Modeling and simulation of EOR due to biofilm formation and gas production

by
Andrea Tronstad Lønn

Master of Science in
Applied and Computational Mathematics

Department of Mathematics
University of Bergen

June 2019

Acknowledgments

First of all I want to thank my supervisors Professor Florin A. Radu and David Landa-Marbán for helping me with this Thesis. I am especially thankful to you David for answering all my questions and for helping me with the code as much as you have. Thank you Florin for suggesting this topic and for all your help and hints throughout this project.

Thank you to Gunhild Bødtker, senior researcher at NORCE and leader of the IMMENS project, for inviting me to the workshop at Solstrand. I also want to thank Bartek Florczyk Vik, senior researcher at NORCE, for showing interest in my project.

Next, I would like to thank my family for always supporting me and believing in me. I also want to thank you for proofreading my Thesis.

Last but not least, I want to thank all my friends at the institute. Thank you for always making me smile and for all the fun we have had together. All in all, thank you for making the last five years great.

Andrea Tronstad Lønn
June 2019

Abstract

Microbial enhanced oil recovery (MEOR) is a collective term for more environmentally friendly enhanced oil recovery (EOR) techniques. EOR methods are techniques used after primary and secondary stage of oil recovery aiming to increase the amount of oil recovered from the reservoir.

In this thesis we look at MEOR methods, more specifically bioclogging using biofilm. We have simulated a two-phase system containing of oil, water, nutrients, biofilm and gas in MATLAB using the MRST package from SINTEF. The gas in our system is produced by the biofilm, which is assumed to dissolve in the oil phase; therefore we modeled it with a transport equation. We assumed that the nutrients are dissolved only in the water phase and therefore we simulate the nutrients in a similar way as the gas. The two phases, water and oil, are simulated by mass conservation equations and the biofilm is modeled by the equation for reversible deposited bacteria proposed by Kim (2006). The implementation of the model has been tested using an analytical solution and a benchmark example.

In addition to making our model, we have studied how the gas produced from the biofilm can affect the oil viscosity and, as a result, affect the amount of oil being produced. We have also studied how the positioning of the biofilm can affect the oil production. Last but not least, we have looked at how the combination of strategic positioning of the biofilm and viscosity reduction due to gas production will affect the oil recovery.

Our results show that the amount of oil recovered by using biofilm to preform bioclogging will increase if we model the oil viscosity as a function of the gas produced. The oil recovery will also increase if the biofilm is located at the beginning of the core instead of evenly distributed in every cell. The best result is obtained when we combine these two scenarios, in other words, when the biofilm is located at the beginning of the domain and the oil viscosity is modeled as a function of the gas produced in the system.

Contents

List of Figures

List of Tables

Notation

- **Properties**

ρ : Density

p : Pressure

ϕ : Porosity

S_α : Saturation of phase α

θ : Water content

$\boldsymbol{k}$: Permeability

μ_α : Dynamic viscosity of phase α

$\mu_{o,min}$: Minimum oil viscosity

$\boldsymbol{u}_\alpha$: volumetric flow rate per area

$\boldsymbol{\kappa}$: Hydraulic conductivity

h : Hydraulic head

g : Magnitude of gravity

Ω : Volume

t : Time period

$\boldsymbol{k}_{r,\alpha}$: Relative permeability

p_c : Capillary pressure

$\boldsymbol{\lambda}_\alpha$: Phase mobility

S_α^{res} : Residual saturation

$\vec{v}_\alpha$: Velocity of phase α

C_n : Concentration of nutrients

C_g : Concentration of gas

C_b : Concentration of bacteria

D_n^{eff} : Effective diffusion coefficient for nutrients

D_g^{eff} : Effective diffusion coefficient for gas

R_n : Reaction term for the nutrients

R_g : Reaction term for the gas

$\mu_{g\ \max}$: Maximum specific gas production

$\mu_{b\ \max}$: Maximum specific biofilm production

Y_g : Yield coefficient for the gas

Y_b : Yield coefficient for the biofilm

$K_{b/n}$: Half-saturation constant for biofilm

σ : Volumetric fraction of bacteria attached totally

σ_1 : Volumetric fraction of bacteria attached reversibly

σ_2 : Volumetric fraction attached irreversibly

k_1 : Reversible attachment rate coefficient

k_2 : Irreversible attachment coefficient

g_1 : Biofilm growth coefficient

d_1 : Bacteria decay rate coefficient

C : Constant from the absolute permeability model

p_e : Entry capillary pressure

λ : Pore size distribution index

$k'_{r,\alpha}$: Endpoint relative permeability of phase α

n_α : Corey exponent for phase α

- **Subscripts**

α : Phase index

n : Nonwetting phase

w : Wetting phase

b : Bacteria

Abbreviations

EOR - Enhanced oil recovery

MEOR - Microbial enhanced oil recovery

REV - Representative elementary volume

IFT - Interfacial tension

AD - Automatic differentiation

MRST - MATLAB reservoir simulation toolbox

TPFA - Two-point flux approximation

BC - Boundary condition

Chapter 1

Introduction

Oil is, and has been since the 20th century, the preferred energy source in the world (Ekt Interactive, 2010). When opening an oil reservoir, 10-40% of the oil is produced with primary and secondary oil recovery methods (Sen, 2008). This means 60-90% of the oil is still in the reservoir after the conventional recovery methods have been used. Scientists have introduced enhanced oil recovery (EOR) methods in order to recover more of the oil. Microbial enhanced oil recovery (MEOR) is one of the promising EOR techniques.

MEOR is a collective term for methods where the intention is to try and modify the structure and/or the microbial environments in an oil reservoir in order to improve the oil recovery (Wood, 2019). The biggest goal for MEOR is to recover the oil trapped in mature reservoirs more cost efficient, both when it comes to money and the environment, than other EOR methods. MEOR has been a topic since Beckman identified the behavior of bacteria on mineral oil in 1926 but it was not until 1946 that the laboratory testing started with the work of Zobell (1946). The first field test was completed in 1954 in Lisbon oil field Arkansas (Wood, 2019). Since then, many experiments have been conducted, both in the laboratories and on a field-scale. Hundreds of patents have been granted related to MEOR (Wood, 2019). Even though a lot of research has been done, the results have been inconsistent going from laboratory experiments to field trials, therefore the methods have still not been approved for commercialization.

In this thesis we look at one of the methods within MEOR, namely bioclogging using biofilm. A biofilm is a mixture containing of biomass and water that clog pore throats and, by doing so, redirect the fluid to alternative channels in the porous medium (Wood, 2019). The redirection of the fluids will hopefully lead to more oil being produced.

We model a two-phase system with water and oil. The system also contains of nutrients, biofilm and gas. To do our simulations we use MATLAB Reservoir Simulation Toolbox (MRST). We use mass conservation equations to model the water and oil in our system and transport equations to model the nutrients, gas and biofilm. After making our model, we tested it against a numerical example from Kim (2006).

The aim of this work is to see how the oil production in a reservoir can be improved by the use of biofilm and gas. We want to see how gas produced from the biofilm can affect the oil viscosity and hopefully lead to more oil recovery. In addition to this, we also examined how the positioning of the biofilm originally in the reservoir will affect the oil production.

The thesis is structured in the following way:

Chapter 2: This chapter gives an introduction to porous media and oil recovery. Here we present the properties necessary to describe flow in porous media, Darcy's law and the mass conservation equation. In addition, we introduce oil recovery, EOR and finally MEOR.

Chapter 3: Next, we present the equations used to describe our problem and all the parameters and relationships used in our model on a general level.

Chapter 4: After introducing the equations, we build up our model step by step and give an overview of how we have implemented everything in MATLAB using MRST. At the end of the chapter we present the values for the parameters used and consider a test example for our model.

Chapter 5: To check that we have implemented everything correctly we consider a numerical experiment for different time steps and spatial distributions in 1D and 2D. We also consider a benchmark example. Then we look at how gas produced from the biofilm can affect the oil viscosity and, as a result, also effect the oil production.

Chapter 6: In Chapter 6 we conclude our work and give our final thoughts. We also discuss possible further work.

Chapter 2

Porous media and oil recovery

2.1 Porous media

A porous medium is a material containing pores and a solid skeleton. Figure 2.1 shows an example of the structure of a porous medium in two dimensions. There can be a single-phase flow through the porous medium, meaning the pores are only filled with one fluid (e.g water or oil), or we can have a multi-phase flow. In a multi-phase flow there are two or more fluids flowing through the pore space. Two-phase flow where there is water and oil going through the medium is the most common multi-phase flow.

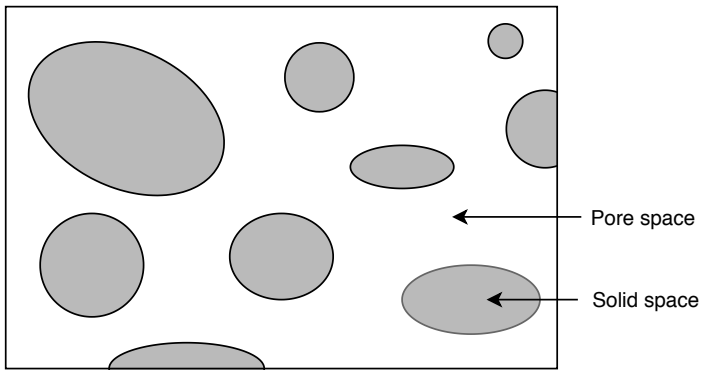


Figure 2.1: Illustration of a porous medium.

In order to describe flow in porous media there are some properties that need to be defined, the first one being the REV. Ideally we want to study a porous medium on a micro level, but because of the geometric complexity this is very hard to do so we choose a representative elementary volume (REV) that we study instead. The REV has to be sufficiently large and contain a big number of pores so that we can define a mean global property, for example the porosity, but still be small enough so that the fluctuation from one pore to another is negligible. In addition to this, the REV has to be small enough so that the variation of properties across the REV can be described by continuous functions (Nordbotten and Celia, 2012).

Another property of the porous medium is porosity. The porosity tells us how much of the REV is void space. It is denoted by ϕ and given by

$$\phi = \frac{\text{Volume of voids in the REV}}{\text{Volume of REV}}.$$

Porosity is a dimensionless quantity that lays in the interval between 0 and 1.

The next property we need to introduce is permeability, $\mathbf{k}$. Permeability measures how easy a fluid flow through the porous rock (Pettersen, 1990). When there is only one fluid flowing through the rock, we talk about absolute permeability. In a multi-phase flow we talk about relative permeability. This will be introduced properly later in the thesis when we introduce the properties of a two-phase flow in Section 2.4.

Having discussed some of the properties for the medium, we now move on to some of the properties of the fluid, the first one being density. The density of the fluid describes the ratio between the mass of the fluid and the volume of the pore space occupied by the fluid (Bear, 1988), i.e.

$$\rho = \frac{\text{Mass of the fluid}}{\text{Volume of the fluid}}.$$

Another property of the fluid is viscosity, μ. Viscosity describes the fluids resistance to flow (Bear, 1988). High viscosity means high resistance to flow.

Compressibility is a measurement of the changes of the volume when the tension on the fluid changes. In this thesis we will consider incompressible fluids. This means that the volume of the fluid will not change when the tension changes.

2.2 Darcy's law

In 1856, Henry Darcy introduced one of the most important building blocks used to describe flow in porous media, namely Darcy's law (Nordbotten and Celia, 2012). Today, Darcy's law is most known on the form

$$\mathbf{u} = -\boldsymbol{\kappa}\nabla h. \tag{2.1}$$

Here $\mathbf{u}$ is the volumetric flow rate per area, $\boldsymbol{\kappa}$ is the hydraulic conductivity and h is the hydraulic head.

Hydraulic conductivity $\boldsymbol{\kappa}$ is a coefficient of proportionality and it is given by

$$\boldsymbol{\kappa} = \frac{\mathbf{k}\rho g}{\mu}. \tag{2.2}$$

It describes how easy the fluid in the system can flow through the material. Hydraulic conductivity is a function of both the porous medium and the fluid flowing through it. In Equation (2.2), μ is the dynamic viscosity of the fluid, ρ is the density of the fluid, $\mathbf{k}$ is the permeability and g is the magnitude of gravity.

The hydraulic head can be expressed as

$$h = \frac{p}{\rho g} + z, \tag{2.3}$$

where p is the pressure and z is the elevation from the datum to the point with pressure p. The hydraulic head describes the distance from the datum (the point where z=0) to the top of the fluid in the system (Nordbotten and Celia, 2012).

Substituting Equation (2.2) and (2.3) into Equation (2.1) gives

$$\mathbf{u} = -\frac{\mathbf{k}}{\mu}(\nabla p + \rho g \nabla z) = -\frac{\mathbf{k}}{\mu}(\nabla p + \rho g \mathbf{e_z}) = -\frac{\mathbf{k}}{\mu}(\nabla p - \rho \mathbf{g}). \tag{2.4}$$

Here we denote $\nabla z = \mathbf{e_z}$ and $\mathbf{g} = -g\mathbf{e_z}$. The negative sign comes from the fact that the vertical coordinate has a positive direction upwards. This is Darcy's law for

single-phase flow.

2.3 Mass conservation

Darcy's law is not enough to analyze general flow problems, we also need the equation of mass conservation. The law of mass conservation states that mass cannot be destroyed or created from chemical reactions.

The idea for the equation of mass conservation is that the change of mass in a particular volume Ω over a given time period t, is equivalent to the net mass flow into the volume through the boundaries plus the mass that appears, or disappears, from within the region without crossing the boundary (Nordbotten and Celia, 2012).

$$\int_{\Omega} \frac{\partial m}{\partial t} dV = -\int_{\partial\Omega} \mathbf{f} \cdot \boldsymbol{\nu}_n dA + \int_{\Omega} \mathbf{r} dV. \tag{2.5}$$

In Equation (2.5) we have

$$m = \rho\phi, \quad \mathbf{f} = \rho\mathbf{u}, \quad \mathbf{r} = \psi. \tag{2.6}$$

Here m represents the mass per volume of the porous medium, $\mathbf{f}$ is the mass flux vector and $\mathbf{r}$ is the sinks/sources within the volume. $\boldsymbol{\nu}_n$ represents the outer normal vector. If $\mathbf{f}$ and $\mathbf{r}$ is equal to zero, m is constant in time.

Using Gauss theorem we get the relationship

$$\int_{\Omega} \left(\frac{\partial \rho\phi}{\partial t} + \nabla \cdot \rho\mathbf{u} - \psi \right) dV = 0. \tag{2.7}$$

Equation (2.7) must hold for any arbitrary volume Ω. Because of this, it is enough to consider the integrand. We now obtain the differential equation for conservation

of mass:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot \rho \mathbf{u} = \psi. \tag{2.8}$$

2.4 Two-phase flow

A two-phase flow system has two phases present in the pore space. The two phases create fluid-fluid interfaces at the pore scale, allowing the two fluids to coexist in the pore space (Nordbotten and Celia, 2012).

The fluids are referred to as the wetting fluid and the nonwetting fluid, where the fluid with stronger surface attraction is the wetting fluid. The angle where the fluid-fluid intersection meets the solid material is called the contact angle. The fluid on the side of the interface with an angle less than 90 degrees with respect to the solid is the wetting fluid (Nordbotten and Celia, 2012). Figure 2.2 illustrates the relationship between the wetting phase, nonwetting phase and the contact angle.

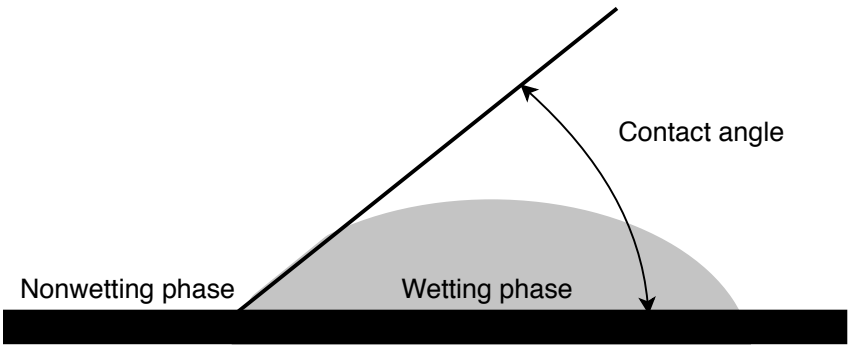


Figure 2.2: The figure simulates the relationship between the nonwetting phase, wetting phase and the contact angle. Figure inspired from PetroWiki (2016) and Skiftestad (2015).

In order to describe two phase-flow, it is necessary to introduce some more properties. The first one is saturation. The saturation explains how much of the void the different phases occupy when the porous medium is filled with two immiscible fluids

(liquids or gases) (Bear, 1988). The saturation is given by (Bear, 1988)

$$S_\alpha = \frac{\text{Volume of fluid } \alpha \text{ within the REV}}{\text{Volume of voids within the REV}}. \tag{2.9}$$

The value of the saturation with respect to a particular fluid will lie between 0 and 1. The fact that the two fluids together fill the void space in the porous medium implies that $S_n + S_w = 1$. Here n represent the nonwetting phase and w the wetting phase.

Relative permeability is another quantity that needs to be introduced in order to describe two-phase flow. In a two-phase system the two fluids block some of the pore space for each other. This reduces the space available and it makes it harder for the fluids to flow (Nordbotten and Celia, 2012). To account for the reduction in permeability due to the presence of another fluid, relative permeability is introduced. Relative permeability is a function of saturation and it is denoted by $\boldsymbol{k}_{r,\alpha} = \boldsymbol{k}_{r,\alpha}(S_\alpha)$.

After introducing relative permeability, we can write Darcy's law for a multiphase flow as (Nordbotten and Celia, 2012):

$$\boldsymbol{u}_\alpha = -\frac{\boldsymbol{k}_{r,\alpha}\boldsymbol{k}}{\boldsymbol{\mu}_\alpha}(\nabla p_\alpha + \rho_\alpha \boldsymbol{g}). \tag{2.10}$$

It is enough to extend Darcy's law with relative permeability in order to model two-phase flow. This is because we consider the pore space blocked by the other fluid as being solid (Nordbotten and Celia, 2012).

The last thing we have to introduce in order to complete the extension of Darcy's law is phase mobility $\boldsymbol{\lambda_\alpha}$. Phase mobility describes the relationship between relative permeability and phase viscosity, i.e $\boldsymbol{\lambda}_\alpha(S_\alpha) = \boldsymbol{k}_{r,\alpha}/\boldsymbol{\mu}_\alpha$ (Nordbotten and Celia, 2012). Darcy's Law can now be expressed as

$$\boldsymbol{u}_\alpha = -\boldsymbol{\lambda_\alpha}\boldsymbol{k}(\nabla p_\alpha + \rho_\alpha \boldsymbol{g}). \tag{2.11}$$

Capillary pressure is the difference in phase pressure between the wetting and non-

wetting phase

$$p_c = p_n - p_w. \tag{2.12}$$

The capillary pressure and the saturation of a system are related but the relationship is not unique. It depends on the history of the saturation for the system (Dullien, 1979). Such behavior is called hysteresis. Figure 2.3 shows the relationship between wetting fluid saturation and capillary pressure.

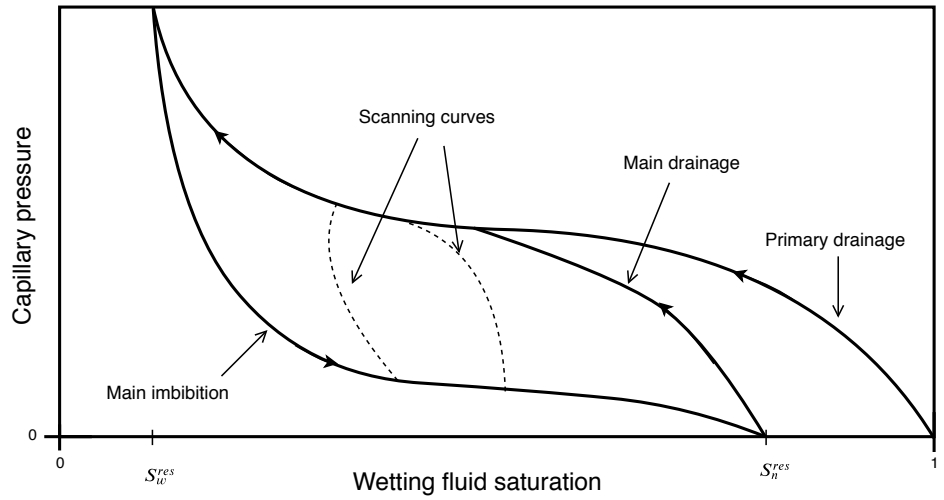


Figure 2.3: Illustration of the relationship between wetting fluid saturation and capillary pressure. Figure inspired by Nordbotten and Celia (2012).

Let us describe Figure 2.3. Imagine you have a porous medium only consisting of wetting fluid. There is no flow through the top and bottom of the domain. On one side of the reservoir you have a reservoir consisting of only wetting fluid and on the other side there is a reservoir filled with nonwetting fluid. We assume we can control the pressure in the two reservoirs. Now we increase the pressure in the nonwetting fluid and measure how much of the wetting fluid is displaced. After the system has reached equilibrium, we get a data point relating the capillary pressure and the nonwetting water saturation. If we now repeat the experiment but increase the pressure in the wetting phase instead, we get data points relating capillary pressure and the wetting fluid saturation (Nordbotten and Celia, 2012). The process when nonwetting fluid displaces wetting fluid is called drainage. Imbibition is when

wetting fluid displaces nonwetting fluid. After the drainage process there will still be wetting fluid in the reservoir. The amount of wetting fluid left in the reservoir is referred to as residual saturation and is denoted as S_w^{res}. This value can be seen in Figure 2.3. S_n^{res} denotes the residual saturation for the nonwetting phase. This is also the reason behind why there is a primary drainage curve and a main drainage curve. Primary drainage is without residual saturation in the system. This will in practice only happen at the beginning of the experiment. If you change between drainage and imbibition before reaching the residual saturation, you will get the scanning curves as seen in Figure 2.3.

2.5 Oil recovery

Oil recovery plays a big part in the world economy. The world relies heavily on petroleum as a primary energy source and as a resource for several other products (Patel et al., 2015). Because of the heavy use of oil, we need to find methods to produce more oil from the reservoirs we already have. This will save time and reduce the cost of oil production.

We divide oil recovery into three main stages. The first one is primary recovery and this stage produces 5-10% of the total reserve. During the second stage we produce between 10% to 40% of the oil in the reservoir (Sen, 2008). In other words, 60-90% of the oil in the reservoir has to be recovered in a third stage.

Primary recovery produces oil and gas using the natural pressure drive in the reservoir present when the reservoir is opened for the first time (Sen, 2008). Secondary recovery is when we inject fluids into the wells to increase the pressure in the reservoir, forcing as much of the oil as possible out through the production wells. The remaining oil is difficult to produce because it is located in regions of the oil reservoir that is hard to access. The oil is also held in the pores by capillary pressure (Sen, 2008). In order to produce the remaining oil in the reservoirs, scientists have created enhanced oil recovery technologies, also known as EOR technologies.

2.6 EOR methods

Enhanced oil recovery (EOR) is a process where chemicals such as surfactants, polymers and acids, amongst others, are used together with secondary oil recovery methods to produce more of the crude oil in the reservoir (Patel et al., 2015).

EOR methods make it possible to produce more oil from the reservoirs but it is not without a cost. The methods have both a high economic cost and a high cost for the environment.

The chemical processes used to produce the EOR products involve toxic chemicals. In addition to this, some of the products themselves are damaging to the environment, especially when presented with oil (Patel et al., 2015). This is why MEOR was introduced, to reduce the environmental and financial costs.

2.7 MEOR

Microbial enhanced oil recovery (MEOR) was first introduced by Beckman in 1926 but it was not until the work of ZoBell in 1946 (Zobell, 1946) that MEOR was given serious consideration (Brown, 2010). MEOR is a tertiary method of oil recovery and it is a collective name for many different methods. If you compare some of the EOR methods to certain MEOR methods, you will see that the only difference between them is how the chemicals are introduced into the reservoir. However, MEOR generally refers to the use of microorganisms, instead of chemicals, in the oil-bearing formation to enhance oil recovery (Brown, 2010).

2.7.1 Implementation

There are three general strategies for the implementation of MEOR (Youssef et al., 2009).

Injection of nutrients to stimulate indigenous microbes: If there are microorganisms naturally present in the reservoir that can perform the desired function

(e.g. plugging, gas or biosurfactant production), they can be stimulated by injecting nutrients (Youssef et al., 2009). Before choosing this method it is necessary to determine what microorganisms are present in the reservoir. Only then it is possible to choose the suitable method to stimulate the microbes naturally present in the reservoir, in order to get the wanted effect (Youssef et al., 2009).

Injection of external microbes and nutrients: If the appropriate microorganisms are not present in the reservoir the strategy is to inject both the microbes and the nutrients. For this strategy to work sufficiently the injected microorganisms must be able to grow in the environmental conditions present in the reservoir (Youssef et al., 2009). The injected microbes also have to be able to grow in the presence of the indigenous microbes.

Injection of ex situ-produced products: If there are no microorganisms suitable for the desired outcome present in the reservoir and the environment is too harsh for exogenous microbes to survive, the last resort is to inject ex situ-produced products (Youssef et al., 2009). Ex situ-produced products are products constructed outside of the reservoir. A major concern with this method is loss of bioproduct during the transport inside the reservoir due to adsorption. On the other hand, efficient amounts of biosurfactants and polymers can be produced from cheap renewable sources without extensive purification (Youssef et al., 2009).

2.7.2 Mechanisms

There are different bioproducts that can be used in MEOR. The products influence the reservoir behavior and oil mobilization differently. The possible bioproducts are surfactants, acids, gases, solvents, biomass and polymers (Sen, 2008). Surfactants can reduce the interfacial tension present in the oil-water interface and the oil-rock interface. The use of acids can improve the permeability and porosity by dissolving parts of the rock. This can lead to reduction of the entrapped oil (Safdel et al., 2017). Biomass changes the wettability and makes it possible to perform selective plugging (Sen, 2008). By preforming selective plugging you channel the floodwater towards the available oil (Safdel et al., 2017). Polymers are used to increase viscosity

of water-flood. The use of gas will increase the pressure in the reservoir. The gas will also effect the interfacial tension and we will have viscosity reduction. Using solvents will increase the permeability through the porous network and re-pressurize the reservoir (Sen, 2008).

2.7.3 Advantages and disadvantages

Even though MEOR has been a research topic since Beckman introduced it in 1926 (Brown, 2010), it is still only used on trial fields.

MEOR methods have many advantages but also disadvantages. These are well summed up in Safdel et al. (2017) and in Wood (2019). Here is a rundown of the advantages and disadvantages presented in the two articles:

Advantages:

- Economically efficient.
- Low injection cost of microbes and nutrients.
- Low expenses and complexity of facilities set up.
- Low energy consumption required for microbial metabolic activities.
- Considerably efficient in sandstone and carbonate reservoirs.
- Microbial metabolic activities enhancement along with time, as opposed to other EOR additives.
- Low environmental pollution.
- Obtaining better results due to occurrence of multiple mechanisms at the same time.
- Possibility of applying to both light and heavy crude oils.
- Possibility to customize microbes to metabolize in extreme and specific subsurface conditions.
- Possibility to exploit MEOR for pipeline and wellbore clean-up in addition to reservoir simulation.
- Extensive research and development within MEOR continues to provide improved knowledge and understanding that is still not fully exploited commercially.

Disadvantages:

- Corrosion of equipment as a result of aerobic bacteria activities.
- Limited applications in offshore platforms in view of requirement of much sugar as anaerobic bacteria activities.
- Complexity of developing a comprehensive model to interpret all aspects of MEOR process.
- Toxicity of microbes due to existence of specific heavy metal ions.
- Microorganisms tolerance limitations in regard to reservoir conditions.
- Reservoir souring caused by some microbes in certain subsurface conditions.
- Potential to cause formation damage in certain conditions.
- Limited tolerances of many microbes to extreme and varying reservoir conditions.
- Extensive laboratory and pilot-testing required to customize MEOR to suit specific reservoir.
- The long-term operational burden for developments in remote areas due to the ongoing nutrient requirements of some microbes.

Chapter 3

Mathematical models

Our system is a two-phase flow system with oil and water as the two phases. In addition to oil and water, our system contains of nutrients, biofilm and gas. In this Chapter we will explain the equations used to model our problem. In the next Chapter we will discuss how we implemented our problem in MATLAB using MRST.

3.1 Equations for water and oil

Mass conservation equations have been used to model water and oil in the system. The two equations are specifications of the general mass conservation equation described in section 2.3.

The mass conservation equation for water is given by

$$\frac{\partial}{\partial t}(\rho_w \phi S_w) + \nabla \cdot (\rho_w \mathbf{v}_w) = 0. \tag{3.1}$$

Equivalently, the mass conservation equation for oil is given by

$$\frac{\partial}{\partial t}(\rho_o \phi S_o) + \nabla \cdot (\rho_o \mathbf{v}_o) = 0. \tag{3.2}$$

In the equations above, ρ_α denotes the density, $\mathbf{v}_\alpha$ denotes the velocity and S_α the

saturation. The porosity is denoted by ϕ (Lie, 2016).

3.2 Equations for nutrients and gas

The equations used for modeling nutrients and gas are transport equations. We assume that the gas is dissolved in the oil phase. Hence, we model it with a transport equation.

The transport equation for nutrients is given by (Li et al., 2010):

$$\frac{\partial(C_n \phi S_w)}{\partial t} - \nabla \cdot (D_n^{eff} S_w \phi \nabla C_n - \mathbf{u}_w C_n) = R_n. \tag{3.3}$$

Similarly, the equation for modeling gas in our system is given by

$$\frac{\partial(C_g \phi S_o)}{\partial t} - \nabla \cdot (D_g^{eff} S_o \phi \nabla C_g - \mathbf{u}_o C_g) = R_g. \tag{3.4}$$

Here C_n denotes the concentration of nutrients and C_g the concentration of gas. $\mathbf{u}_\alpha$ is the Darcy flux vector for the water phase and oil phase. The effective diffusion coefficient of nutrients and gas is denoted by D_α^{eff}. R_n and R_g represent the reaction term given by

$$R_n = -\mu_{b\ \max} \frac{C_n}{K_{b/n} + C_n} (\rho_b \sigma), \tag{3.5}$$

$$R_g = \mu_{g\ \max} \frac{C_n}{K_{b/n} + C_n} (\rho_o \sigma). \tag{3.6}$$

In Equation (3.6) $\mu_{g\ \max} = Y_g \mu_{b\ \max}$, where Y_g represent the yield coefficient for the gas. $\mu_{b\ \max}$ denotes the maximum specific biomass production rate. Similarly, $\mu_{g\ \max}$ denotes the maximum gas production rate in Equation (3.5). We will explain yield coefficients in more detail in Section 3.3.

The biggest difference between the two reaction terms is that the reaction term for the nutrients is negative and the reaction term for the gas is positive. The reaction

term for the nutrients is negative because nutrients are being consumed by the biofilm. The gas on the other hand is produced from the biofilm and therefore the reaction term is positive. The gas produced is dissolved in the oil phase. $K_{b/n}$ is the half-saturation constant for the biofilm. The half-saturation constant describes when the concentration of the specific growth rate reaches half of its maximum value (Li et al., 2010).

3.3 Equation for the biofilm

To model the biofilm we use the equation for reversible deposited bacteria proposed by Kim (2006) and later used by Li et al. (2010). We will use the same notation as Li et al. (2010) did in their work. The equation is given by

$$\frac{\partial(\rho_b\sigma_1)}{\partial t} = k_1(\phi_0 - \sigma)C_b - k_2\rho_b\sigma_1 + g_1\rho_b\sigma_1 - d_1\rho_b\sigma_1. \tag{3.7}$$

Here, σ_1 is the volumetric fraction of bacteria attached reversibly. k_1 denotes the reversible attachment rate coefficient and k_2 the irreversible attachment coefficient. ρ_b is the density of the bacteria and ϕ_0 is the initial porosity. The volumetric fraction of bacteria attached totally is denoted by σ. C_b is the concentration of bacteria. g_1 is the biofilm growth coefficient and d_1 is the bacteria decay rate coefficient (Kim, 2006).

In our case we are neglecting the attachment of bacteria that is detached due to the fluid flow so there will be no reversible attachment rate, this means $k_1 = 0$. In addition, we have gas produced due to the biofilm so we have to add another term to the equation. Therefore (3.7) can be written as

$$\frac{\partial(\rho_b\sigma_1)}{\partial t} = -k_2\rho_b\sigma_1 + Y_b g_1\rho_b\sigma_1 + Y_g g_1\rho_b\sigma_1 - d_1\rho_b\sigma_1. \tag{3.8}$$

Here Y_b and Y_g represent the yield coefficients for the biofilm and gas respectively. The yield coefficients tell us how much of the nutrients consumed by the biofilm goes towards growth of the biofilm and how much turns to gas. In our system, Y_b and Y_g are constants and we assume that the sum of them is one. In general, this is

not always the case. In processes where the dynamic of energy changes there will be non-constant yield coefficients (Landa-Marbán et al., 2019). If the yield coefficients are non-constant, the sum of them will not be one.

As mentioned above, g_1 denotes the biofilm growth rate and it is given by

$$g_1 = \mu_{b\,\max} \frac{C_n}{K_{b/n} + C_n}. \tag{3.9}$$

Here, $\mu_{b\,\max}$ is the maximum growth rate (Li et al., 2010) and $K_{b/n}$ is the half-saturation constant for the biofilm.

The porosity will be affected when the biofilm changes. Li et al. (2010) defines the modified porosity caused by bacterial attachment as

$$\phi - \sigma = \phi_0 - \sigma_1 - \sigma_2 \geq 0. \tag{3.10}$$

As previously mentioned, σ denotes the volumetric fractions of bacteria attached totally and σ_1 denotes the volumetric fraction of bacteria attached reversibly. σ_2 denotes the bacteria attached irreversibly. In our case, we are considering biofilm already injected into the system. The biofilm will grow by consuming nutrients, parts of the biofilm will die and some of it will detach due to the water flux in the system. We do not consider reattachment of the biofilm detached by the water flux. In our problem we only look at the total volumetric fraction of the biofilm and we denote it by σ.

Since we are only looking at the total volumetric fraction of the biofilm, (3.10) becomes

$$\phi = \phi_0 - \sigma \geq 0, \tag{3.11}$$

which is the relationship we have used to model the change of porosity.

In addition, since we only look at the total volumetric fraction of the biofilm,

Equation (3.8) can be written as

$$\frac{\partial(\rho_b\sigma)}{\partial t} = -k_2\rho_b\sigma + Y_b g_1\rho_b\sigma + Y_g g_1\rho_b\sigma - d_1\rho_b\sigma. \tag{3.12}$$

When preforming bioclogging in a porous medium it effects both the storage and the flux of the fluids (Li et al., 2010). The effects on the storage are taken care of by the modification of the porosity mentioned in Equation (3.11) (Li et al., 2010).To model the effects on the flux of the fluids we alter the absolute permeability using a parametric model. In this model, the relationship between the permeability and the modified porosity can be defined by a single parameter C (Li et al., 2010). The parametric model is given by

$$k = k_0\left(\frac{\phi}{\phi_0}\right)^C = k_0\left(1 - \frac{\sigma}{\phi_0}\right)^C. \tag{3.13}$$

Here k is the absolute permeability and k_0 is the initial permeability. C is a parameter that, depending on the model assumptions, varies from 2 to 19/6. In general, the permeability is a tensor but in our system k is a scalar since we will consider an isotropic homogeneous porous medium.

3.4 Viscosity models

In this work, we want to study what happens to the oil viscosity when it is modeled as a function of the gas production, i.e. $\mu_o = \mu_o(C_g)$. This relationship should ideally be determined by experiments but we want to see how different heuristic relationships between oil viscosity and the concentration of gas can effect the oil production. In Section 5.3 we present two heuristic equations for the oil viscosity and the resulting oil production.

3.5 Capillary pressure and relative permeability

There are many ways to model the capillary pressure and relative permeabilities. In this Thesis we have used the model suggested by Brooks and Corey.

To model the capillary pressure we use the relationship

$$p_c = p_e(S_w^*)^{-1/\lambda}. \tag{3.14}$$

Here p_e represent the entry capillary pressure and λ is the pore size distribution index (Li and Horne, 2006). S_w^* is the normalized wetting-phase saturation given by

$$S_w^* = \frac{S_w - S_w^{res}}{1 - S_w^{res}}. \tag{3.15}$$

Here S_w is the water saturation and S_w^{res} is the residual saturation for the wetting phase. The model originally introduced by Corey did not consider the imbibition case, it only considered the drainage curve. In the work of Li and Horne (2006), they extended the expression for normalized wetting-phase saturation given in (3.15) to also consider the imbibition case. The relationship was extended to

$$S_w^* = \frac{S_w - S_w^{res}}{1 - S_w^{res} - S_n^{res}}, \tag{3.16}$$

where S_n^{res} is the residual saturation for the non-wetting phase (Li and Horne, 2006). This is the relationship we have used for the normalized wetting-phase saturation in this thesis.

To model the relative permeability we have used the modified Brooks and Corey model, also referred to as the power law model (Goda and Behrenbruch, 2004). The

model can be expressed as

$$k_{r,w} = k'_{r,w}(S_w^*)^{n_w} = k'_{r,w}\left(\frac{S_w - S_w^{res}}{1 - S_w^{res} - S_n^{res}}\right)^{n_w} \tag{3.17a}$$

$$k_{r,o} = k'_{r,o}(1 - S_w^*)^{n_o} = k'_{r,o}\left(1 - \frac{S_w - S_w^{res}}{1 - S_w^{res} - S_n^{res}}\right)^{n_o}, \tag{3.17b}$$

where $k_{r,w}$ and $k_{r,o}$ represent the relative permeability for the water and oil phase respectively, $k'_{r,w}$ and $k'_{r,o}$ are the endpoint relative permeabilities for the two phases and n_w and n_o are the Corey exponents for water and oil (Goda and Behrenbruch, 2004). All the relative permeabilities are normalized to absolute plug air permeability.

3.6 Summary of equations

Below, you see a summary of the equations used in our model. In the next chapter we will describe how we have implemented the equations in MRST and then later show analytical solutions and numerical examples.

$\frac{\partial}{\partial t}(\rho_w \phi S_w) + \nabla \cdot (\rho_w \mathbf{v}_w) = 0$	in $\Omega \times [0,t]$
$\frac{\partial}{\partial t}(\rho_o \phi S_o) + \nabla \cdot (\rho_o \mathbf{v}_o) = 0$	in $\Omega \times [0,t]$
$\mathbf{v}_w = -\frac{\boldsymbol{k}_{r,w}\boldsymbol{k}}{\boldsymbol{\mu}_w}(\nabla p_w + \rho_w \boldsymbol{g})$.	in $\Omega \times [0,t]$
$\mathbf{v}_o = -\frac{\boldsymbol{k}_{r,o}\boldsymbol{k}}{\boldsymbol{\mu}_o}(\nabla p_o + \rho_o \boldsymbol{g})$.	in $\Omega \times [0,t]$
$\frac{\partial (C_n \phi S_w)}{\partial t} - \nabla \cdot (D_n^{eff} S_w \phi \nabla C_n - \mathbf{u}_w C_n) = R_n$	in $\Omega \times [0,t]$
$\frac{\partial (C_g \phi S_o)}{\partial t} - \nabla \cdot (D_g^{eff} S_o \phi \nabla C_g - \mathbf{u}_o C_g) = R_g$	in $\Omega \times [0,t]$
$R_n = -\mu_b \max \frac{C_n}{K_{b/n}+C_n}(\rho_b \sigma)$	in $\Omega \times [0,t]$
$R_g = \mu_g \max \frac{C_n}{K_{b/n}+C_n}(\rho_o \sigma)$	in $\Omega \times [0,t]$
$\frac{\partial (\rho_b \sigma)}{\partial t} = -k_2 \rho_b \sigma + Y_b g_1 \rho_b \sigma + Y_g g_1 \rho_b \sigma - d_1 \rho_b \sigma$	in $\Omega \times [0,t]$
$g_1 = \mu_b \max \frac{C_n}{K_{b/n}+C_n}$	in $\Omega \times [0,t]$
$\phi = \phi_0 - \sigma \geq 0$	in $\Omega \times [0,t]$
$k = k_0(1 - \frac{\sigma}{\phi_0})^C$	in $\Omega \times [0,t]$
$p_c = p_e (S_w^*)^{-1/\lambda}$	in $\Omega \times [0,t]$
$S_w^* = \frac{S_w - S_w^{res}}{1 - S_w^{res} - S_n^{res}}$	in $\Omega \times [0,t]$
$k_{r,w} = k'_{r,w}(S_w^*)^{n_w}$	in $\Omega \times [0,t]$
$k_{r,o} = k'_{r,o}(1 - S_w^*)^{n_o}$	in $\Omega \times [0,t]$
$\mu_o = \mu(C_g)$	in $\Omega \times [0,t]$
Initial conditions for C_g, C_n, S_w, S_o, ϕ_0, $\mathbf{k}_0$, p_w, p_o, σ	in $\Omega \times [0,t]$
Boundary conditions for C_n, C_g, p_w, p_o, S_w, S_o, $\mathbf{v}_w$, $\mathbf{v}_o$	on $\partial\Omega \times [0,t]$

Chapter 4

Implementation in MATLAB

To model our two-phase flow problem we used the MATLAB Reservoir Simulation Toolbox (MRST) from SINTEF. MRST is a free open-source software for reservoir simulation and modeling (Lie, 2019). The software is organized into two parts; the core module consisting of the basic data structures and functionality and the add-on modules consisting of solvers, physical models and a large number of simulators and workflow tools.

The base for our problem is the polymer simulator in MRST, more specifically the PolymerBCExample.m script in the ad-eor module. We have manipulated the code to match the problem we want to study. There are three main MRST-files we have changed in the ad-eor module. The first one is PolymerBCExample, this is the script with all the data for the problem, e.g. the fluid parameters, rock parameters and domain parameters. The next file is the MATLAB function for the equations. Here, all the equations for the system are solved. The last file is the class with the actual model. This is where we define everything that is in the model like the properties, e.g. what phases the model contains, and methods used to solve the system.

We study our problem on a core-scale level. The core we have simulated has dimensions 0.3 m$\times$0.05 m$\times$0.05 m in the x-, y- and z-direction respectively. The domain is discretized with 100 steps in the x-direction and 5 steps in the y- and z-direction, giving a total of 1188 cell centroids. Originally, our domain is square but we wanted to study a cylindrical core so we removed the cells outside a radius

of $0.025^2 = 6.25e - 4$. We have neglected the effect of gravity.

In this chapter we will explain how we have implemented our system step by step. At the end of the chapter we present the values for the parameters we have used in our model. They are listed in Table 4.1.

4.1 Numerical methods used in MRST

Before we go into how we have implemented our system, we will describe some of the numerical methods used in MRST and a general overview of how the simulator for polymer injection is build up.

Our problem is modeled with a nonlinear system. The standard way to approach a nonlinear system is to compute the Jacobian matrix of first order derivatives for the system and then use Newton's method to consecutively find a better approximation to the solution (Lie, 2019). In MRST, they use automatic differentiation to compute the Jacobian matrices.

4.1.1 Automatic differentiation

As mentioned above, MRST use automatic differentiation to compute the Jacobian matrices needed to solve our system. The idea behind automatic differentiation (AD) is to keep track of quantities and their derivatives simultaneously. This is done by applying the corresponding differential operator to the derivative every time an operation is applied to a quantity (Lie, 2019).

Lets consider a scalar variable x and a function $f = f(x)$. The AD-representations of this variable and function would be the pairs $\langle x, 1\rangle$ and $\langle f, f_x\rangle$. This is because $\frac{dx}{dx} = 1$ and $\frac{df}{dx} = f_x$ (Lie, 2019). Now, the elementary functions and operations from calculus must be defined for pairs like these. Some examples are given in Lie (2019), and to get a better understanding of what we mean, we have repeated them bellow;

$$\langle f, f_x\rangle + \langle g, g_x\rangle = \langle f + g, f_x + g_x\rangle,$$
$$\langle f, f_x\rangle \cdot \langle g, g_x\rangle = \langle fg, fg_x + f_x g\rangle,$$
$$\langle f, f_x\rangle / \langle g, g_x\rangle = \langle f/g, (f_x g + f g_x)/g^2\rangle,$$

$$exp(\langle f, f_x \rangle) = \langle exp(f), exp(f) f_x \rangle.$$

Here, they have used the product rule, quotient rule and chain rule from calculus.

There are many libraries on AD for MATLAB. The thing that makes MRST different is that MRST use a list of matrices that represent the derivatives with respect to different variables, instead of using one big matrix to represent the Jacobian for the full discrete system. The list of matrices used in MRST are sub-blocks from the Jacobian for the whole system (Lie, 2019).

4.1.2 Two-point flux approximation

MRST use the two-point flux approximation (TPFA) to approximate the flux across the boundaries between the cells in the domain. TPFA is derived from the principle of conservation of quantities over the cell volumes (Lie, 2019). Lets consider the simplified single-phase flow equation:

$$\nabla \cdot \vec{v} = q, \ \vec{v} = -K \nabla p, \ \text{in } \Omega \subset \mathbb{R}^d. \tag{4.1}$$

If we rewrite the equation above in integral form using Ω_i, where Ω_i is a single cell in the discrete grid, as control volume we get (Lie, 2019)

$$\int_{\partial \Omega_i} \vec{v} \cdot \vec{n} \ ds = \int_{\Omega_i} q \ d\vec{x}. \tag{4.2}$$

This is similar to Equation (2.5) described in Section 2.3, but here $\frac{\partial m}{\partial t} = 0$ since ρ and ϕ are not dependent on time. ρ has been eliminated since it is a constant and therefore it will not effect the derivation of the finite-volume discretization.

Next, Darcy's law is used to compute the flux across each face of the cell.

$$v_{i,k} = \int_{\Gamma_{i,k}} \vec{v} \cdot \vec{n} \ ds, \ \Gamma_{i,k} = \partial \Omega_i \cap \partial \Omega_k. \tag{4.3}$$

The half-face $\Gamma_{i,k}$ is the face of the cell Ω_i and it is associated with the normal vector $\vec{n}_{i,k}$. Similarly, $\Gamma_{k,i}$ is the face of cell Ω_k with respect to the normal vector

$\vec{n}_{k,i}$. Since the cells are matching, each half-face $\Gamma_{i,k}$ have a twin half-face $\Gamma_{k,i}$. The two half-faces have the same area but opposite normal vectors, i.e. $A_{i,k} = A_{k,i}$ but $n_{i,k} = -n_{k,i}$. If we use this, and further assume that the integral in Equation (4.3) can be approximated by the midpoint rule, Darcy's law can be used to write the flux as

$$v_{i,k} \approx A_{i,k}\vec{v}(\vec{x}_{i,k}) \cdot \vec{n}_{i,k} = -A_{i,k}(K\nabla p)(\vec{x}_{i,k}) \cdot \vec{n}_{i,k}. \tag{4.4}$$

Here, $\vec{x}_{i,k}$ represent the centroid on $\Gamma_{i,k}$. We want to be able to calculate the pressure gradient in Equation (4.4). We only know the cell averaged value of the pressure but we need the point values in order to calculate the pressure gradient. Therefore we need to make some more assumptions (Lie, 2019). We assume that the pressure is linear inside each cell and that the reconstructed pressure value $\pi_{i,k}$ at the cell face is identical to p_i inside the cell. By using the assumptions above we get

$$v_{i,k} \approx A_{i,k}K_i \frac{(p_i - \pi_{i,k})\vec{c}_{i,k}}{|\vec{c}_{i,k}|^2} \cdot \vec{n}_{i,k} = T_{i,k}(p_i - \pi_{i,k}). \tag{4.5}$$

$T_{i,k}$ is one-sided transmisibilities related to a single cell. It gives a two-point relationship between the flux across a cell face and the difference between the pressure at the cell centroids and the face centroids.

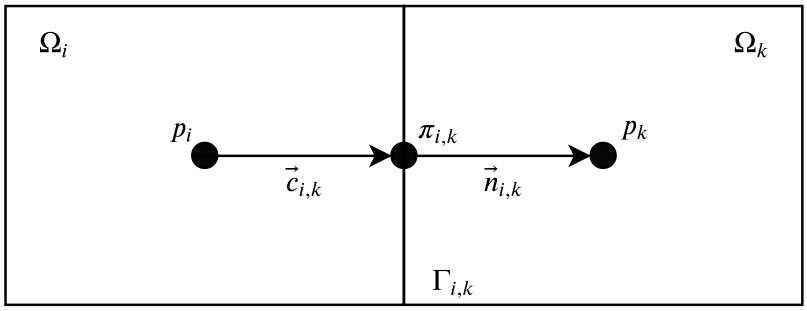


Figure 4.1: A schematic of two-point flux approximation. Figure inspired by Lie (2019).

Next, we impose continuity of fluxes across all faces and continuity of face pressures, i.e. $\vec{v}_{i,k} = -\vec{v}_{k,i} = \vec{v}_{ik}$ and $\pi_{i,k} = \pi_{k,i} = \pi_{ik}$ (Lie, 2019). This gives us two equations:

$$T_{i,k}^{-1} v_{ik} = p_i - \pi_{ik} \tag{4.6a}$$

$$-T_{k,i}^{-1} v_{ik} = p_k - \pi_{ik} \tag{4.6b}$$

By subtracting these two equations, we eliminate π_{ik} and get the relationship

$$v_{ik} = [T_{ik}^{-1} - T_{ki}^{-1}]^{-1}(p_i - p_k) = T_{ik}(p_i - p_k), \tag{4.7}$$

where T_{ik} is the transmissibility associated with the connection between the two cells (Lie, 2019). By inserting v_{ik} into Equation (4.2) we get

$$\int_{\partial\Omega_i} T_{ik}(p_i - p_k) \cdot \vec{n} \; ds = \int_{\Omega_i} q \; d\vec{x}. \tag{4.8}$$

From this, we can see that the two-point flux approximation scheme for the simplified single-phase flow equation (4.1), in compact form, seeks a set of cell averages that satisfy the following system of equations (Lie, 2019):

$$\sum_k T_{ik}(p_i - p_k) = q_i \quad \forall \; \Omega_i \subset \Omega \tag{4.9}$$

This system of equations is symmetric and there exist a solution up to an arbitrary constant. The system is made positive definite and we keep the symmetry by specifying the pressure in a single point. In MRST, they have added a positive constant to the first diagonal of the matrix $\mathbf{A} = [a_{ij}]$ in order to set $p_1 = 0$. $[a_{ij}]$ is given by

$$[a_{ij}] = \begin{cases} \sum_k T_{ik} & \text{if } j = i \\ -T_{ij} & \text{if } j \neq i. \end{cases} \tag{4.10}$$

The matrix $\mathbf{A}$ is a sparse matrix with a banded structure. The number of diagonals depends on the dimension of the grid in the problem. A 1D grid gives a tridiagonal

matrix (Lie, 2019).

4.1.3 Newton's method

MRST's nonlinear solver uses Newton's method to find the best approximation to the solution.

To explain the method we look at function f and we want to determine the roots of the function. We let r be a root of f and x be an approximation of r (Kincaid, 2002). If f'' exists and is continuous we can use Taylor's theorem and we get the relationship

$$0 = f(r) = f(x + h) = f(x) + hf'(x) + \mathcal{O}(h^2), \tag{4.11}$$

where $h = r - x$. We can ignore $\mathcal{O}(h^2)$ if x is close to r and then solve the remaining equation for h;

$$h = -\frac{f(x)}{f'(x)}. \tag{4.12}$$

If x is an approximation of r, then $x - \frac{f(x)}{f'(x)}$ is a better approximation of r (Kincaid, 2002). We therefore use $x - \frac{f(x)}{f'(x)}$ to estimate r. To begin Newton's method, we make a prediction x_0 of r and then use the relationship in Equation (4.13) to estimate x_1.

$$x_{n+1} = x_n - \frac{f(x)}{f'(x)} \quad \text{where } n \geq 0. \tag{4.13}$$

The method continue to do this until it finds an acceptable approximation of r. The derivative in Equation (4.13) is calculated by using AD in MRST.

4.1.4 Euler's method

Forward Euler's method (FE) is the Taylor-series with n=1. It is given on the form:

$$x_{n+1} = x_n + hf(t_n, x_n) \quad \text{where } n \geq 0. \tag{4.14}$$

Forward Euler's method is an explicit method, meaning it uses the known values from this time step to estimate the answer for the next time step. The advantage of using forward Euler's method is that it is not necessary to compute the derivatives. On the other hand, the method requires h to be small in order to get acceptable precision (Kincaid, 2002). In Equation (4.14), h is the time step and $f(t_n, x_n) = \frac{dx(t)}{dt}$.

Euler's method can also be expressed as an implicit method. It is then referred to as Backward Euler's method (BE) and it is given by:

$$x_{n+1} = x_n + hf(t_{n+1}, x_{n+1}) \quad \text{where } n \geq 0. \tag{4.15}$$

BE uses both the value at the current time step and at the next time step to estimate the solution at the next time step.

MRST use backwards Euler to discretize our system with regards to time before using Newton method to solve the system. The reason why MRST use BE and not FE is because backwards Euler is stable. It is not dependent on a small value for h in order to converge.

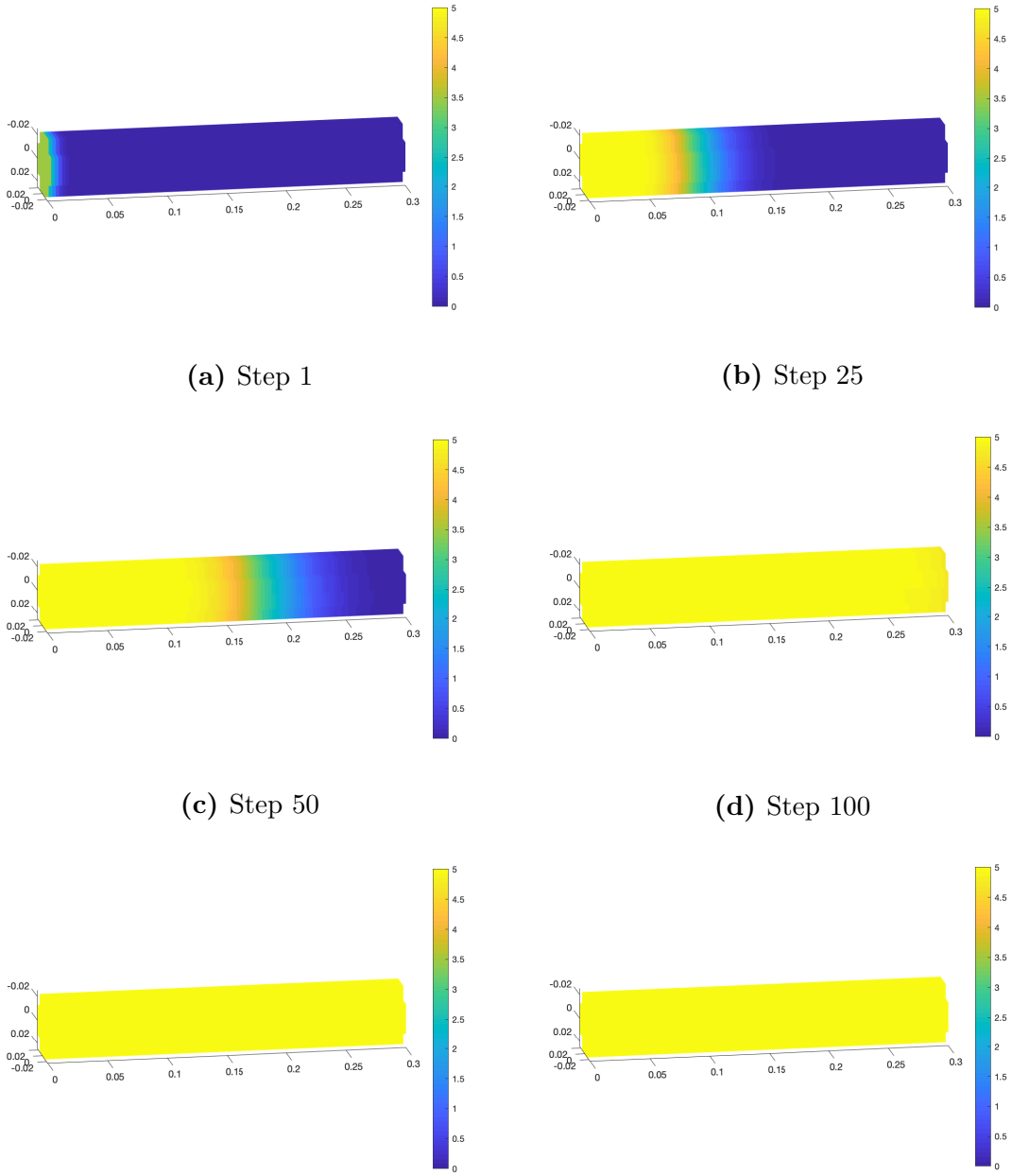
4.1.5 The polymer simulator in MRST

The simulator we have used to build our model, the one for polymer flooding in MRST, is a fully implicit simulator for polymer injection. It uses upstream weighting to discretize in space and an implicit method to discretize in time. By using this combination the simulator offers unconditional stability for many different physical flow regimes and different variations in the reservoir (Bao et al., 2017). The module is easy to extend since it combines fully implicit formulation with automatic differentiation. It lets us implement equations in a very compact form. After we

have implemented the discrete equations, the software will generate the discretizations and linearizations needed to have a working simulator that runs on general unstructured grids. The software uses a separate inner Newton iteration process within a global nonlinear solution process (Bao et al., 2017). The simulator takes into account possible inaccessible pore space. It also takes into account physical adsorption and how it will reduce the effective permeability of the system due to the resistance introduced when the polymer attach to the surface of the rock. The effective permeability of the rock with respect to water is also modified automatically in the simulator. To compute the effective viscosities, MRST uses the Todd-Longstaff mixing model. This model uses a mixing parameter ω to represent the degree of mixing. The parameter is in the interval $\omega \in [0, 1]$, where if $\omega = 0$ means that the polymer solution is completely segregated from the pure water. If $\omega = 1$, water and polymer is fully mixed. For more detailed information about the Todd-Longstaff mixing model and the polymer simulator in general, see Bao et al. (2017).

4.2 Implementation of nutrients

Now, lets move on to how we have implemented our system. As indicated by the name of the file we have modified, PolymerBCExample, polymer was originally modeled in the code. We modified the model to consist of nutrients instead by renaming the parameters for the polymers and adding the equation for nutrients described in Section 3.2. Now we have a system containing of oil, water and nutrients where the nutrients are injected with the water from the left-hand side. Figure 4.2 shows a simulation of the amount of nutrients in the core at different time steps when there is only nutrients, water and oil in the core.



(a) Step 1

(b) Step 25

(c) Step 50

(d) Step 100

(e) Step 1000

(f) Step 2500

Figure 4.2: Simulation of the results when the system contains of water, oil and nutrients. The steps shows which time step the simulation is from.

The simulations in Figure 4.2 ran for 2500 time steps, where each time step is 20 seconds long. The steps mentioned in the caption of each sub figure represents which

time step the simulation is from. The water consisting of nutrients is injected on the left-hand side of the core with the same boundary conditions on the whole left side of the domain, as seen in Figure 4.2a. We can see from the figure that there were no nutrients in the core when we start the simulation. After we have ran the simulation for 2500 time steps, we have nutrients in the whole domain. This makes sense, since the injection of nutrients is constant and there is nothing else in the core blocking the way for the nutrients or consuming the nutrients.

The colorbar in Figure 4.2 is restricted from 0 to 5. We inject 5 kg/m^3 nutrients at a constant rate into the system. Since there is no source of nutrients inside the core, the maximum possible concentration of nutrients remains 5 kg/m^3. That is why the colorbar is restricted between 0 and 5.

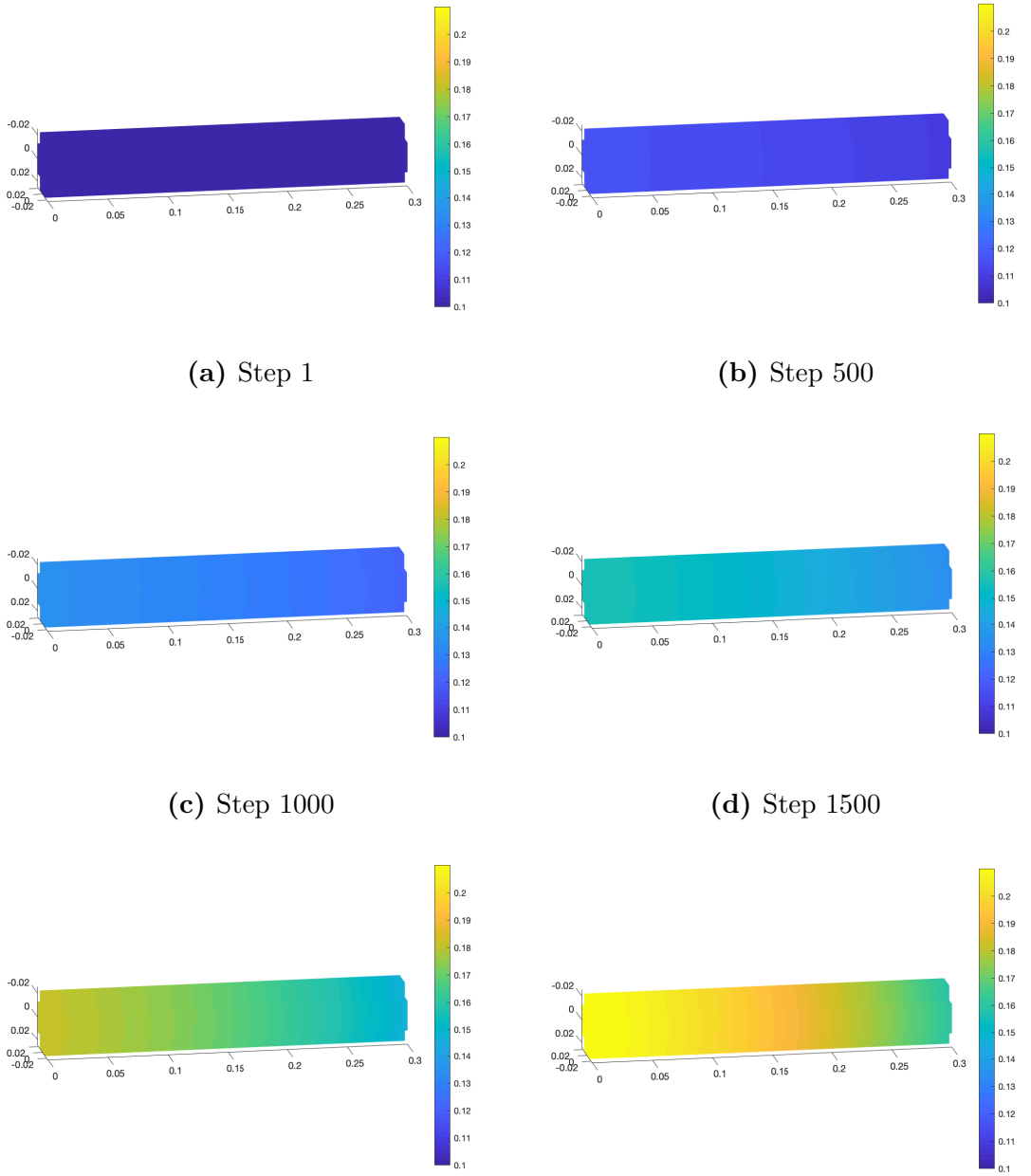
4.3 Implementation of biofilm

After implementing nutrients we implemented the biofilm. In this thesis, we are not looking at the injection of bacteria, we consider bacteria already injected into the core where the bacteria have formed a biofilm. A biofilm is a film on the pore walls consisting of biomass and water. We study how the biofilm grows within the system.

The biofilm originally in the core is spread equally with a volumetric fraction of 0.1 in all the cells. Since we study the volumetric fraction of the biofilm, the biofilm is dimensionless.

When adding biofilm to our system we first had to add biofilm as a property in our model. After adding the necessary variables we could add the equation described in Section 3.3. In addition to adding the equation for biofilm, we also implemented the changes to porosity and the permeability mentioned in Section 3.3.

Figure 4.3 show the simulation of biofilm when the system contains of nutrients, biofilm, water and oil. Also here we have run the simulation for 2500 time steps where each time step is 20 seconds long. We have done that for all the simulations in this chapter.



(a) Step 1

(b) Step 500

(c) Step 1000

(d) Step 1500

(e) Step 2000

(f) Step 2500

Figure 4.3: Simulation of the biofilm when the system contains of water, oil, nutrients and biofilm. Here the colorbar is locked between 0.1 and 0.21.

Notice that the colorbar is locked between 0.1 and 0.21. We have done this in order to see how the biofilm grows in the system. As mentioned before when describing

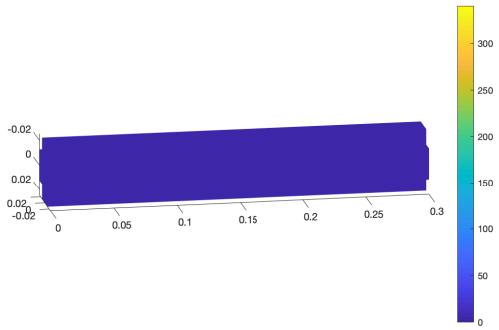
the equation for the biofilm, we only consider the total volumetric fraction of the biofilm. When we looked at the results for the volumetric fraction of biofilm after the last time step we saw that the maximum volumetric fraction was just above 0.21. This is why we have chosen this value as the maximum value on the colorbar.

We can see from the plots that the biofilm grows as the nutrients are injected into the system. At the last time step shown in Figure 4.3f, there is more biofilm in the beginning of the core than at the end. This is what we expect, since the biofilm at the left-hand side of the domain is more exposed to nutrients and therefore have a greater opportunity to grow since the biofilm consume nutrients to grow. At the end of this Chapter we will have a closer look at how the nutrients are affected by the inclusion of biofilm to the system.

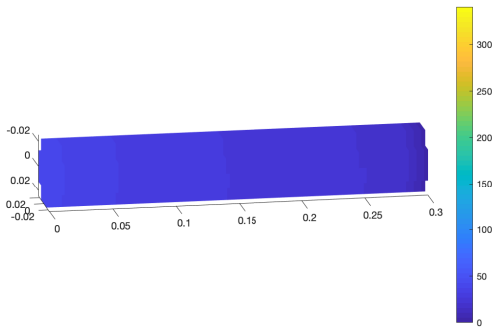
4.4 Implementation of gas

The last thing we implemented into our system was the gas. As mentioned in Section 3.2, we do not consider gas as a third phase. Therefore we could implement it in a similar way as the nutrients. As with the biofilm, we had to add gas as a property and add all the necessary parameters before we could implement the equation discussed in Section 3.2.

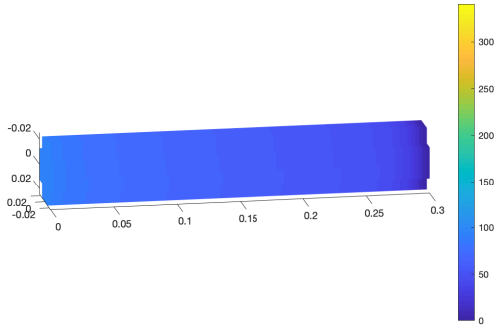
Figure 4.4 show the simulation of how the gas develops in the core at different time steps. The system now consists of gas, biofilm, nutrients, water and oil.



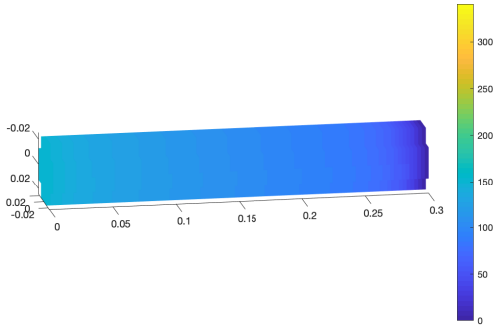
(a) Step 1



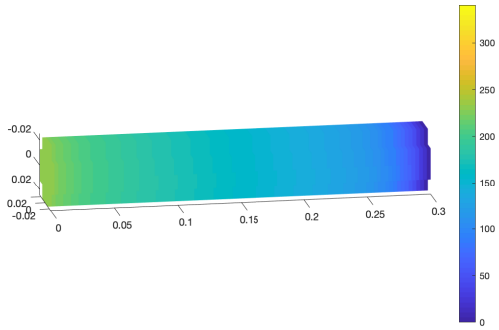
(b) Step 500



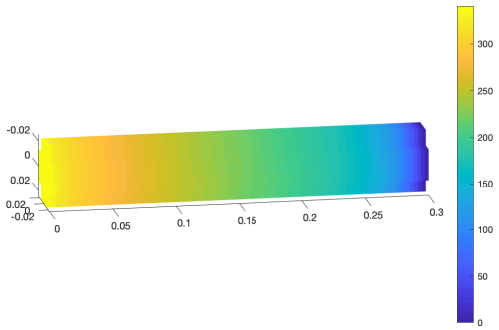
(c) Step 1000



(d) Step 1500



(e) Step 2000



(f) Step 2500

Figure 4.4: Simulation of the gas when there is gas, biofilm, nutrients, water and oil in the system.

Here, the colorbar goes from 0 to 340. At the beginning of our simulation there is zero gas in the system. This correspond to the result in Figure 4.4a. To determine

the top boundary of the colorbar we used the same approach as for the biofilm, we looked at the state of the gas at the last time step and found the maximum value. The highest concentration of gas in our system at time step 2500 is approximately 338 kg/m^3. Because of this, we have set the top boundary of the colorbar to 340.

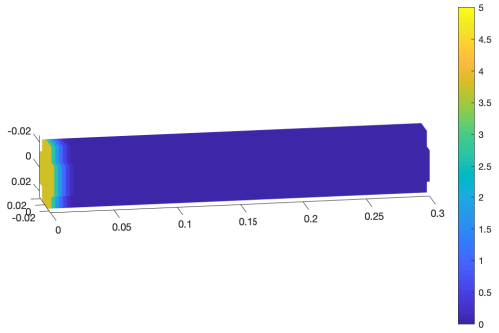
We see in Figure 4.4 that the gas develops the same way in the core as the nutrients and biofilm. This is logical since the gas is a product of the biofilm and therefore the gas concentration will advance faster where it is more biofilm.

4.4.1 Boundary conditions

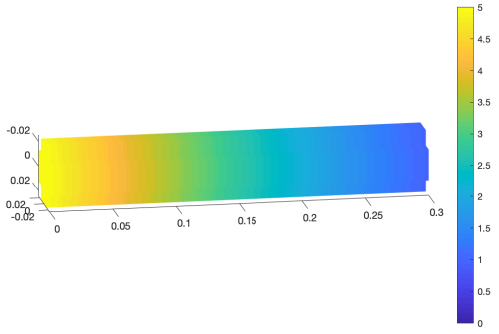
Originally, Neumann boundary conditions and Dirichlet boundary conditions (BC) were implemented in the polymerBCExample script on the left- and right-hand side of the core. We are not considering the pressure so we removed the Dirichlet BC and modified the Neumann BC to fit our problem. The Neumann boundary conditions originally implemented on the right-hand side of the system were defined as zero flow conditions. We want to have flow out on the right side of the core so we removed the BC for the right-hand side but kept the BC for the left-hand side of the core to let the injection of water enter the system. On the other sides of the domain we have no-flow conditions, in other words $\vec{v} \cdot n = 0$. This means that the flow will only flow out on the right-hand side of the core.

4.5 The full numerical example

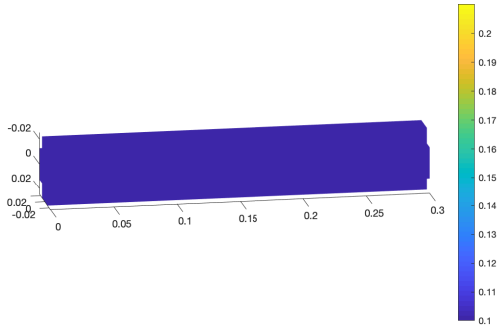
Now that we have introduced how we have implemented the different parts of our model, lets look at the model as a whole. In Figure 4.5 we have plotted the simulation of nutrients, biofilm and gas at the first and last time step when everything is implemented in the model. In Table 4.1 we have listed the values and parameters used to get these results.



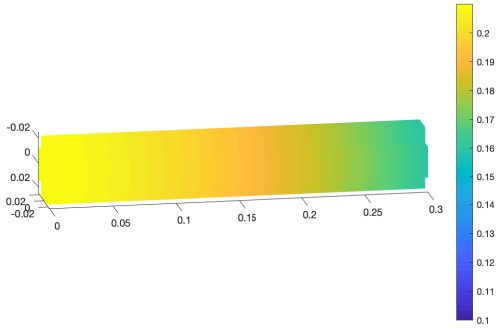
(a) Step 1, nutrients.



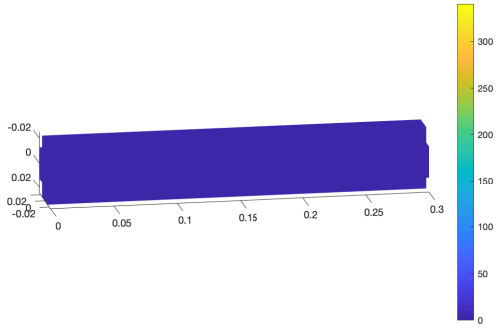
(b) Step 2500, nutrients.



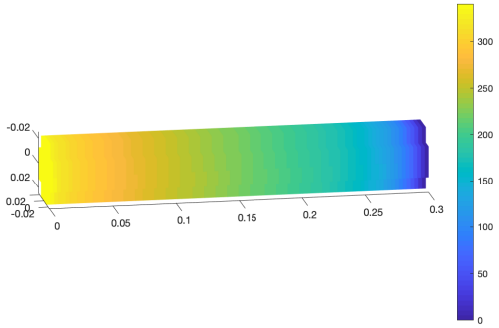
(c) Step 1, biofilm.



(d) Step 2500, biofilm.



(e) Step 1, gas.



(f) Step 2500, gas.

Figure 4.5: Simulation of the different parts in the simulation at the first time step and the last time step.

If we look at the plots in Figure 4.5, the important thing to notice is that the nutrients in the last time step, see Figure 4.5b, is very different from Figure 4.2f.

We clearly see that adding biofilm and gas to the system have had an impact on the nutrient concentration. This is what we expect since the biofilm consume the nutrients to grow.

In Equation (3.8), Y_g and Y_b represent the yield coefficient for the gas and biofilm respectively. As described in Section 3.3, the yield coefficients describes how much of the nutrients consumed by the biofilm turns to gas and how much go towards growth of the biofilm. In our simulations we have chosen the values $Y_g = 0.5$ and $Y_b = 0.5$, meaning half of the consumed nutrients goes towards the gas production and half of it towards growing the biofilm.

As mentioned above, Table 4.1 sums up the values and units for the parameters used throughout the thesis. Table 4.2 lists up the initial conditions and Table 4.3 specifies the dimensions and properties of the core we have used and the time dimensions.

Table 4.1: Values for the parameters used to build up the full numerical example.

Parameter:	**Value:**	**Unit:**
Water viscosity, μ_w	1e-3	$kg\,m^{-1}\,s^{-1}$
Oil viscosity, μ_o	3.92e-3	$kg\,m^{-1}\,s^{-1}$
Diffusion nutrient coefficient, D_n^{eff}	1.5e-9	$m^2\ s^{-1}$
Diffusion gas coefficient, D_g^{eff}	1.5e-9	$m^2\ s^{-1}$
Water density, ρ_w	1000	$kg\ m^{-3}$
Oil density, ρ_o	800	$kg\ m^{-3}$
Active biofilm density, ρ_b	1025	$kg\ m^{-3}$
Maximum bacteria growth rate, $g_{b,max}$	2e-5	s^{-1}
Yield coefficient for bacteria growth, Y_b	0.5	-
Yield coefficient for gas growth, Y_g	0.5	-
Half-saturation constant for bacteria, $K_{b/n}$	0.9	$kg\ m^{-3}$
Bacterial decay rate, d_1	1e-6	s^{-1}
Stress coefficient, k_2	1e-6	s^{-1}
Maximum nutrient concentration, $C_{n,max}$	5	$kg\ m^{-3}$
Residual water saturation, S_w^{res}	0.2	-
Residual oil saturation, S_o^{res}	0.2	-
Endpoint relative permeability for water, $k'_{r,w}$	0.6	-
Endpoint relative permeability for oil, $k'_{r,o}$	0.5	-
Corey exponent for water, n_w	3	-
Corey exponent for oil, n_o	3	-
Pore size distribution index, λ	2.129	-
Entry capillary pressure, p_e	2.740	$kg\ m\ s^{-2}$
Constant from the absolute permeability model, C	2	-

Table 4.2: Initial conditions for the full numerical example.

Initial conditions:	**Value:**	**Unit:**
Concentration of nutrients, C_n	0	$kg \cdot m^{-3}$
Volumetric fraction of biofilm, σ	0.1	-
Concentration of gas, C_g	0	$kg \cdot m^{-3}$
Initial reservoir pressure, p_0	5199	$kg \cdot m \cdot s^{-2}$
Initial water saturation, S_w	0.3	-
Initial oil saturation, S_o	0.7	-
Initial permeability, k_0	0.94e-12	m^2
Porosity, ϕ_0	0.4	-

Table 4.3: Core details and time specifications.

Core setup:	**Value:**	**Unit:**
Length in the X-direction	0.3	m
Length in the Y-direction	0.05	m
Length in the Z-direction	0.05	m
Number of cells in the X-direction	100	-
Number of cells in the Y-direction	4	-
Number of cells in the Z-direction	4	-
Total number of cells	1188	-
Time specifications:	**Value:**	**Unit:**
Length of time step, dt	20	s
Number of time steps, nt	2500	-

The values of the parameters used in the full numerical example are a combination of values already present in the code from MRST, values from different articles and values from Landa-Marbán (2016) and Landa-Marbán et al. (2017). A few of the values are constructed in order to check if the code runs.

Chapter 5

Numerical studies and results

In this Chapter we check that we have implemented our system correctly by introducing analytical solutions. We also test our implementation against a benchmark example. Next, we test how simulating the viscosity as a function of the gas produced will affect the oil recovery and how the positioning of the biofilm will effect the amount of oil produced.

5.1 Analytical solution

In this Section we want to test if we have implemented the system correctly. To do this, we compute analytical solutions for the transport equation used to simulate the nutrients and gas production.

$$\frac{\partial(C_\alpha \phi S_\alpha)}{\partial t} - \nabla \cdot (D_\alpha^{eff} S_\alpha \phi \nabla C_\alpha - \mathbf{u}_\alpha C_\alpha) = R_\alpha \tag{5.1}$$

Since the equations are implemented in the same way, we only need to test one of the equations. We have chosen to test the implementation of nutrients.

For simplicity reasons we choose all the parameters to be 1, i.e.

$$D_n^{eff} = S_w = \phi = 1 \text{ and } u_w = 1 \text{ in 1D, } \mathbf{u}_w = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ in 2D.}$$

Now Equation (5.1) can be written as

$$\frac{\partial C_n}{\partial t} - \nabla \cdot (\nabla C_n - \mathbf{u}_w C_n) = R_n. \tag{5.2}$$

We have implemented two analytical solutions in 1D and one in 2D.

To calculate the error between the numerical and analytical solution we have used the L_2-norm, also referred to as the Euclidean norm. We define the inner product between two functions as

$$\langle f, g \rangle = \int_0^1 f(x)g(x)\ dx \quad \forall \quad f, g \ \in L_2[0, 1].$$

The inner product induces the L_2-norm (Iserles, 2009):

$$\|f\|_{L_2} = \left(\int_0^1 |f(x)|^2 \ dx \right)^{1/2}. \tag{5.3}$$

We want to calculate the L_2-norm of the error between the numerical and analytical solution. We therefore write Equation (5.3) as

$$\|C_{an} - C_{nu}\|_{L_2} = \left(\int_0^1 |C_{an} - C_{nu}|^2 \ dx \right)^{1/2}. \tag{5.4}$$

Here, C_{an} and C_{nu} are the analytical and numerical solution respectively. The values for the analytical and numerical solution are calculated at the midpoint of the cells. Because of this, we use the midpoint rule to approximate the integral in Equation (5.4).

$$\|C_{an} - C_{nu}\|_{L_2} = \sqrt{\sum_{i=1}^{n} \int_{x_{i-1/2}}^{x_{i+1/2}} (C_{an} - C_{nu})^2 \ dx} \approx \sqrt{\Delta x \sum_{i=1}^{n} (C_{an,i} - C_{nu,i})^2}\ . \tag{5.5}$$

This gives us the following relationship for the error:

$$e_i = \|C_{an} - C_{nu}\|_{L_2} \approx \sqrt{\Delta x \sum_{i=1}^{n} (C_{an,i} - C_{nu,i})^2}. \tag{5.6}$$

When calculating the error in 2D we add Δy to Equation (5.6) to take into account the discretization in the y-direction.

5.1.1 1D

The first equation we have tested in 1D is the Gaussian distribution given by $C_n = te^{-x^2}$. By calculating the terms in Equation (5.2), we get the source/sink term

$$R_n = e^{-x^2}(1 - 2t(2x^2 + x - 1)). \tag{5.7}$$

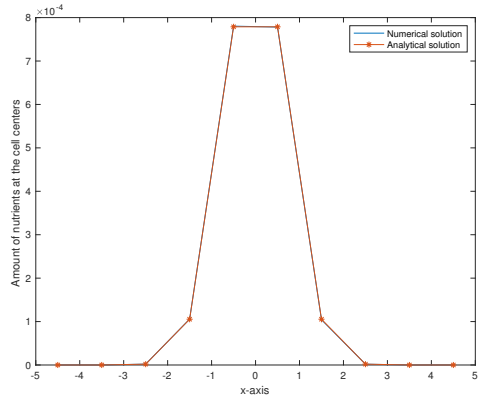
We consider a domain with length L=10, going from -5 to 5, in the x-direction and with length 1 in the y- and z-direction. We have used a total time interval of 0.01 seconds, i.e. $T = nt \cdot dt = 0.01$.

The results can be seen in Table 5.1.

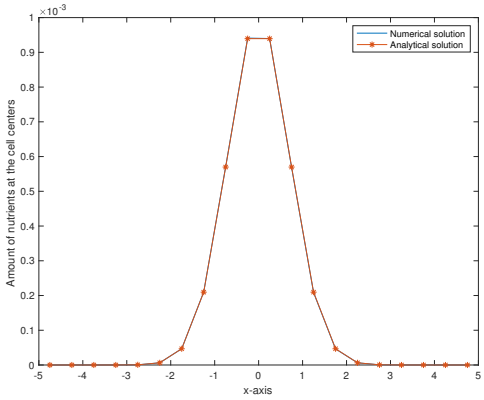
Table 5.1: Results from the analytical solution in 1D for Equation (5.2) when $C_n = te^{-x^2}$. The total time interval is T=0.01 and the domain has length L=10.

	Δx	Δt	L_2 error	Reduction
1	1	1e-3	2.0694e-06	-
2	0.5	0.5e-3	1.9415e-06	1.0659
3	0.25	0.25e-3	1.7408e-06	1.1153
4	0.125	0.125e-3	1.6186e-06	1.0755
5	0.0625	0.0625e-3	1.5517e-06	1.0431

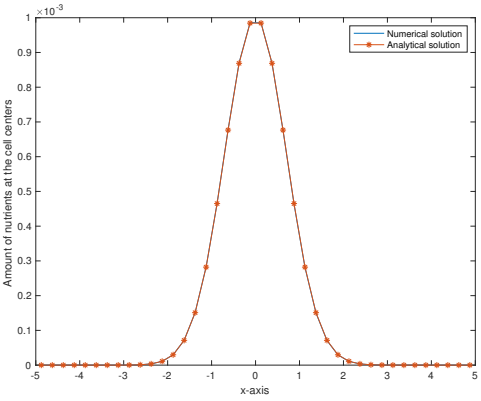
We now plot the analytical and numerical solution in order to compare the solutions for the five cases in Table 5.1. The result can be seen in Figure 5.1.



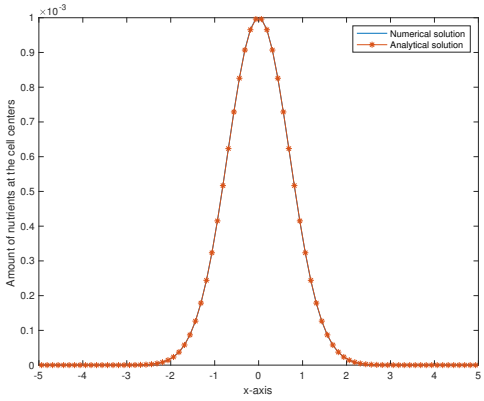
(a) Step 1



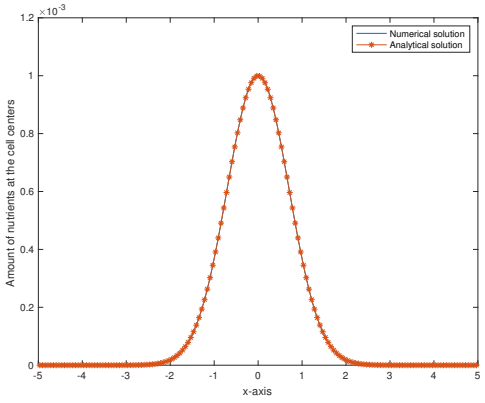
(b) Step 2



(c) Step 3



(d) Step 4



(e) Step 5

Figure 5.1: Plots of the analytical and numerical solution given different values for Δx and Δt. The total time interval is T=0.01 and the total length of the domain is L=10. Each plot correspond to one row in Table 5.1.

The reductions in Table 5.1 is calculated by using the following relationship;

$$\text{Reduction} = \frac{e_{i-1}}{e_i}. \tag{5.8}$$

Here, i represent the row number where the reduction value is given and $i-1$ represent the previous row number.

From Table 5.1, we see that the error gets smaller as Δx decreases. We also see from the plots in Figure 5.1 that the analytical and numerical solution get closer to the bell curve as we increase the number of steps. From this we can conclude that the solution converges.

The next equation we tested in 1D was $C_n = t \sin x$. By using this equation for C_n we get the reaction term;

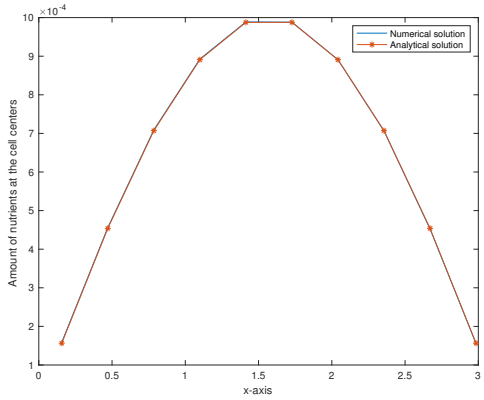
$$R_n = \sin x + t(\sin x + \cos x). \tag{5.9}$$

We used the interval from 0 to π in the x-direction. The y- and z-direction is still 1 and the total time interval is T=0.01. The results can be seen in Table 5.2.

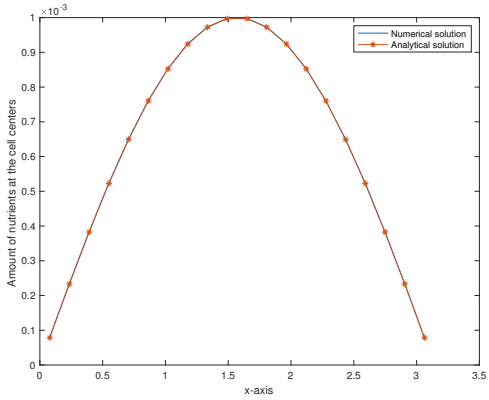
Table 5.2: Results from the analytical solution in 1D when R_n is given by (5.9), T=0.01 and the interval goes from 0 to π.

	Δx	Δt	L_2 error	Reduction
1	0.3142	1e-3	2.7462e-06	-
2	0.1571	0.5e-3	2.0805e-06	1.3200
3	0.0785	0.25e-3	1.7426e-06	1.1939
4	0.0393	0.125e-3	1.5724e-06	1.1082
5	0.0196	0.0625e-3	1.4870e-06	1.0574

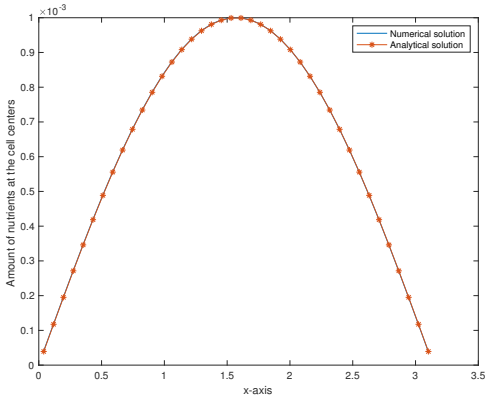
The plots comparing the analytical and numerical solution for the five cases in Table 5.2 can be seen in Figure 5.2.



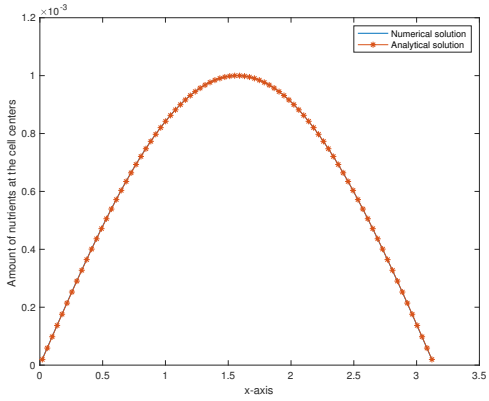
(a) Step 1



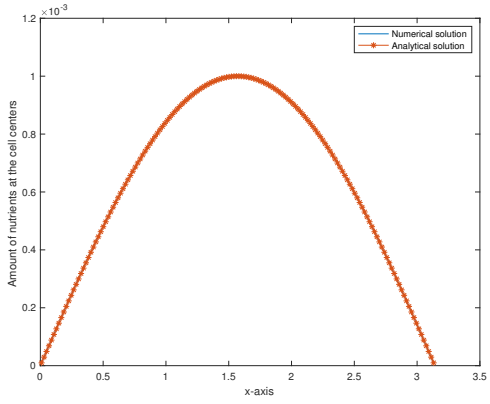
(b) Step 2



(c) Step 3



(d) Step 4



(e) Step 5

Figure 5.2: Plots of the analytical and numerical solution given different values for Δx and Δt. T=0.01 and L=π. Each plot correspond to one row in Table 5.2.

Also here we see that the error in Table 5.2 gets smaller and that the numerical and analytical solution in Figure 5.2 gets smoother as we increase the number of steps in the x-direction.

Ideally we would have wanted the reduction to be closer to 4. The reason we do not get the convergence rate we want, is probably because we have not implemented any specific boundary conditions for the analytical solution, we have kept the boundary conditions from our numerical model. For the numerical model and the analytical solution we have used the default boundary conditions in MRST, which is no-flux boundary conditions on all the sides of the domain where boundary conditions are not defined specifically. We did not aim to change the default boundary conditions since we get convergence without modifying the boundary conditions in 1D.

5.1.2 2D

To test our implementation in 2D we used the equation $C_n = te^{-(x^2+y^2)}$. Again, calculating the terms in Equation (5.2) gives the source/sink term

$$R_n = e^{-(x^2+y^2)}(1 - 2t(2x^2 + 2y^2 + x + y - 2)). \tag{5.10}$$

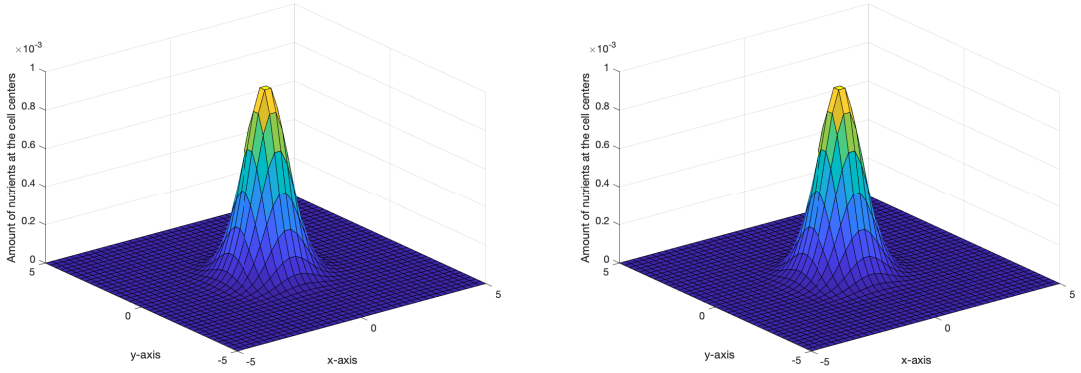
To do our simulation for the analytical and numerical solution in 2D we use a domain with length L=10 and width W=10 in the x- and y- direction respectively. The core has length 1 in the z-direction. Also in the 2D case we use T=0.01.

The results from the analytical solution in 2D can be seen in Table 5.3.

Table 5.3: Results for analytical solution in 2D when L=10, W=10 and T=0.01. R_n is given by Equation (5.10).

	Δx	Δy	Δt	L_2 error	rate of convergence
1	1	1	1e-3	4.9158e-07	-
2	0.5	0.5	0.5e-3	2.7400e-07	1.7941
3	0.25	0.25	0.25e-3	6.0221e-08	4.5498
4	0.125	0.125	0.125e-3	1.3742e-08	4.3823
5	0.0625	0.0625	0.0625e-3	3.2568e-09	4.2194

Figure 5.3 shows the plots of the analytical and numerical solution when $\Delta t = 0.25e-3$ and $\Delta x = \Delta y = 0.25$.



(a) Analytical solution **(b)** Numerical solution

Figure 5.3: Plots of the analytical and numerical solution in 2D. Here $\Delta x = \Delta y = 0.25$ and $\Delta t = 0.25e-3$. The core has length L=10 and width W=10. The plots represent the results from column 3 in Table 5.3.

In 2D we have used a flux vector that is zero in the y-direction and constant in the x-direction. This is also what we have simulated in our numerical model. By doing this, we have eliminated the problem with the no-flux boundary conditions given as a default in MRST. We want a reduction of 4 when we half Δx and Δy. The reduction results in Table 5.3 shows us that we get the reduction we want.

From the error analysis in Table 5.3 we see that the error gets smaller when the number of steps in the x- and y-direction increase. Looking at the plots in Figure 5.3 we can see that the plot for the analytical solution and the plot for the numerical solution are very similar. From this, we can draw the conclusion that our model is implemented correctly in 2D as well.

5.2 Benchmark example

We also want to test our implementation against a benchmark example. The results we have used as a reference is the results from the one-dimensional single phase

bioclogging problem in Li et al. (2010). The problem in Li et al. (2010) is a recreation of the numerical experiment in Kim (2006) which again is calibrated with the experimental data from Hendry et al. (1997).

In our model we consider nutrients being injected into the system. In the example from Kim (2006) and Li et al. (2010) they considered bacteria being injected. To do this simulation we have adapted our code to consider the concentration of bacteria. We have also implemented the whole model for reversible and irreversible attachment of bacteria explained in Kim (2006) and in Li et al. (2010).

The experiment in Hendry et al. (1997) considered a 40 cm long column with injection of artificial ground water from one side. They injected bacteria together with the water for 38.4 hours and then stopped the injection and continued the simulation for a total of 500 hours. The aim of the experiment was to examine the transport and sorption behavior through the column.

The following initial conditions and boundary conditions were used in the experiment (Kim, 2006):

$$C_b(x,0) = \sigma_b^1(x,0) = \sigma_b^2(x,0) = 0, \tag{5.11}$$

$$-D\frac{\partial C_b}{\partial x}(0,t) + v_w C_b(0,t) = \begin{cases} v_w C_{b0} & \text{at } 0 < t \leq t_0 \\ 0 & \text{at } t > t_0 \end{cases} \tag{5.12}$$

$$\frac{\partial C_b}{\partial x}(L,t) = 0. \tag{5.13}$$

Here, C_{b0} is the influent bacterial concentration, L is the length of the column and t_0 is the duration of bacterial injection. The parameters we have used are listed in Table 5.4 and the result from our simulation can be seen in Figure 5.4.

Table 5.4: Model parameters used in the benchmark simulation.

Parameter		Value
C_{b0}	Influent concentration of bacteria ($mg\ l^{-1}$)	4.32
L	Column length (cm)	40
ϕ_0	Porosity	0.4
u_w	Pore-water velocity ($cm\ s^{-1}$)	2.17×10^{-4}
$\alpha_{b,L} = \alpha_{b,T}$	Dispersivity (cm)	0.27
v_g	Settling velocity of bacteria ($cm\ s^{-1}$)	1.61×10^{-5}
ρ_b	Density of bacteria ($mg\ l^{-1}$)	1.085×10^{6}
k_1	Reversible attachment rate coefficient (s^{-1})	2.28×10^{-5}
k_2	Detachment rate coefficient (s^{-1})	3.56×10^{-7}
k_3	Irreversible detachment rate coefficient (s^{-1})	1.72×10^{-6}
g_1	Bacterial growth rate coefficient (s^{-1})	0
d_1	Bacterial decay rate coefficient (s^{-1})	0

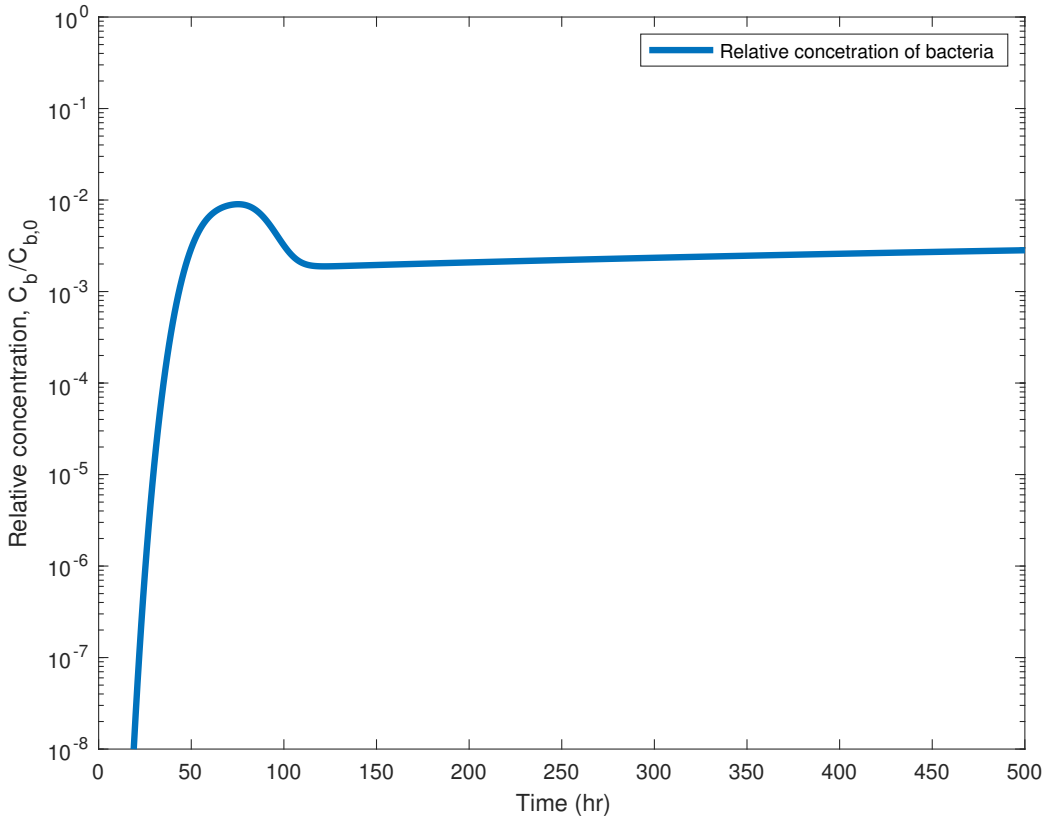


Figure 5.4: Results from our recreation of the numerical study in Li et al. (2010).

If we compare our result in Figure 5.4 with the results in Li et al. (2010) and in Kim (2006), we see that our simulation is in agreement with the result in both articles.

5.3 Viscosity reduction

The viscosity of a fluid is the measure of its resistance to flow. The lower the viscosity, the easier the fluid flows. In our system, some of the gas produced will mix with the oil. In the numerical example described in Section 4.5, the oil viscosity is constant. In this section we want to look at how the viscosity can affect the oil saturation if the viscosity is a function of the gas produced.

We want the viscosity to decrease when the concentration of gas increase. To model this, we use heuristic equations that give the relationship we want between viscosity and gas.

To see how the change in viscosity affect the oil saturation we compared how much of the initial oil in the reservoir is produced when the viscosity is constant and when the viscosity is modeled as a function of the gas. We do this by plotting the percentage of the initial oil saturation produced against the pore volume. One pore volume is when the void space in the porous medium is filled one time going from totally empty to full. The pore volume can be expressed as

$$\text{PV}= \frac{\text{Rate}\cdot\text{Time}}{\text{Porosity}\cdot\text{Volume}},$$

where the rate is the rate with which the water is injected into the system, porosity is the initial porosity of the porous medium, volume is the volume of the whole core and time is the time elapsed from the start of the simulation.

To calculate the percentage of initial oil saturation produced, $S_{o,per}$, at each time step we use the equation

$$S_{o,per} = \left(\frac{1}{N} \sum_{i=1}^{N} \frac{S_{o,0}(\phi_0 - \sigma_0) - S_{o,i}(\phi_0 - \sigma_i)}{S_{o,0}(\phi_0 - \sigma_0)} \right) \cdot 100. \tag{5.14}$$

Here we sum over the saturations in all the cells at each time step and then divide the sum by the number og cells, N. In Equation (5.14), $S_{o,0}$ is the initial oil saturation in

the reservoir, ϕ_0 is the initial porosity and σ_0 represent the initial volumetric fraction of biofilm in the core. $S_{o,i}$ and σ_i represent oil saturation and the volumetric fraction of biofilm at time step i respectively.

When simulating the viscosity we need to define a minimum value for the oil viscosity in the core. This is because the gas reduce the oil viscosity but at some point the gas cannot reduce it further. Purwwasena et al. (2009) found that the oil viscosity can be reduced by 40% of the original viscosity as a consequence of bacteria growth. Therefore, we use the relationship $\mu_{o,min} = 0.6 \cdot \mu_{o,org}$ when we simulate the viscosity reduction. We have used 1000 time steps to do the simulations where each time step is 20 seconds long. The size and dimensions of the core are the same as in Table 4.3.

The first equation we test for the viscosity is the one in Equation (5.15). The goal is to see how the relationship between the percentage of oil saturation recovered at each time step change when we vary the value of the arbitrary constant β.

$$\mu_o(C_g) = (1 - \frac{C_g}{\beta + C_g}) \cdot (\mu_o - \mu_{o,min}) + \mu_{o,min}. \tag{5.15}$$

In (5.15), C_g denotes the gas, μ_o is the oil viscosity at the given time and $\mu_{o,min}$ is the minimum possible oil viscosity. We have tested the scenarios when $\beta = 0.5\ kg\ m^{-3}$, $\beta = 1\ kg\ m^{-3}$, $\beta = 100\ kg\ m^{-3}$ and $\beta = 500\ kg\ m^{-3}$. These constants are chosen randomly but we wanted to test a range of values and that is why we have chosen a big gap between the values.

The results of how the relationship in Equation (5.15) affects the oil production given different values of β can be seen in Figure 5.5. We compare the results to the result with constant viscosity.

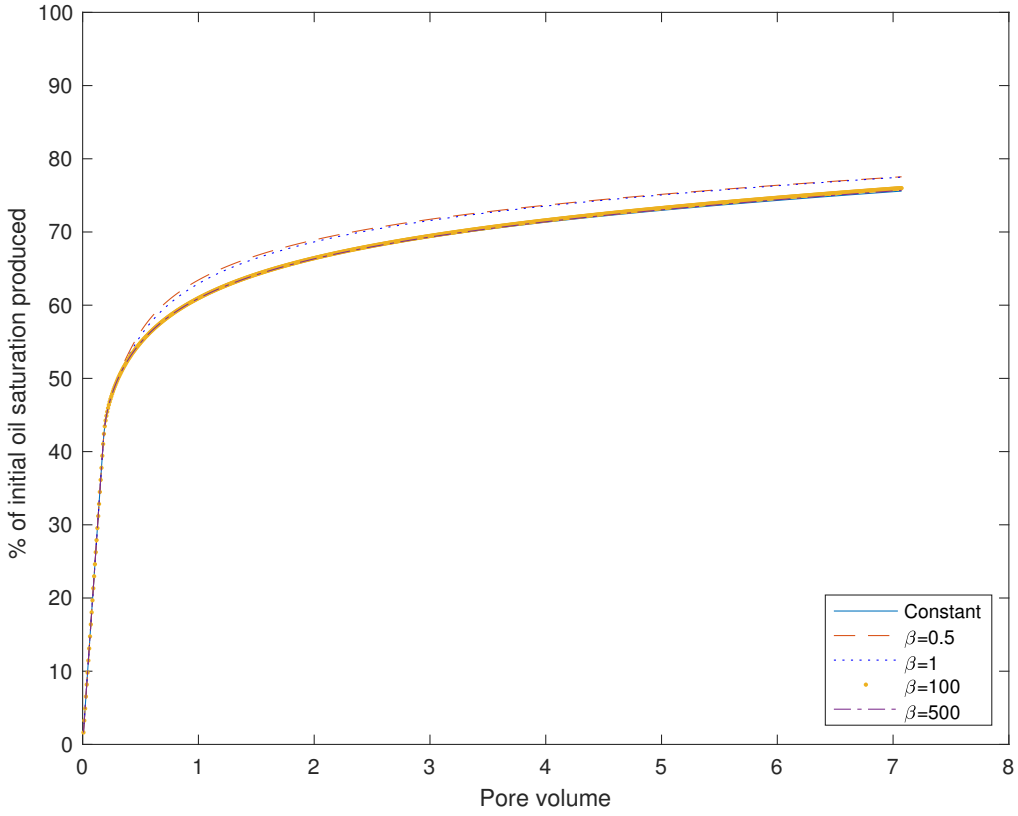


Figure 5.5: Percentage of the initial oil saturation produced against the pore volume when μ_o is given by (5.15). The different graphs represent different values of β.

We see from the plot that this relationship for the viscosity does not make a great impact on the percentage of initial oil saturation produced. We have also tested the equation for other values of β but the results were similar to the results in Figure 5.5. Therefore the results are not included here.

The next equation we tested was the exponential function given in (5.16).

$$\mu_o(C_g) = \mu_{o,min} e^{-\alpha C_g}. \tag{5.16}$$

Here, we have tested the cases where $\alpha = 0.1$, $\alpha = 0.3$, $\alpha = 0.5$ and $\alpha = 0.6$. Also here we compared the result with the percentage of initial oil saturation produced when the viscosity is constant. The results can be seen in Figure 5.6.

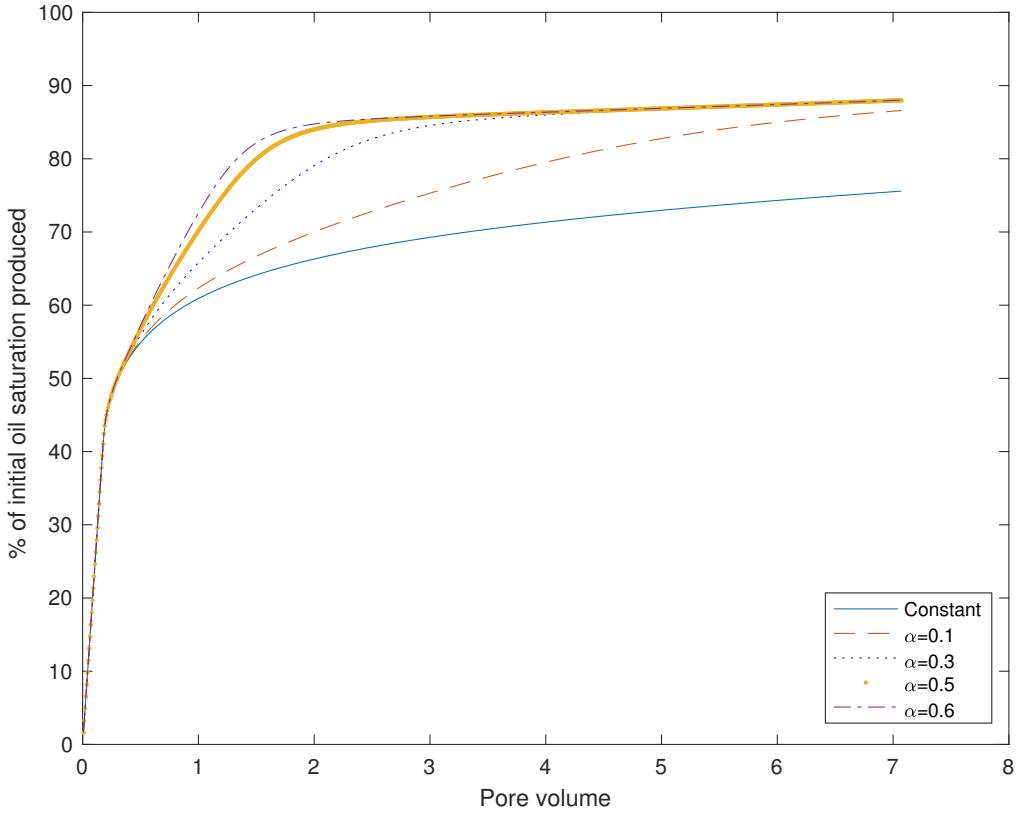


Figure 5.6: Percentage of the initial oil saturation in the reservoir produced against the pore volume. Here μ_o si given by equation (5.16) and the different graphs represent different values of α.

We see from Figure 5.6 that there is a big impact on the oil recovery if the viscosity is modeled by the exponential function given in Equation (5.16). The graph flattens out just before 90% of the initial oil saturation is recovered. This is the point where the oil saturation reaches the level of the residual oil saturation.

Out of the two cases we have tested it is clear that the best relationship to use to simulate viscosity is the exponential function. If we compare the mean of the y-values for the graphs with the best results in Figure 5.5 and Figure 5.6, we see that modeling the viscosity with Equation (5.16) increase the percentage of initial oil saturation produced by approximately 11.6% compared to the results from modeling the viscosity with Equation (5.15).

5.4 Positioning of the biofilm

In addition to testing the affect of modeling the viscosity as a function of the gas produced, we have looked at the importance of the placement of biofilm originally in the system. In order to compare the different scenarios of biofilm distribution, it is important to use the same total volumetric fraction in the core. The reason being that the amount of biofilm in the system will affect the oil production.

We have compared five different cases; biofilm placed at the bottom or top of the domain, in every other cell of the domain, at the end of the domain, at the beginning of the domain and biofilm randomly distributed in the domain.

Originally, the biofilm was distributed evenly in every cell of our domain with a volumetric fraction of 0.1 in each cell. This is equivalent to 25% of the porosity in each cell. Figure 5.7 illustrates the core with even distribution of biofilm.

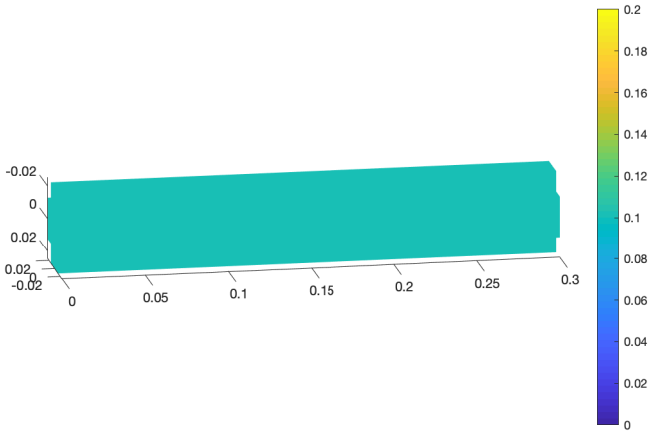
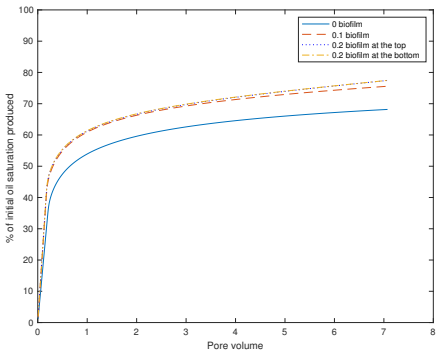


Figure 5.7: Biofilm evenly distributed in the domain with a volumetric fraction of 0.1 in every cell.

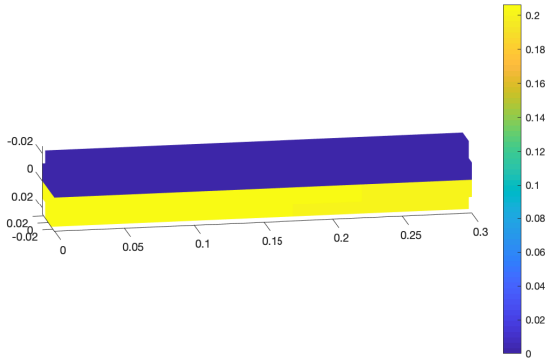
To compare the results from the different biofilm distributions we plot the pore volume against the percentage of initial oil saturation produced, same as we did for the viscosity reduction. All the results are compared to the case with zero biofilm in the system and the original case were the biofilm takes up a volumetric fraction of 0.1 in every cell.

First, we have looked at the case with biofilm at the top half and bottom half of

the core. Every cell on the top half or bottom half of the domain has a volumetric fraction of 0.2 containing biofilm at the beginning of the simulation. This ensures that it is the same amount of biofilm initially in the system as in the original case. The results of this comparison can be seen in Figure 5.8.



(a) The % of initial oil saturation produced when the biofilm is located at the top and bottom of the domain, as showed in Figure 5.8b, compared to even distribution of biofilm and zero biofilm in the system.

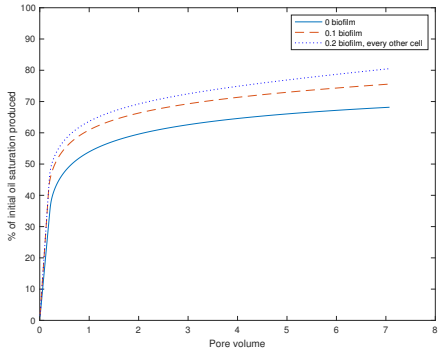


(b) Illustration of the core when the biofilm is concentrated at the bottom half of the domain.

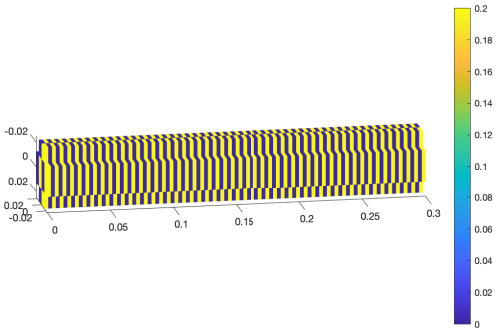
Figure 5.8: Biofilm at the top and bottom half of the domain.

We see from Figure 5.8 that it makes a very small difference on the oil production if we distribute the biofilm evenly in every cell or if we locate it on the top or bottom of the domain.

Next, we have tested placing the biofilm in every other cell of the domain. Also here the biofilm has a volumetric fraction of 0.2. The results can be seen in Figure 5.9.



(a) The % of initial oil saturation produced when the biofilm is located in every other cell, as showed in Figure 5.9b, compared to even distribution of bioiflm and zero biofilm in the system.

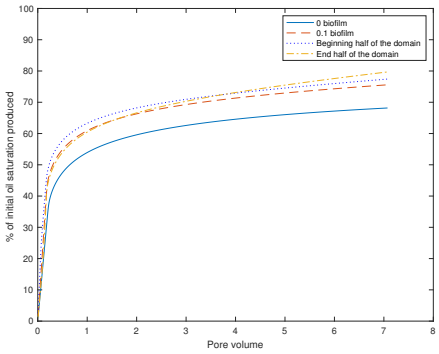


(b) Illustration of the core when the biofilm is concentrated in every other cell of the domain.

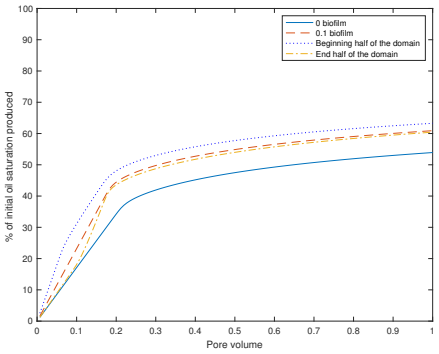
Figure 5.9: Biofilm in every other cell of the domain.

We see from Figure 5.9a that the oil production is increased when the biofilm is located with a volumetric fraction of 0.2 in every other cell instead of being distributed with a volumetric fraction of 0.1 in every cell. The total volumetric fraction of biofilm is still 25% of the initial porosity.

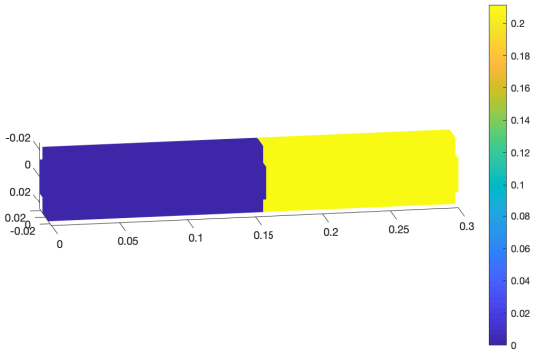
We also tested placing the biofilm at the beginning half of the domain and at the end half of the domain. The graph in Figure 5.10 shows the comparison of the percentage of initial oil saturation produced when the biofilm is evenly distributed in the core, when there is zero biofilm in the core and when the biofilm is concentrated at the beginning and end of the domain.



(a) The % of initial oil saturation produced when the biofilm is located in the beginning and end half of the domain, as showed in Figure 5.10c, compared to even distribution of biofilm and zero biofilm in the system.



(b) Zoomed in on figure 5.10a to see the difference between the cases better.



(c) Illustration of the core when the biofilm is concentrated at the end half of the domain.

Figure 5.10: Biofilm at the beginning half and end half of the domain.

We can see from the plots in Figure 5.10 that the oil is recovered faster when the biofilm is at the beginning of the domain. If the biofilm is located at the end half of the domain, there will be more oil recovered at the end of the simulation. This can be seen in Figure 5.10a.

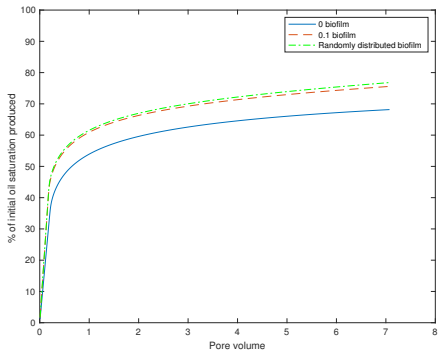
Last but not least, we tested how random distribution of the biofilm will affect the system. In order to distribute the biofilm randomly we used the randfixedsum.m function by Stafford (2006). This is a function in MATLAB that lets you decide the

size of the matrix you want to generate, the sum of the random numbers and the minimum and maximum value the random numbers can have. In our case, we have generated a 1188×1 vector where the sum of the entries have to be 118.8, which is the total volumetric fraction of biofilm we want in our system. In addition, we have chosen that the values should lay between 0 and 0.2. 0.2 is chosen so that the porosity in each cell can have a minimum value of 0.2. This comes from Equation (3.11);

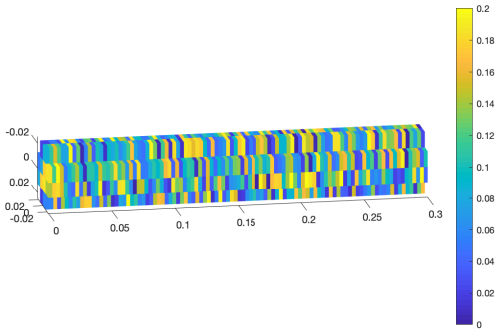
$$\phi = \phi_0 - \sigma,$$

where, as mentioned earlier, σ is the volumetric fraction of the biofilm. The initial porosity in our system is 0.4 so if we choose the maximum value of the random numbers to be 0.4 or greater we risk having cells with zero porosity or negative porosity.

The results with random distribution of biofilm can be seen in Figure 5.11.



(a) The % of initial oil saturation produced when the biofilm is randomly located in the domain, as showed in Figure 5.11b, compared to even distribution of biofilm and zero biofilm in the system.



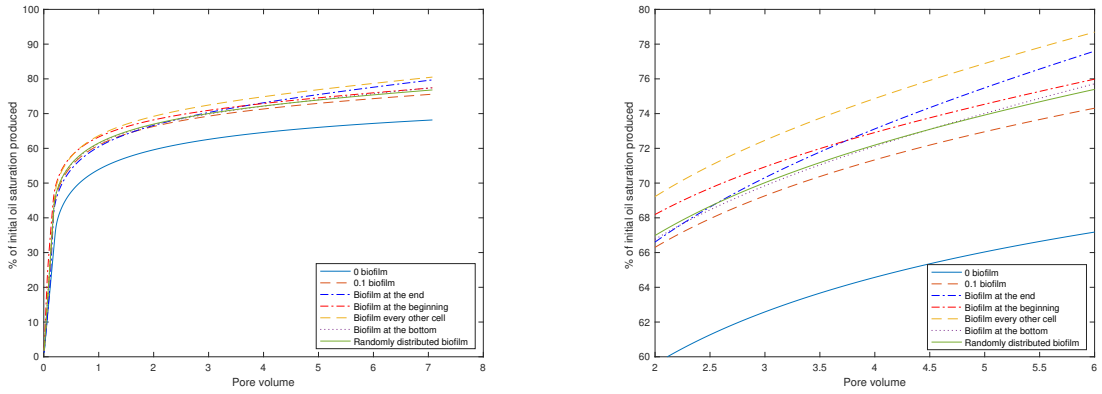
(b) Illustration of the core when the biofilm is randomly distributed in the domain.

Figure 5.11: Biofilm randomly distributed in the core.

We observe that there is a small improvement on the oil recovery when we distribute the biofilm randomly in the core then when we distribute it evenly. Both of them are significantly better than without biofilm.

Finally, we have compared all the different scenarios for biofilm placement to see

which one has the greatest impact on the amount of oil recovered. The results can be seen in Figure 5.12.



(a) Comparing how the different biofilm distributions affect the oil production.

(b) Zoomed in on Figure 5.12a to see the difference between the scenarios better.

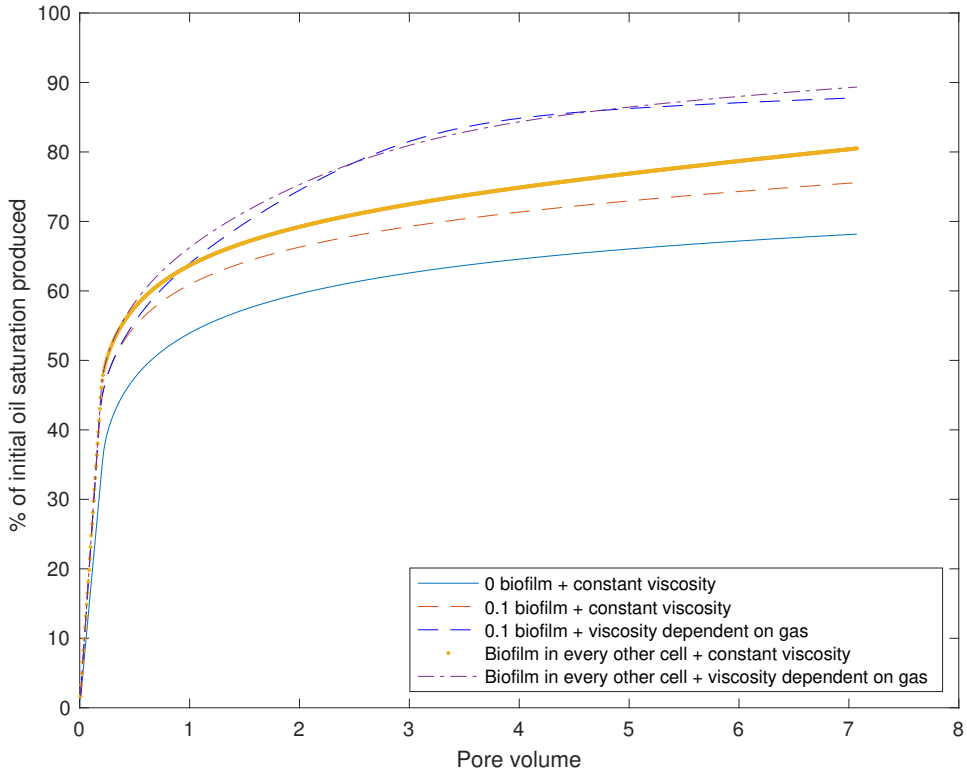
Figure 5.12: Comparing all the results of the different biofilm distributions.

We can see from the plot in 5.12a that we obtain the best results when the biofilm is located in every other cell of the domain.

5.5 Positioning of the biofilm combined with viscosity reduction

Now that we have looked at the effects of modeling oil viscosity as a function of the gas produced and how the positioning of the biofilm affects the oil recovery, we want to combine the two cases and see if we can improve the oil recovery even more.

Our studies in Section 5.3 shows that we obtain the best oil recovery when modeling the oil viscosity with the exponential function in Equation (5.16). In Section 5.4, our research shows that we obtain best oil recovery by positioning the biofilm in every other cell of the domain. In this section we have combined these two scenarios. The result can be seen in Figure 5.13.



(a) Results from combining the viscosity reduction due to gas and positioning of the biofilm. Here we compare the system with zero biofilm and constant oil viscosity, biofilm located in every cell with constant oil viscosity, biofilm located in every other cell with constant oil viscosity, biofilm in every other cell and with viscosity modeled with the exponential function given in Equation (5.16) and biofilm located in every cell combined with viscosity modeled with Equation (5.16).

Figure 5.13: Combining the affect of viscosity reduction due to gas, the different locations of biofilm and the combinations of the two scenarios.

We chose to use $\alpha = 0.2$ as the arbitrary constant in Equation (5.16) to do this comparison. This is because we will reach the residual oil saturation before our simulation is done if we choose a higher value for α. In Figure 5.13, we have compared five different scenarios. The first one is the case where we have zero biofilm and constant oil viscosity. Next we have biofilm located in every cell with constant oil viscosity. Then we have two cases with biofilm located in every other cell. One of them is with constant viscosity and the other one is with viscosity modeled with the exponential function given in Equation (5.16). Lastly, we compare the case with

biofilm in every cell combined with viscosity modeled with Equation (5.16).

From Figure 5.13 we see that the case where the biofilm is located in every other cell of the core combined with oil viscosity modeled by the exponential function in Equation (5.16), with $\alpha = 0.2$, leads to most oil recovery.

Chapter 6

Conclusion and future work

In this Thesis we have studied a two-phase flow system where we have used biofilm to preform bioclogging. We have simulated a system containing of oil, water, nutrients, biofilm and gas. To do the simulations we have used MATLAB and the MRST package from SINTEF. The equations we have used are transport equations to simulate the nutrients and gas in the system and mass conservation equations to model the two phases, water and oil. To model the biofilm we have used the equation for reversible deposited bacteria proposed by Kim (2006).

The implementation of the model has been tested using analytical solutions and a benchmark example from Li et al. (2010). The results from these tests indicate that our implementation is done correctly.

A big part of this Thesis was to see how we could manipulate the system to recover more oil. We have tested two scenarios; the first one was to model the oil viscosity as a function of the gas produced in the system and the second one was how the positioning of the biofilm affects the oil recovery. We can see from the results in Section 5.3 that using the exponential function given in Equation (5.16) leads to a significant improvement in the amount of oil being produced. The results obtained from the different distributions of the biofilm can be seen in Section 5.4. Our studies shows that the fastest oil recovery is obtained when the biofilm is located at the beginning half of the domain but we recover the most oil when the biofilm is located in every other cell of the domain. In addition to testing these two cases separately,

we have combined them and our results show that the most amount of oil will be recovered if the biofilm is located in every other cell of the domain and the oil viscosity is modeled by the exponential function given in Equation (5.16).

Our goal for this Thesis was to implement a mathematical model for MEOR due to biofilm formation and gas production. We also wanted to examine how viscosity modeled as a function of the gas produced in the core would affect the oil recovery. We have succeeded with this but there are of course future work that can be done in order to make the model even better.

Future work

Ideally we would have wanted to compare our model against experimental data. This would have given us a better understanding of how realistic our model is. Also, if we could have used laboratory results to calibrate our parameters we might have gotten a different result when it comes to the viscosity reduction due to gas production.

The gas in our model is not included as a third phase. To improve the model, this is one thing that should be done.

Next we suggest scaling up the core and adding injection wells and production wells. By adding an injection well the flow of water would not be injected from the whole left-hand side of the domain, it would be injected from a smaller cross-sectional area. This would probably effect our results.

When modeling oil recovery mathematically, we do a lot of simplifications and assumptions. These simplifications and assumptions will of course affect the accuracy of the results. We would recommend to make the model more complex and also use it to simulate other possible scenarios when it comes to MEOR.

All these suggestions for future work would help make the model more complex and, in that way, also more realistic.

Bibliography

Bao, K., Lie, K.-A., Møyner, O., and Liu, M. (2017). Fully implicit simulation of polymer flooding with mrst. *Computational Geosciences*, 21(5):1219–1244.

Bear, J. (1988). *Dynamics of FLuids in porous Media*. Dover books on physics and chemistry. Dover, New York.

Brown, L. R. (2010). Microbial enhanced oil recovery (meor). *Current Opinion in Microbiology*, 13(3):316–320. doi: https://doi.org/10.1016/j.mib.2010.01.011.

Dullien, F. A. L. (1979). *Porous Media, Fluid Transport and Pore Structure*. Academic press, New York, 396p.

Ekt Interactive (2010). History of oil. `https://www.ektinteractive.com/history-of-oil/`, (Visited May 2,2019).

Goda, H. M. and Behrenbruch, P. (2004). Using a modified brooks-corey model to study oil-water relative permeability for diverse pore structures. *SPE Asia Pacific Oil and Gas Conference and Exhibition*, page 14. doi: https://doi.org/10.2118/88538-MS.

Hendry, M. J., Lawrence, J. R., and Maloszewski, P. (1997). The role of sorption in the transport of klebsiella oxytoca through saturated silica sand. *Groundwater*, 35(4):574–584. doi: https://doi.org/10.1111/j.1745-6584.1997.tb00122.x.

Iserles, A. (2009). *A first course in the numerical analysis of differential equations*. Cambridge texts in applied mathematics. Cambridge University Press, Cambridge, 2nd ed. edition.

Kim, S.-B. (2006). Numerical analysis of bacterial transport in sat-

urated porous media. *Hydrological processes*, 20(5):1177–1186. doi: https://doi.org/10.1002/hyp.5930.

Kincaid, D. (2002). *Numerical analysis : mathematics of scientific computing*, volume 2 of *The Sally series, Pure and applied undergraduate texts*. American Mathematical Society, Providence, R.I, 3rd ed. edition.

Landa-Marbán, D. (2016). Modeling and simulation of microbial enhanced oil recovery: A new approach which includes the role of interfacial area. Master's thesis. University of Bergen, Norway.

Landa-Marbán, D., Liu, N., Pop, I. S., Kumar, K., Pettersson, P., Bødtker, G., Skauge, T., and Radu, F. A. (2019). A pore-scale model for permeable biofilm: Numerical simulations and laboratory experiments. *Transport in porous Media*, 127(3):643–660. doi: https://doi.org/10.1007/s11242-018-1218-8.

Landa-Marbán, D., Radu, F. A., and Nordbotten, J. M. (2017). Modeling and simulation of microbial enhanced oil recovery including interfacial area. *Transport in Porous Media*, 120(2):395–413. doi: 10.1007/s11242-017-0929-6.

Li, J., Liu, J., Trefry, M. G., Park, J., Liu, K., Haq, B., Johnston, C. D., and Volk, H. (2010). Interactions of microbial-enhanced oil recovery. *Transport in Porous Media*, 87(1):77–104. doi: https://doi.org/10.1007/s11242-010-9669-6.

Li, K. and Horne, R. N. (2006). Comparison of methods to calculate relative permeability from capillary pressure in consolidated water-wet porous media. *Water Resources Research*, 42(6). doi: https://doi.org/10.1029/2005WR004482.

Lie, K.-A. (2016). *THE SURFACTANT MODEL IN MRST*. Documentation in the MRST package. Go to modules - ad-eor - docs.

Lie, K.-A. (2019). *An Introduction to Reservoir Simulation Using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST)*. Cambridge University Press.

Nordbotten, J. M. and Celia, M. A. (2012). *Geological Storage of CO2: Modeling approaches for large-scale simulation*. John Wiley and Sons, Inc., Hoboken, NJ, USA.

Patel, J., Borgohain, S., Kumar, M., Rangarajan, V., Somasundaran, P.,

and Sen, R. (2015). Recent developments in microbial enhanced oil recovery. *Renewable and Sustainable Energy Reviews*, 52:1539–1558. doi: https://doi.org/10.1016/j.rser.2015.07.135.

PetroWiki (2016). Petrowiki. `https://petrowiki.org/Capillary_pressure`, (Visited March 21,2019).

Pettersen, Ø. (1990). *Grunnkurs i reservoarmekanikk*. Matematisk institutt, Universitetet i Bergen c1990, Bergen.

Purwwasena, I. A., Sugai, Y., and Sasaki, K. (2009). Estimation of the potential of an oil-viscosity-reducing bacteria, petrotoga isolated from an oilfield for meor. International Petroleum Technology Conference. doi: https://doi.org/10.2523/IPTC-13861-MS.

Safdel, M., amin Anbaz, M., Daryasafar, A., and Jamialahmadi, M. (2017). Microbial enhanced oil recovery, a critical review on worldwide implemented field trials in different countries. *Renewable and Sustrainable Energy reviews*, 74:159–172. doi: https://doi.org/10.1016/j.rser.2017.02.045.

Sen, R. (2008). Biotechnology in petroleum recovery; the mocrobial eor. *Progress in Energy and Combustion Science*, 34(6):714–724. doi: https://doi.org/10.1016/j.pecs.2008.05.001.

Skiftestad, K. (2015). Numerical modelling of microbial enhanced oil recovery with focus on dynamic effects: An iterative approach. Master's thesis. University of Bergen, Norway.

Stafford, R. (2006). Random vectors with fixed sum. `https://se.mathworks.com/matlabcentral/fileexchange/9700-random-vectors-with-fixed-sum?focused=5064802tab=function`, (Visited May 14,2019).

Wood, D. A. (2019). Microbial improved and enhanced oil recovery (mieor): Review of a set of technologies diversifying their applications. *Advances in Geo-Energy Research*, 3(2):122–140. https://doi.org/10.26804/ager.2019.02.02.

Youssef, N., Elshahed, M. S., and Mclnerney, M. J. (2009). Microbial processes in oil fields: Culprits, problems, and opportunities. *Advances in Applied Microbiology*, 66:141–251. doi: https://doi.org/10.1016/S0065-2164(08)00806-X.

Zobell, C. E. (1946). Action of microörganisms on hydrocarbons. *Microbiology and Molecular Biology Reviews*, 10(1-2):1–49.