y commute with $\mathcal{A}_\xi$ and $\mathcal{A}_y$, that is,

$$\mathcal{A}_\xi \Delta_y = \Delta_y \mathcal{A}_\xi,$$

and correspondingly for $\mathcal{A}_y$. There exists a classical result about joint diagonalization of linear operators that commute: If $\mathcal{A}$ and $\mathcal{B}$ commute and are diagonalizable then they can be diagonalized in the *same* basis of eigenvectors. This result relies on the fact that an eigenvector space for, say A, which we denote E_λ is stable by B, meaning that $B(E_\lambda) \subset E_\lambda$. Indeed, let $u \in E_\lambda$, we have

$$\mathcal{A}u = \lambda u,$$

which implies

$$\mathcal{B}\mathcal{A}u = \lambda B u.$$

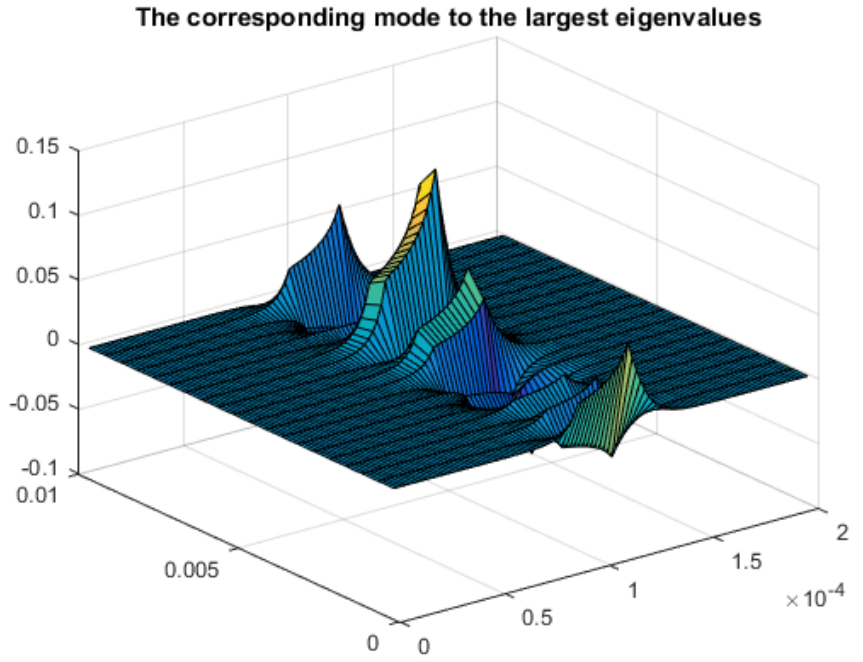


Figure 2.22: Eigenmodes to the fastest growth rate

Since $\mathcal{B}\mathcal{A} = \mathcal{A}\mathcal{B}$, we end up with

$$\mathcal{A}\mathcal{B}u = \lambda \mathcal{B}u,$$

which means that $Bu \in E_\lambda$. The eigenvalues of Δ_y are easy to compute and are equal to $u(\xi, y) = e^{i\omega y}u(\xi)$. Indeed, for such u, $\Delta_y u = -\omega^2 u$. Let us denote by E_λ, the set of u which satisfies (2.85). For any $u \in E_\lambda$, we have

$$\Delta_y(A_\xi u + A_y\Delta_y)u = \Delta_y \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix} u,$$

which implies, using the commutation properties of the operators,

$$(A_\xi u + A_y\Delta_y)\Delta_y u = \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix} \Delta_y u.$$

Hence, $\Delta_y u \in E_\lambda$ and the space E_λ is stable with respect to the operator Δ_y. We can therefore find a decomposition of E_λ with respect to the eigenvector of Δ_y, something of the form $u(\xi)e^{i\omega y}$.

Conclusion

We have gone through a very interesting and challenging process by analyzing viscous fingering with Matlab. Even on the way to get to know the physical and the mathematical model we have faced various obstacles. The highly nonlinear nature of the differential equations and the strong coupling between them are inspiring us to improve numerical schemes to find better and more precise solutions.

The numerical scheme and model proved to be very effective and give immediate results but as we can see later the apparent speed of the frame is covering some very important natures of the phenomena.

We could easily overcome this issue by using a moving frame, but there was still lots to investigate. We determined the optimal mesh sizes which don't hide the capillary effects and we showed that additional mesh refinement does not improve the results.

Although the simulation gave us confidence that our results are convergent and we can trust them, the analysis has created more challenges. So far we had to invert a big matrix only once in the calculation and multiply with the previous state (vector). Suddenly we were working multiple very big matrices and the running times have grown substantially. This forced us to improve our codes at other points and led to the final base state.

We are really happy that we managed to show the instability of the system and we numerically supported some known theoretical results between the number of fingers and the growth rate.

We planned to move on with dimension splitting which could make the process faster and more effective. Therefore, we provided some theoretical background and we hope that we can continue this work on viscous fingering in a Ph.d. thesis.

Bibliography

[1] Capillary pressure. https://en.wikipedia.org/wiki/Capillary_pressure. Accessed: 2017-10-30.

[2] E. Chikhliwala, Y. Yortsos, et al. Investigations on viscous fingering by linear and weakly nonlinear stability analysis. *SPE reservoir engineering*, 3(04):1–268, 1988.

[3] Grid one cell. http://www.xmswiki.com/xms/GMS:3D_Grid_Module. Accessed: 2017-10-30.

[4] Hele-shaw schematic. https://en.wikipedia.org/wiki/Hele-Shaw_flow. Accessed: 2017-10-30.

[5] G. M. Homsy. Viscous fingering in porous media. *Annual review of fluid mechanics*, 19(1):271–311, 1987.

[6] K.-A. Lie. An introduction to reservoir simulation using matlab: user guide for the matlab reservoir simulation toolbox (mrst). sintef ict, 2014.

[7] R. D. Neidinger. Introduction to automatic differentiation and matlab object-oriented programming. *SIAM review*, 52(3):545–563, 2010.

[8] Capillary pressure. http://petrowiki.org/Capillary_pressure. Accessed: 2017-10-30.

[9] Relative permeability. http://perminc.com/resources/fundamentals-of-fluid-flow-in-porous-media/chapter-2-the-porous-medium/relative-permeability/relative-permeability-curves/. Accessed: 2017-10-30.

[10] Multiphase in porous media. http://www.epgeology.com/reservoir-engineering-f10/wettability-oil-and-gas-reservoirs-t5978.html. Accessed: 2017-10-30.

[11] Viscous fingering. http://news.mit.edu/2011/fluid-mixing-0519. Accessed: 2017-10-30.

[12] Buckley-levertt case. https://www.researchgate.net. Accessed: 2017-10-30.

[13] Wettability in oil and gas reservoirs. http://www.epgeology.com/reservoir-engineering-f10/wettability-oil-and-gas-reservoirs-t5978.html. Accessed: 2017-10-24.

[14] Y. Yortsos, A. Huang, et al. Linear-stability analysis of immiscible displacement: Part 1-simple basic flow profiles. *SPE Reservoir Engineering*, 1(04):378–390, 1986.