University of Stavanger

Faculty of Science and Technology

MASTER'S THESIS

Study program/ Specialization: Petroleum technology Drilling engineering	Spring semester, 2012 Open / ~~Restricted access~~
Writer: Yasir Akdeniz	………………………………………… (Writer's signature)
Faculty supervisor: Kjell Kåre Fjelde External supervisor(s):	
Title of thesis: **Making a simple steady state model more appropriate for underbalanced drilling design**	
Credits (ECTS): 30	
Key words: • Underbalanced drilling • Flow models • Steady-state model	Pages: 57 + enclosure: CD Stavanger, 16.05.2012 Date/year

Making a simple steady state model more appropriate for underbalanced drilling design

Preface

This master thesis is a result of my Masters education in Drilling engineering at the University of Stavanger.

I would like to express gratitude to my supervisor Kjell Kåre Fjelde for guiding me through with my thesis.

Yasir Akdeniz

2012

Contents

List of figures

List of tables

Abstract

In an underbalanced operation the well needs to be kept in underbalanced conditions at all time. That is, from the beginning to the end of the drilling process. This is necessary to keep well from formation damage and potential hazardous drilling problems such as lost circulation and differential sticking.

With flow modeling, it is possible to simulate an underbalanced drilling operation scenario, with given liquid and gas injection rates, and also production influx.

The concept of steady state modeling is that the amount of mass going in to the system, is the same amount of mass coming out of it.

By using Matlab programming language, and the implementation of the numerical method of “Halving the interval”, it has been possible to find the bottomhole pressure of a underbalanced well with guessing. The guessing has been necessary to start the whole looping process with the method of “Halving the interval”.

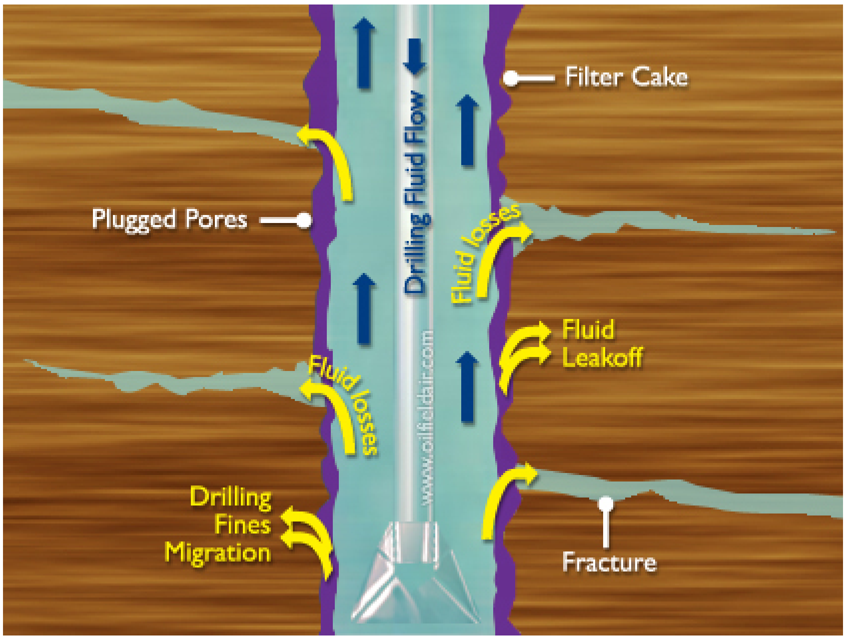
Some changes of liquid rates versus gas rates has been made to plot those values in to graphs, showing a trend for the various combination of rates, production influx and different choke pressures.

Introduction

Methods of drilling

Overbalanced drilling; the conventional way of drilling

The most common way of drilling a well today is by overbalance. This method is the way it has been drilled ever since the dawn of the ever developing petroleum industry. By drilling overbalanced, the bottom hole wellbore pressure is kept higher than the formation fluid pressure at all times while drilling the well. To keep the well overbalanced at all times requires adjustments of the mud weight during the whole drilling operation. The designed mud weight must be so that it is lower than the formation *fracture* pressure, but above the formation *pore* pressure.



Figur 1: Overbalanced drilling

Overbalanced drilling method can be favored upon certain situations like:

1. When there is a high risk of borehole collapse or the wellbore caving in due to insufficient support by the wellbore pressure.
2. If there is a salt formation nearby. Because salt behaves like plastic. And if the wellbore pressure would be lower than the formation pressure, the salt would flow towards the wellbore, creating an increased risk for the drilling operation.
3. When there is unconsolidated sand, particularly in horizontal wells, where it tends to slough into the wellbore and consequently cause drilling problems.

Too high mud weight can also fracture the formation, which can lead to lost circulation. One important aspect of the mud weight in overbalanced drilling is to design it so that it forms a good filter cake around the wellbore wall, so it prevents mud invasion into the formation.

Here is a list of the advantages and disadvantages when drilling overbalanced:

Advantages:

1. It’s a well-known drilling technique, thus one have well developed procedures due to its history of use.
2. The safety issues concerning it are very well known.
3. An overbalanced drilling operation requires fewer personnel.
4. It’s more economical.
5. Less rig space required.
6. Good borehole stability.
7. No need for handling of hydrocarbons during drilling.

Disadvantages:

1. Skin factor; potential damages to the formation.
2. Potential mud loss to the formation.
3. Low rate of penetration through harder formations.
4. Potential for differential sticking.
5. Potential for getting a kick in case of a section with unknown pore pressure.

Underbalanced drilling

Underbalanced drilling is a mode of rotary drilling that is carried out with a bottom hole wellbore pressure *lower* than the formation pore pressure. Recent technological advances, like improved planning, operational practices and enhanced crew training, has placed underbalanced drilling operations (UBO) in a very competitive position compared to overbalanced drilling. When candidate wells are selected and the operation is professionally planned and conducted, superior results are expected in drilling and production performance and in safety records. Underbalanced condition is held by keeping the drilling fluid such that the pressure column provides a hydrostatic pressure column lower than formation pore pressure. A wellbore pressure that is lower than the formation pore pressure will provide a potential for hydrocarbons to flow into the well.

A four phase separation facility is required as hydrocarbons are produced while drilling. The four phases separated are gas, light liquid (oil and condensate), heavy liquid (mud and water) and solids/cuttings.

One of the most important equipment related to underbalanced drilling, is the rotating blowout preventer (RBOP). This equipment is continuously closed around the drill string with the effect of rubber elements that rotates with the string. The packing elements provides a seal between rotating and stationary elements.

Underbalanced drilling requires more safety precautions than overbalanced drilling operations. The lack of a hydrostatic column that provides a bottom hole pressure greater than the formation fluid pressure does not mean that the underbalanced drilling method is riskier than the overbalanced drilling method.

Mud loss during overbalanced drilling can bring the well in underbalance. The well is in this situation vulnerable for a potential kick. The situation can then be a totally gas filled well, which is the worst case scenario.

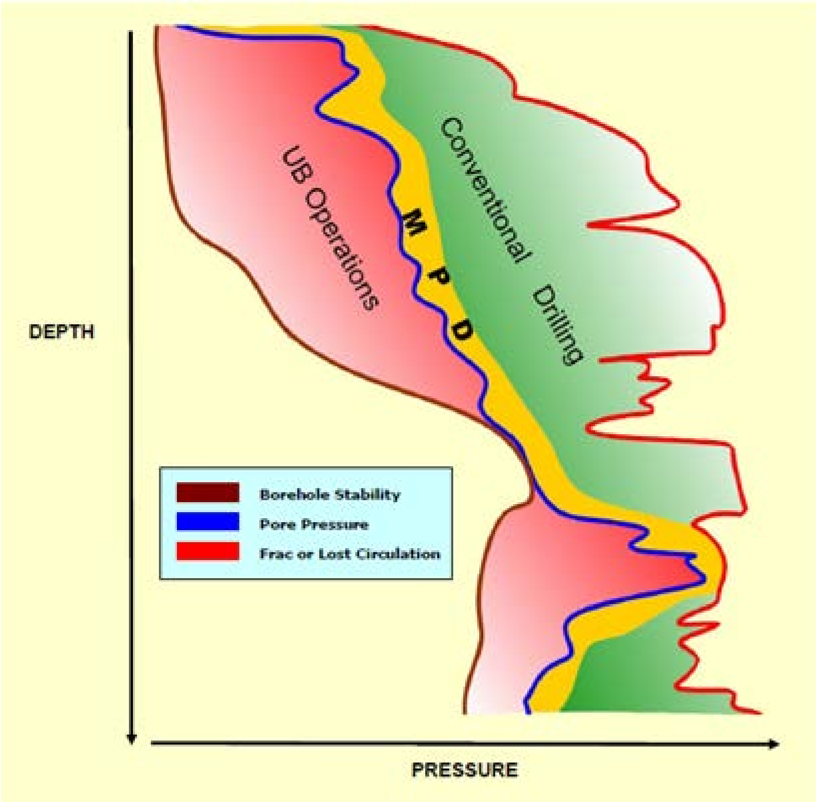
In underbalanced drilling this would not provoke a problem, because there is a surface arrangement to maintain a controlled inflow of gas/hydrocarbons. And a mud loss would not occur in the same manner and there would be better facilities to handle the inflow from formation.

UBO and well control are similar technologies. That is, the well is shut in against pressure in the wellbore and the well is circulated through a choke. Thus, the basic concepts of pressure in well control are also basic to UBO. The terms used in UBO may awake confusion, because they may be defined in a different manner.

For example, in UBO, simply using the term pressure is usually not precise enough because many kinds of pressure exist. To avoid confusion, a definition of terms used in UDO is of interest.

Examples of UBO include:

1. Drilling with air or gas. The pressure exerted by the column of gas and cuttings is almost always less than the formation pressure.
2. Drilling with cable tools. Because no mud column exists, cable-tool drilling is a form of UBO.
3. Drilling with any type of fluid where the pressure of the fluid column is less than formation pressure.



Figur 2: Drilling windows for Conventional drilling operations, Managed pressure drilling operations and Underbalanced drilling operations.

Formation pressure

Taken from [1], p. 1-2

Formation pressure is the force exerted by fluids in the formation. Field personnel often use *pore pressure* and *reservoir pressure* as synonyms for formation pressure. Pore pressure actually means the pressure of the fluids in the pores of the formation. Pore pressure often refers to the pressure of water within shale, or within sandstone. Formation pressure is the pressure in the formation at the wellbore. In the field, many use the term reservoir pressure interchangeably with formation pressure and pore pressure. Strictly defined, reservoir pressure refers to the actual fluid or gas pressure in a reservoir. While these terms have slightly different meanings, in this thesis, they mean the same.

In UBO, pressure is often expressed as the *equivalent mud weight (EMW)* necessary to balance the formation pressure. Whether pressure in a subsurface formation is expressed as formation pressure, pore pressure, or reservoir pressure, it is the force exerted by the fluid in the formation. In any formation, formation pressure may change over time as fluid is produced. This change is typically seen in a depleted formation where gas or oil has been withdrawn (or produced). Withdrawal results in lower formation pressure.

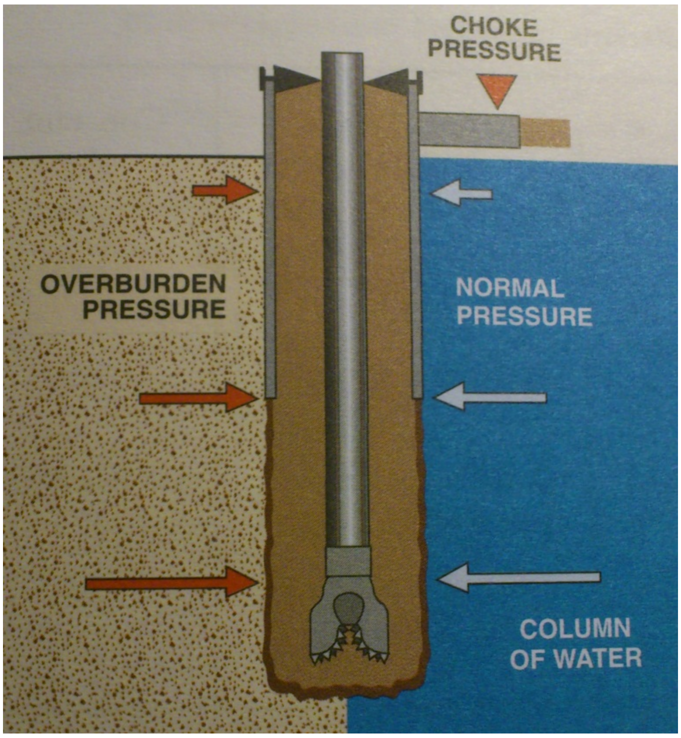
Figur 3: Formation pressure

By drilling deeper and deeper into the formation, the formation pressure will increase, according to the overburden pressure. Formation pressure varies with the depth of the hole and other geologic conditions. In the field, personnel can measure it by using *shut-in drill pipe pressure,* a *measurement-while-drilling (MWD)* recorder, or a *wireline pressure bomb.* Formation pressure in drilling operations can vary from near zero at the surface to pressures in excess of 140 000 kPa (1400 bar). There are 3 conditions for formation pressure, that is, it can be normal, subnormal, or abnormal.

Hydrostatic pressure

Taken from [1], p. 2-3

Hydrostatic pressure is the pressure exerted by the fluid in the wellbore. Drilling personnel can control this pressure, something which makes the study of hydrostatic pressure important. Hydrostatic pressure can vary from less than 689 kPa (6.89 bar, 1.20 kg/m^3) to 137 900 kPa (1379 bar, 2388 kg/m^3), in a super-pressured reservoir.



Figur 4: Pressures from the earth

In UDO the EMW window is often below 994.5 kg/m^3. However, exceptions occur. They occur in normally or abnormally pressured formations where operators use UBO to increase the drilling rate, avoid differential-pressure sticking, or avoid damaging the reservoir with filter cake or filtrate.

In drilling normally pressured formations, the hydrostatic pressure is usually controlled by increasing the drilling fluid density with salt or barite. The hydrostatic pressure may be decreased with oil, which is lighter than water.

With UDO in overpressured formations, the rig crew usually increases the hydrostatic pressure in the conventional manner using salt or barite. In normal and subnormal pressure formations, injecting air or gas into the drilling fluid decreases the hydrostatic pressure.

Uses of Underbalanced Drilling

Underbalanced drilling (UBO) requires more equipment and attention to the drilling process than conventional overbalanced drilling. To justify the extra cost of equipment and personell, a major economic advantage must exist for UBO. UBO must also be done in a geological climate where a maximum chance of success exists. Then finally, the drilling program must be set up to maximize the underbalanced the underbalanced advantage and minimize other drilling and completion problems. This section of the thesis will discuss the use of UBO, the proper geology or hole conditions for UBO, and the best drilling program for UBO.

The best uses for underbalanced drilling are to:

- Increase the drilling rate,
- Avoid or limit lost circulation
- Limit or avoid reservoir damage
- Reduce completion enhancement costs
- Avoid differential sticking
- Find masked potential reservoirs

Areas that fulfill several or all of these conditions include:

- Horizontal drilling
- Geothermal drilling
- Depleted reservoirs
- Old or poorly logged hydrocarbon producing sections.

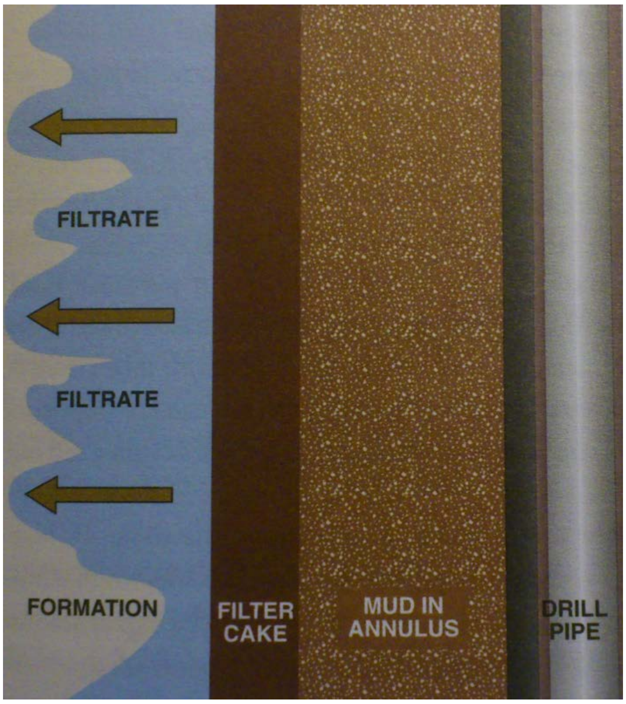
Increased drilling rate

Taken from [1], p. 21-22

It is a common knowledge to all drilling personnel that the drilling rate (or the ROP) decreases when the mud weight increases. However, it may not always be clear that when the mud weight decreases, the drilling rate increases. The critical point for the drilling rate is about 3500 kPa (35 bar) overbalance. That is, after overbalance pressure decreases to values of less than 3500 kPa, the drilling rate continually increases. The drilling-rate increase continues into the underbalanced range until a point is reached where more *bit energy* (weight and speed) is needed, or where the bit begins to flounder. *Bit flounder* occurs when the cuttings are not cleaned out from under the bit fast enough and the bit redrills cuttings. The driller will normally see this as *bit balling.*

The chip hold down effect is one of the most important factors influencing on the ROP. In overbalanced condition the formation will be harder to drill, as the cutting will be held back in the formation because the pressure acting with a straight angle towards the cuttings. In underbalanced conditions the cuttings are being chipped away from the formation, while the drill bit barely touches the rock. In the same manner as the fluid from the production zone is directed from high pressure to lower pressure, the cuttings will due to the chip effect flow towards lower pressure conditions.

As the pressure conditions goes from being overbalanced to approach balanced conditions, the formation matrix are exposed for less stress. In underbalanced conditions, the matrix stresses are even lower. Thus there will be a formation which is much softer to drill, something that will grant a higher ROP. Underbalanced conditions will provide improved cleaning of the bit face, as the cuttings always will flow towards a lower pressure state, significantly less cuttings will be collected at the bit face. Along with better hole cleaning, this will provide a more effective drill bit, leading to a higher ROP.



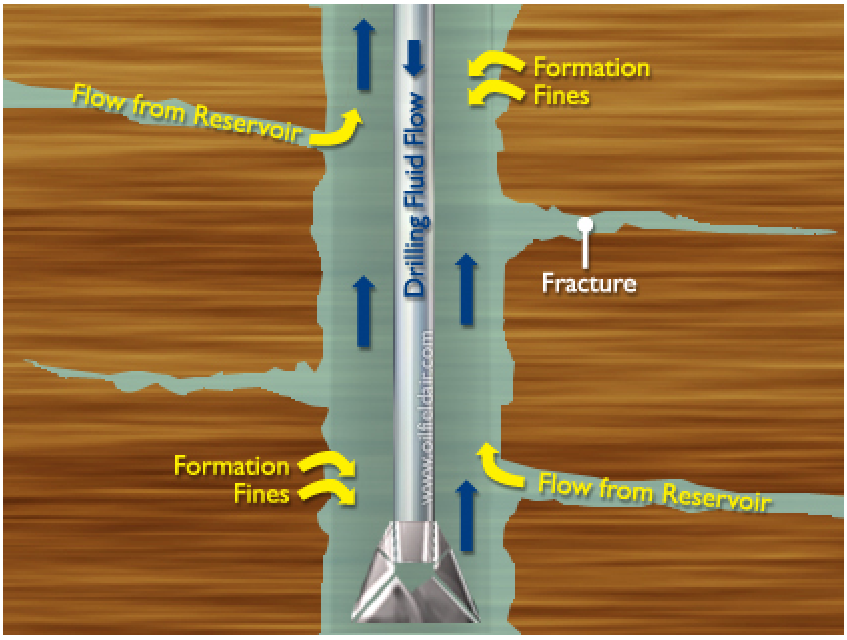
Figur 5: The skin effect; mud filtrate that damages the formation

Avoiding reservoir damage

Taken from [1], p. 22-23

In overbalanced drilling there is the appearance of the skin effect. The skin effect is due to the drilling fluid filtrate migrating into the pores of the reservoir zone and the mud cake which is situated at the wellbore. The filter cake plugs the formation and limits the amount of filtrate that enters the formation. When finally completing a well, most of the filter cake is removed, but there will always be some left, obstructing the flow from the reservoir into the wellbore. Thus, the reservoir will not be able to produce at its maximum, due to the damage caused by the filter cake, or the skin effect.

When drilling with underbalanced conditions the skin effect will be eliminated. In this case it is very important to maintain the underbalanced conditions throughout all the drilling and completion sequence to avoid reservoir damages. UBO thus allows the well to produce naturally without expensive enhancement techniques.



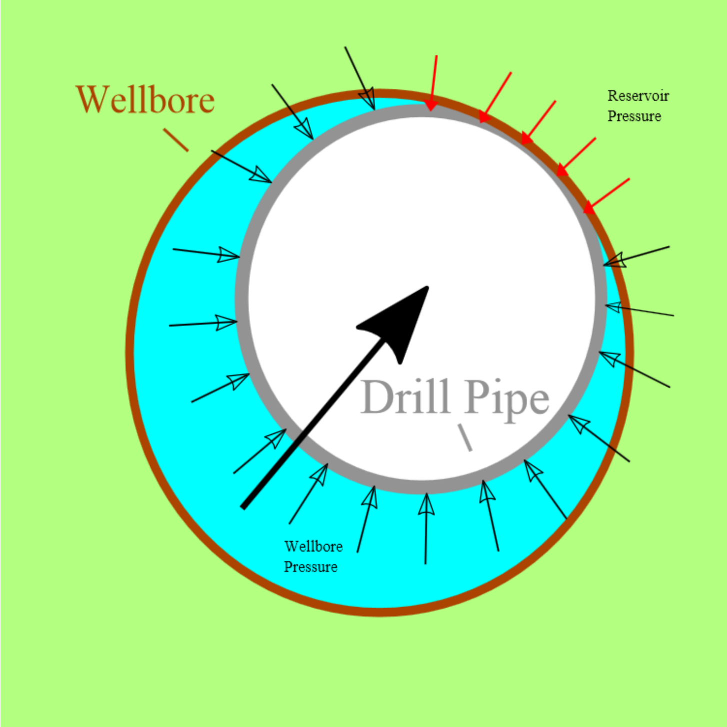
Figur 6: UBO, no skin effect, no filter cake, just natural flow while drilling

Avoiding differential sticking

Taken from [1], p. 23-24

Filter cake is formed on the wellbore, as filtrate leaks out of the drilling mud into the formation. This is a typical operational phenomenon for overbalanced drilling.

Filtrate is a clear or colored liquid with very little solid content. When the liquid filtrate leaves the drilling mud, the clay and barite solids in the mud that are left behind on the wellbore form a relatively impermeable cake, called filter cake. Filter cake continually build up and then is abraded by the rotation or sliding of drill pipe and the flow of drilling mud. Differential sticking occurs when the drill stem comes to rest against filter cake in an overbalanced hole. Low pressure on the reservoir side pulls, while high pressure on the wellbore side pushes the pipe to the side of the hole.



Figur 7: Differential sticking

In UBO, no filter cake builds up because flow is from a permeable zone to the wellbore, and not from the wellbore into the formation.

In horizontal drilling, the drill pipe rests on the bottom of the hole during a great deal of the operation. Because much of the horizontal portion of the well is usually in a permeable reservoir, some horizontal wells use UBO techniques to avoid differential sticking.

A literature review on underbalanced drilling and steady state model well planning

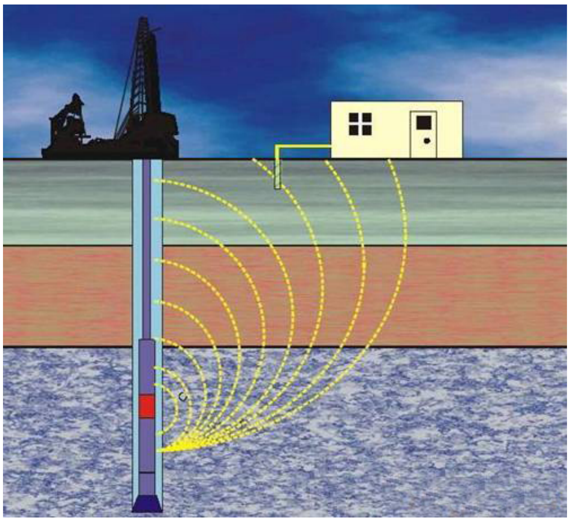
UBO: Pros and cons

Taken from [2]

In "SPE Drilling & Completion" (SPE 52889) the author gives praises and perils to the UBO concept. Some of the main concerns are these:

When artificially generated, an UBO condition is most often mechanically accomplished by injecting gas into the drillstring. In this process, the noncondensable gas reduces the density of the entire circulating fluid system in both the injection path (inside the drillstring) and in the returning fluid flowing back to surface in the annular space outside the string. Specialized surface equipment for pressurized flow, solids separation, cuttings sampling, and well control are required for this operation.

The MWD Problem. There is a drawback of this injection method. Conventional mud-pulsed logging techniques cannot be used while maintaining an underbalanced condition because of the presence of a compressible gas in the fluid system. Some MWD service companies say that their tools can tolerate up to 28 % of gas in the mud stream, but this is often somewhat vague assumptions. In addition, the underbalanced condition may be lost or compromised on a regular basis during connections of drillstring as the drilling operation proceeds. The use of alternate mechanical configurations, such as a parasite tubing string or concentric drillstring, eliminates this concern and facilitates more continuous underbalanced operation and conventional measurement while drilling (MWD) operations by injection of the noncondensable gas directly into the returning fluid stream at some intermediate location in the annular wellbore. Added cost and complexity are the downsides of these applications.



Figur 8: EMT signal sending

The solution for the MWD problem is the development of EMT (Electromagnetic Telemetry) tools, which directly transmit downhole information back to the surface while drilling, even in an underbalanced mode. Depth and temperature limitations and some formation restrictions on these tools still currently limit their applicability in deeper wells but it is expected that, as technology continues to advance in this area, deeper wells will be drilled with this technology. An increased use of coiled tubing drilling technology for UBO that utilizes an internal wireline for MWD purposes can also minimize problems associated with MWD operations during UBO.

The expense. UBO is usually more expensive than a conventional drilling program, particularly if drilling in a sour environment or in the presence of adverse operational or surface conditions (i.e.remote locations, offshore, etc.). Also, there is little advantage to drilling a well in an underbalanced mode if the well is not completed in an underbalanced fashion. This often results in additional costs for snubbing equipment required to strip the drillstring from the hole in an underbalanced flow condition. A portion of this expense may be offset by increased ROP conditions resulting in a reduction in drilling and rig time and if the well can be drilled in a truly underbalanced fashion, limited or no completion work will be required, reducing the cost of extensive and expensive completion and stimulation treatments which may often be required in severely damaged horizontal and vertical wells. Obviously, the major objective in implementing a UBO operation in most cases is to improve well productivity over a conventional overbalanced completion. Therefore, in a properly executed operation, it is expected that the potential downside of increased drilling costs will be more than offset by increased productivity of the well.

The safety concerns. The technology for drilling and completing wells in an underbalanced fashion continues to improve. Recent developments in surface control equipment, rotating blowout prevention equipment, and the increased usage of coiled tubing in UBO, has increased the reliability of many UBO operations. The fact that wells must be drilled and completed in a flowing mode, however, always adds safety and technical concerns in any drilling operation. The use of air, oxygen content-reduced air, or processed flue gas as the injected gas in a UBO operation, although effective at reducing the cost of the operation, can cause concerns with respect to flammability and corrosion problems. Considerable work has been done recently in high pressure testing to ascertain safe combustible limits of produced mixtures of natural gas, oil, and drilling mud with air, flue gas, and oxygen content-reduced air.

The failure to maintain a continuously underbalanced condition. A major factor in the disappointing results from many UBO operations conducted in the past is that the underbalanced condition is not maintained 100% of the time during drilling and completion operations. The major issue here is that there is no impetus for the formation of any type of classic sealing filter cake on the surface of the rock because the formation pressure is greater than the circulating fluid pressure in a truly underbalanced operation. Obviously, this is advantageous with respect to formation damage and differential sticking concerns that may be associated with the influx of potentially damaging filtrate or mud solids into the formation, but it also means that the protective ability and presence of this filter cake as a barrier to fluid and solids invasion is negated. If the formation is abruptly (or gradually) exposed to a condition of periodic pulses of overbalance pressure, very rapid and severe invasion of filtrate and associated solids may occur. This problem is often compounded by the fact that very thin, low viscosity, base fluid systems are usually used in most UBO operations to facilitate effective disengagement of the noncondensable gas from the fluid in the surface equipment for solids control purposes. In some UBO situations, the invasive damage is more significant than if a properly designed overbalanced system had been used in the first place because invasive depth and profile can often be minimized in many overbalanced systems with the proper mud and bridging agent design.

The multiphase flow simulator

Taken from [3]

An SPE paper with the following title "Practical Use of a Multiphase Flow Simulator for Underbalanced Drilling Applications Design" discusses the importance of multiphase flow modeling for UBO operations. UBO operations are synonymous with multiphase flow modeling. After that a candidate well has passed requisite reservoir valuation and screening criteria, the next step will be to establish the technical possibility of the technology and determine the key operational parameters including equipment specifications. Multiphase flow modeling is an essential part of this process and influences all aspects of design and construction of an UBO well.

Parameters that will satisfy an underbalanced condition will then be implemented into the multiphase flow model over a range of conditions, including sensitivities and safety factors, allows the sizing of the surface equipment (with maximum reservoir inflow considerations), and thus implicitly influences the design of the entire UBO operation.

The primary goal of this paper, according to the authors, is "to describe the strategy of effective multiphase modeling for UBO including modeling sequence and necessary sensitivities, to ultimately deliver a "robust" design."

The primary suspect of flow modeling flaws is the uncertainty of reservoir pressure.

The main goal of a multiphase flow model is to establish a pressure profile along the entire circulating path. This requires combination of conservation principles with closure relationships such as PVT relationships. In general, a complete UBO flow modeling software program should include, at a minimum, the following features:

- A flow regime model, which predicts the two-phase flow regime at any given location in the flow path,
- A liquid hold-up model, to calculate the liquid fraction at any given location, with taking into account for slip between phases,
- A model for predicting frictional pressure losses in a two-phase flow system. Although hydrostatic pressure loss may dominate the total pressure loss, frictional pressure drop is an important contribution to the bottom hole pressure, and must be accurately modeled, especially in friction-dominated systems,
- Thermal, PVT and property models to account for the effect of temperature and pressure on the phase behavior and transport properties of the fluids,
- A model to predict hole cleaning performance

This paper also discusses and recommends a modeling sequence which is divided into 6 steps.

1. Step number 1 is to model the annular pressures and velocities assuming no flow from the formation. This is done to determine what combination of liquid and gas gives the desired pressure drawdown across the reservoir section. Specifying the surface pressure and the liquid and gas that will be exiting the bit allows for the modeling of the annulus. The model calculates the resulting pressure profile in the annulus, and thus the bottomhole pressure.

2. When the flow simulations are developed, the next step would be to develop an underbalanced operating window. Only those combinations of liquid, gas and surface choke pressures that meet the pressure drawdown requirement, minimum liquid velocity requirement and are within motor limits, are acceptable. The “window” allows for a better understanding of the possibility of successful underbalanced drilling operations. For example if all the UBO constraints are satisfied (UBO pressure, hole cleaning, and motor performance) and the operating envelope is large, the preliminary engineering of the UBO potential appears promising.

3. The third step would be to model injection pressure and MWD operability. The model is utilized to calculate the injection pressures and the equivalent liquid volume through the motor. For these calculations, the volume of liquid and gas pumped into the standpipe is specified along with the bottomhole pressure.

   Once the combination of injection pressures and injection rates are known the gas volume fraction (GVF) profile in the drill pipe may be determined. This is usually determined at the minimum circulating bottom hole pressure conditions, and with the maximum gas rates within the operating envelope. This is of importance to determine the operability of the conventional MWD equipment. Mud pulse telemetry has proven unreliable for GVF between 8% and 28%, depending upon the pulsing method and manufacturer. The results of this modeling are used to determine downhole tool limitations (motor, MWD), mud pump requirements (liner sizing) and gas compression requirements.

4. What remains now is to take reservoir inflow into account. The modeling must be repeated considering reservoir inflow. It is a common trend, that a marginal attention is given to reservoir inflow in UBO multiphase flow simulation.
   In modeling inflow, it is useful to work with Production Engineers familiar with the subject reservoirs to establish the inflow performance relationship and PVT behavior models, both of which are critical in accurately establishing UBO operability under inflow conditions.
   It is also important to model various openhole depths, and to use the expected productivity with UBO, rather than conventionally realized productivity.
   Depending on the productivity of the reservoir, it is suggested that as much as 200 meter increments be modeled with the productivity expected during UBO. The primary purpose of this approach is to ensure that the well does not become overbalanced at the bit due to high productivity along the remaining lateral.

5. Next step is to model the controllability of the well. Questions to answer here is like "how controllable is the bottomhole pressure by varying the choke pressure?"
   The results of this modeling sequence may be used to develop BHCP versus flowing wellhead (or choke) pressure curves. The maximum allowable wellhead pressure, at maximum UBO PI, is determined by the choke pressure applied before the well becomes overbalanced. It is important that this analysis is conducted for both circulating (injecting fluid) and for production (if the well is capable of natural flow).

6. The final step in this modeling sequences to summarize the parameters of the UBO equipment including safety factors. The process of flow analysis described above continues iteratively until the applicable operating parameter range is identified, together with injection and well controllability boundaries. With all of these model results (including adequate sensitivities) it now becomes possible to confidently specify equipment requirements.

The aim of this paper is to present a strategy for UBO flow modeling. This is of much importance, because a multiphase flow model is critical to the design and construction of an UBO well. The author also discusses that a similar multiphase modeling approach to that presented should also be applied to the hydraulic design and equipment sizing and specification prior to an underbalanced drilling operation. The strategy, including modeling sequence and necessary sensitivities, should deliver a “robust” design while optimizing equipment requirements.

The steady state flow model

In this section of the thesis, a steady state flow model which was given as an assignment in the course MPE190 Well Intervention, autumn 2007 at UiS, will be presented. The model has been modified to satisfy certain conditions. The model has been extended for a more realistic UBO situation.

The extensions were:

- The wellpath was extended from a vertical well to include a deviated and horizontal section.
- The flowareas has been made to look a bit more realistic. Thus the well has a cased section and a open hole. The drillstring has drillcollars and BHA attached to it. All of these additions give a more realistic flowpath for the circulating fluid inside the annulus.
- More realistic density models have been used. The geothermal gradient has been used to provide a temperature profile in the well such that temperature could be taken into account in the density calculations.

Description of the existing Steady State Model

Main Program

The source code for the flow model is written in Matlab programming language. The source code can be found in the appendix.

The source code consists of 6 files:

This Matlab file is, as it already states in the filename, the main file in the code. The whole simulation is run by this code.

The variable input in this code is:

- welldepthMD
- welldepthTVD
- tc; the bottomhole temperature in Celsius degrees.
- nobox; number of segments in the well.
- nopoints; number of nodes in the well, which is nobox+1, because of the first segment in the well, which is at the bottom of the well.
- boxlength; welldepthMD/nobox, this gives the length of the segments.
- liquidrate; injectionrate of the water.
- gasrate; injectionrate of the nitrogen gas.

All the calculations are performed using SI units, that is, Pascal for pressure and m^3/s for rates.

The nopoints is an index array keeping track of the end point of the boxes. The inputs, liquidrate and gasrate, will be sent into the function itsolver which again calls upon the function wellpressure down the code, where it will finally return the BHP.

The numerical solver

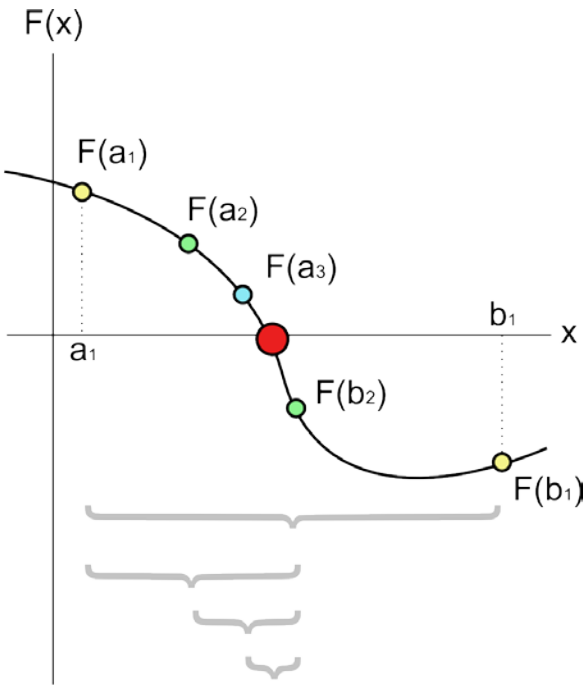
Taken from [4]

This is the core of the program, where the correct value for the BHP is found by use of the "method of halving the interval".

In mathematics, this is a root-finding method which repeatedly bisects (halves) an interval and then selects a subinterval in which a root must lie for further processing. It is a very simple and robust method, but it is also relatively slow. Because of this, it is often used to obtain a rough approximation to a solution which is then used as a starting point for more rapidly converging methods. But in our model, speed is not of importance, because the iteration is not that complicated as opposed to other simulator algorithms.

The method is applicable when we wish to solve the equation f(x) = 0 for the real variable x, where f is a continuous function defined on an interval [a, b] and f(a) and f(b) have opposite signs. In this case, (a) and (b) are said to bracket a root since, by the intermediate value theorem, the f must have at least one root in the interval (a, b).

At each step the method divides the interval in two by computing the midpoint c = (a+b) / 2 of the interval and the value of the function f(c) at that point. Unless c is itself a root (which is very unlikely, but possible) there are now two possibilities: either f(a) and f(c) have opposite signs and bracket a root, or f(c) and f(b) have opposite signs and bracket a root. The method selects the subinterval that is a bracket as a new interval to be used in the next step. In this way the interval that contains a zero of f is reduced in width by 50% at each step. The process is continued until the interval is sufficiently small.



Figur 9: The method of halving the interval

Explicitly, if f(a) and f(c) are opposite signs, then the method sets c as the new value for b, and if f(b) and f(c) are opposite signs then the method sets c as the new a. (If f(c)=0 then c may be taken as the solution and the process stops.) In both cases, the new f(a) and f(b) have opposite signs, so the method is applicable to this smaller interval.

In this program, we have said that if $\mathbf{f(x)<f_{tot}}$ we are satisfied. Since we use Pascal, we can set the tolerance value $\mathbf{f_{tot} = 1000}$ **Pa**, which will give a good answer.

The pressure guessed for the BHP, x_{guess}, is set to the hydrostatic pressure of liquid in the well. This is not correct thou, since we have gas and friction effects in addition. But it is a good starting point for the iteration.

The implementation of numerical solver

The numerical solver in this simulator solves the function **wellpressure(p_{bot}) = 0**. This corresponds to the general **f(x) = 0**, in mathematics.

In reality, it will be impossible to find the **wellpressure(p_{bot}) = 0**. That is why we approximate, and set a tolerance value, **f_{tot} = 1000 Pa**, which is indeed a very small value when dealing with pressures.

Then there are two variables which must be set, and that is;

- xguess
- xint

xguess = 1000 x 9.81 x welldepthTVD (which was already set in main.m)
xint = 400000000 Pa (400 bars)

Hereby, the search interval is set. xguess is the pressure which is guessed for the bottomhole. As it's seen from its formula, the hydrostatic pressure of liquid in the well is given as the bottomhole pressure. This is just a guessed value, but a good and dynamic way to start the whole process of iteration. All it remains for the iteration is actually to "find" out how much the friction effects are to be accounted for in addition to the hydrostatic bottomhole pressure.
xint is just another set value, which in this simulator has been set to 400 bars.

Further down in the code, the variables of **x1** and **x2** are calculated.

Where;

- x1 = xguess-xint/2.0
- x2 = xguess-xint/2.0

Then there are the functions of **f(x1)** and **f(x2)**, which will give a call to the function f = wellpressure, something that will be explained later in this chapter.

f(x1) and **f(x2)**, is called as follows;

- f1 = wellpressure(x1,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc)
- f2 = wellpressure(x2,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc)

As its seen from the programming statements above, to be able to find f1(f(x1)) and f2(f(x2)), there is need for 7 inputs, or "arguments" in the function **wellpressure**.
6 of these arguments have already been given, and are coming from **main.m**. The only arguments originating from **itsolver.m** are **x1** and **x2**.

Before starting the whole iteration process, or the loop, there has to be a check on whether **f1xf2>0**. If this was the case, there would be no iteration. As already given, the numerical solver assumes a positive **f1** and a negative **f2**, where as to find **f(x) = 0**, or in our case **f(x) = 1000** Pa.

When finally entering the loop, a variable **x3** is calculated, which is the bisection of **x1** and **x2**.

Then there is a variable **noit**, which basically counts the number of iterations done, giving us a final total number in the end of the simulation.

Here is the code for the iterator. We can say, that this is the core of the whole simulator.

```
if (f1*f2)>=0
     error = 1;
     pbot = 0;
 else
 % start iterating, we are now on the track.
     x3 = (x1+x2)/2.0;
     f3 = wellpressure(x3,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);
     while (f3>ftol | f3 < -ftol)
        noit = noit +1 ;

        if (f3*f1) < 0
           x2 = x3;
        else
           x1 = x3;
        end

        x3 = (x1+x2)/2.0;
        f3 = wellpressure(x3,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);
        f1 = wellpressure(x1,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);

     end
     error = 0;
     pbot = x3
     noit
 end
```

Pressure calculation along the wellpath function

This is the most comprehensive code in the simulator. Most of the code is about defining variables, such as wellpath, inclination, TVD, flowarea, fluid and gas viscosity, gas slippage parameters, liquid and gas mass rates. And then there is the creation of matrices for each variable. The cells in the matrix represent the well segments. Which is the variable defined in the **main.m -> nopoints**. And then, fill these empty matrices with real values through iteration.

Before going further with this code, we assume that the outlet pressure of the well is 1 Bar. This is a physical boundary condition that we have to ensure that the model reaches. Since this is a UBO, there is a choke present, and this choke will define the outlet pressure. The choke pressure in the code can be set to desired value, but it has to be done in **wellpressure.m**.

When iterating in **wellpressure.m** and filling the well segments with values, we always start with the last segment in the well, which is the deepest. By guessing the BHP, with the variable **pbotguess**. When the BHP is guessed, all that remains is to find out the pressure and flowrate values for all of the segments above. The top segment, which is at the outlet, will have a surface flowrate and a surface outlet pressure. This outlet pressure, which is a calculated pressure, should equal the physical outlet pressure condition. Henceforth, the definition for the wellpressure(pbot) is:

- wellpressure(pbot) = pcalcsurface – prealsurface

When the correct bottomhole pressure is found, this function will be zero.

The choke pressure in the code can be varied to study the effect it has on the flow conditions.

Extensions at the Steady state model

In this code there are some sets of information that are needed to make the simulation more realistic in nature. Parameters like geothermal gradient, well geometry and flowareas have been added.

Inclusion of the temperature profile.

A simple temperature gradient has been implemented in the code. That is, for every 1 km TVD, the temperature has to increase 25 degrees Celcius. A surface temperature of 15 degrees Celcius is assumed.

This approach for the geothermal gradient is a very vague one. This has many reasons. To name a few:

- There is constant production, so the fluid flowing in the annulus is not reaching the same temperature as the geothermal gradient.
- The major part of the borehole is cased. So there is no direct contact with the fluid flowing and the formation with the calculated geothermal gradient.

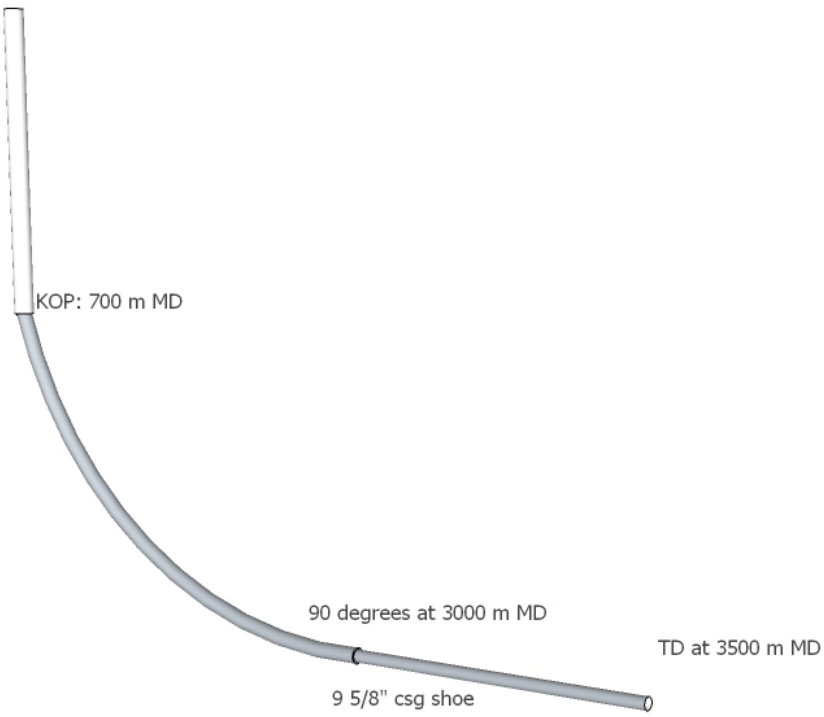
The calculation of gas density (in the code rogas.m) and liquid density (in the code roliq.m) takes this geothermal gradient into account. The temperature values in these formulas are converted into Kelvin degrees before being taken into calculation.

Inclusion of the area changes

This particular simulation case has 4 flow areas.

- At the bottom of the well, from MD 3500 m – 3200 m, we have inner diameter of the annulus, which is the OD of the BHA, and we have the outer diameter of the annulus, which is the 8 ½" open borehole.
- From MD 3200 m – 3000 m, we have inner diameter of the annulus, which is the OD of the drill collars, and the outer diameter of the annulus, which is the 8 ½" open borehole.
- From MD 3000 m – 2500 m, we have inner diameter of the annulus, which is the OD of the drill pipes, and the outer diameter of the annulus, which is the 8 ½" open borehole.
- From MD 2500 m – 0 m, we have the inner diameter of the annulus, which is the OD of the drill collars, and the outer diameter of the annulus, which is the 9 $^{5/8}$" casing, closed hole.

The well geometry has to be entered into the source code manually in the routine **wellpressure.m**. Wherein, before the extension of the code, all that mattered was the total depth of the well. There was no geometry changes to account for, just a true vertical scenario.



Figur 10: The well geometry

The implementation of inclination

The inclination values for the well had to be entered manually. The deviated part of the well was defined by increasing each box (which is to be 100 meter segments of the box) in the section by 9 degrees from MD 700 m on until reaching MD 1200 m, where the well path was kept at a constant 45 degrees until reaching MD 2600 m, where the inclination increments continued with 9 degrees per box (per 100 meter segment), until reaching 90 degrees at MD 3000 m.

This set up was the easiest path to follow. Because the well geometry is dependent on many different factors. The author's main concern was to have a well geometry which was consistent with the discretization of the well. Having the total length of the well to be 3500 m MD, and the segment lengths at 100 meters per segment, would give 35 boxes. From the top of the well till segment number 7 (700 m MD), from segment number 8 till 12 there is the inclination for the deviated section, then from segment 13 till segment 25 there is the deviated section, then from segment 26 till segment 30 there is another inclination section, from there on is the horizontal section until segment number 35.

The inclination sections are set as shown:

```
inc(6) = cos(81*(pi/180));
inc(7) = cos(72*(pi/180));
inc(8) = cos(63*(pi/180));
inc(9) = cos(54*(pi/180));

inc(24) = cos(36*(pi/180));
inc(25) = cos(27*(pi/180));
inc(26) = cos(18*(pi/180));
inc(27) = cos(9*(pi/180));
```

And the vertical, deviated and horizontal sections:

```
if i>=1 & i<6
  inc(i) = cos(90*(pi/180));
  elseif i>=10 & i <24
      inc(i) = cos(45*(pi/180));
  elseif i>27 & i<36
      inc(i) = 1;
  end
```

The definition of variables and matrices

Further down in **wellpressure.m** there is the definition of viscosity for gas and liquid. Then there is the gas slippage parameters, which are given ideally, set to **k = 1.2** and **s = 0.55**. This gas slippage, express that gas moves faster upward than liquid (e.g. a gas bubble rising in stagnant water)

Then there is the whole concept of steady state mass flow, that is, that is, production rates and etc. are constant at surface. This is the main assumption for all the calculations done for the whole well. This is a very simple concept; mass going into a fictive box, is the same mass coming out of the box.

Then there is the section in the code where we define the variable matrices needed. Several matrices are made. These are just 36 x 1 (number of points (nodes) x 1 column).

Here is a list over the matrices, each row of the different matrices to be filled in the "for loops" coming after;

```
% vl - liquid vel, vg -gas velocity,
% vgs,vls are superficial velocities.
% eg-el - phase volume frac gas and gas
% p - pressure., rhol liquid density, rhog gas density
% fricgrad - friction gradient
% hydgrad - hydrostatic gradient
% temp - temperature
% inc - inclination
% TVD - True Vertical Depth

vl = zeros(nopoints,1);
vg = zeros(nopoints,1);
vls = zeros(nopoints,1);
vgs = zeros(nopoints,1);
eg = zeros(nopoints,1);
el = zeros(nopoints,1);
p  = zeros(nopoints,1);
fricgrad = zeros(nopoints-1,1);
hydgrad = zeros(nopoints-1,1);
temp = zeros(nopoints,1);
inc = zeros(nopoints-1,1);
TVD = zeros(nopoints-1,1);
```

Discretization

The next step in wellpressure.m is to discretize the matrices with values.

By using the inlet values for each segment, the pressures from hydrostatic and friction contributions are calculated. Then, after the calculation of hydrostatic and friction gradients, pressures for each segment is calculated. By subtracting from the initial BHP the hydrostatic gradient and friction gradient from the segment before.

When discretizing the superficial velocities of liquid and gas, the boundaries must be taken into account. That is, the flow area boundaries. For this the following procedure has been used:

```
%here we have the flowarea boundaries. The first shift is from
%flowareaBHA -> flowareaDC, which happens at vls(3)

if i>2 & i<4
vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDC/flowareaBHA;
vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDC/flowareaBHA;

%here we have the flowarea boundary flowareaDC -> flowareaDPoh, which
%happens at vls(5)

elseif i>4 & i<6
vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDPoh/flowareaDC;
vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDPoh/flowareaDC;

%here we have the flowarea boundary flowareaDPoh -> flowareaDPch, which
%happens at vls(10)

elseif i>9 & i<11
vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDPch/flowareaDPoh;
vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDPch/flowareaDPoh;

else
 vls(i+1)=vls(i)*roliq(p(i),tk)/roliq(p(i+1),tk);
 vgs(i+1)=vgs(i)*rogas(p(i),tk)/rogas(p(i+1),tk);
end
```

Then these superficial velocities are used to calculate the velocities of liquid and gas in the flow. No specific flow regimes have been implemented to this work. Thus we have constant gas slippage parameters. These were given earlier.

Then there is the last part of the loop, where we have **sumfric** and **sumhyd**. These are just the sums of the hydrostatic and friction contributions to the BHP.

The friction gradient

This code calculates the friction gradient in the well. This is a mixture model which is based on a SPE paper by Antonio C.V.M. Lage [5].

After the mixture values are defined, the mixture Reynolds number is calculated, which is used to distinguish laminar from turbulent flow. Pending on whether we have laminar or turbulent flow, different expressions for the friction factor are given. For Reynold numbers in the transitional zone, interpolation is used. This friction factor is then used in the expression for the friction pressure loss gradient.

Then there is the calculation of the friction factor itself. The Reynolds number is checked for boundary conditions, to find if the flow is turbulent or laminar. If not interpolation is implemented.

The last step is to find the friction loss gradient.

```
 rhol = roliq(pressure,t);
 rhog = rogas(pressure,t);
 romix = rhol*el+rhog*eg;
 viscmix = viscl*el+viscg*eg;
 vmix = vg*eg+vl*el;

 % Calculation of the mixture Reynolds number

 re = romix*vmix*(do-di)/viscmix;

 % Calculation of the friction factor. For re > 3000, the flow is
turbulent.
 % For re < 2000, the flow is laminar. Interpolate in between.

 if (re >= 3000)
  fricfactor = 0.052*re^(-0.19);
 elseif ( (re<3000) & (re > 2000))
  f1 = 24/re;
  f2 = 0.052*re^(-0.19);
  xint = (re-2000)/1000.0;
  fricfactor = (1.0-xint)*f1+xint*f2;
 else
  fricfactor = 24/re;
 end


 % calculate friction loss gradient (Pa/m)

  friclossgrad = 2*fricfactor*romix*vmix*abs(vmix)/(do-di);
```

Simulations

In this section of the thesis some simulations will be presented for the given well case scenario.

These simulations are to illustrate how tweaking certain variables may affect the BHP, and whether the given "tweak" is feasible for the underbalanced drilling operation.

The simulations were done in 3 separate conditions.

1. No reservoir inflow and a constant choke pressure at 3 Bars.
2. Reservoir inflow and a constant choke pressure at 3 Bars.
3. Reservoir inflow and a variable choke pressure.

Each of these simulations were done in different liquid injection rates. These were 1000 l/min, 1500 l/min and 2000 l/min.

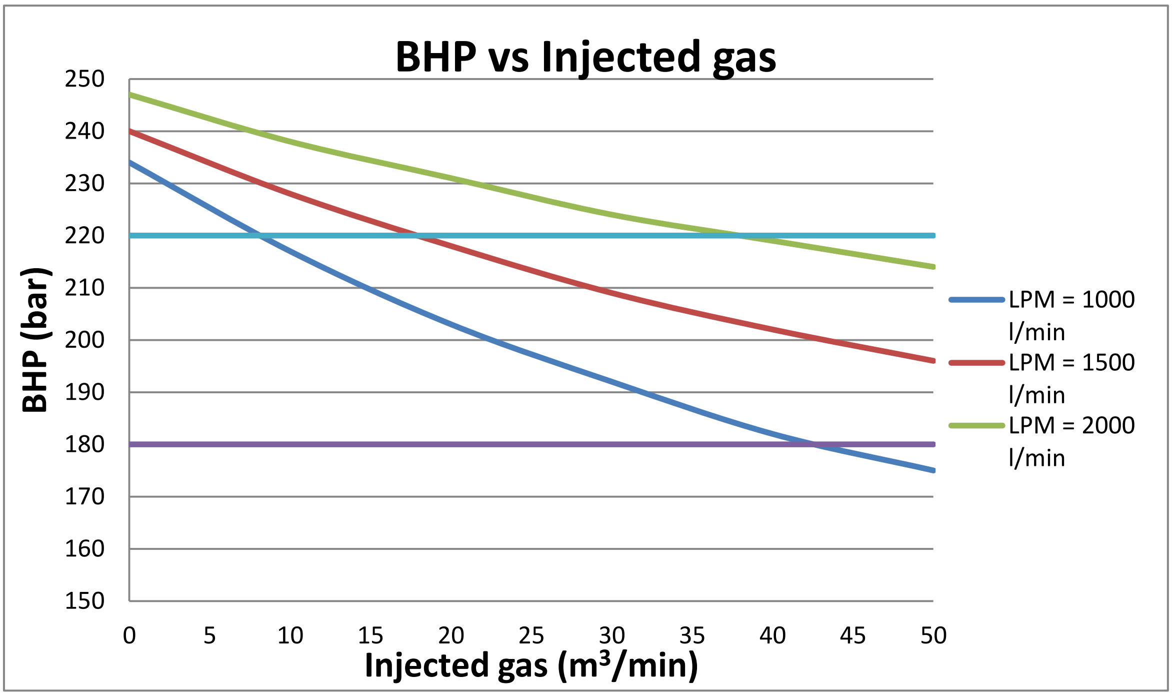
The production index was set to 0.6 m^3/day. And it is produced gas.

The variable choke pressures were 3 Bars, 6 Bars and 9 Bars.

The injected liquid is water with a density of 1000 kg/m^3. The drilling fluid is also water.

The injected gas is nitrogen with a density of 1 kg/m^3.

Case 1: Variable LPM, Constant choke, No reservoir inflow



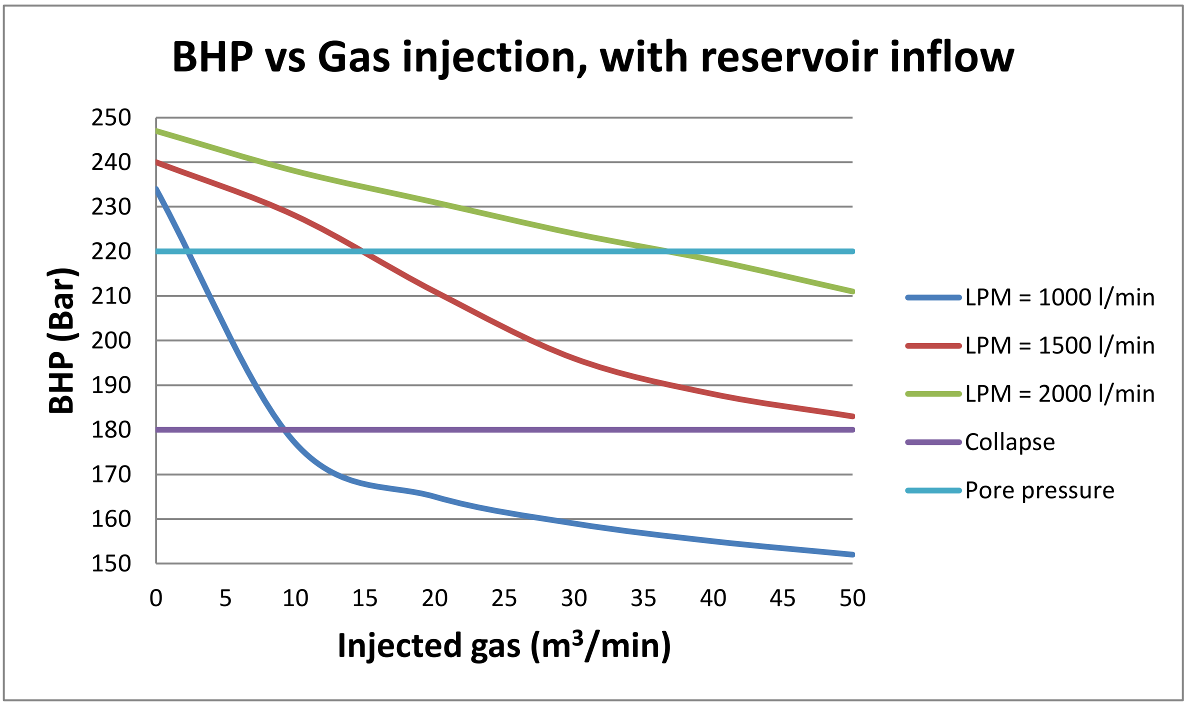
Figur 11: Variable LPM, Constant choke, No reservoir inflow.

	Injected gas rate (m^3/min)					
LPM (l/min)	**0**	**10**	**20**	**30**	**40**	**50**
1000	234	217	203	192	182	175
1500	240	228	218	209	202	196
2000	247	238	231	224	219	214

Tabell 1: Variable LPM, Constant choke, No reservoir inflow.

This graph is showing the effect the injection gas has on the BHP on different injection rates. The overall trend seen is a curvature, which is trending towards a horizontal line. The explanation for this is the friction dominion, which increases with the injection rate of gas.

Case 2: Variable LPM, Constant choke, with Reservoir inflow



Figur 12: Variable LPM, Constant choke, with Reservoir inflow.

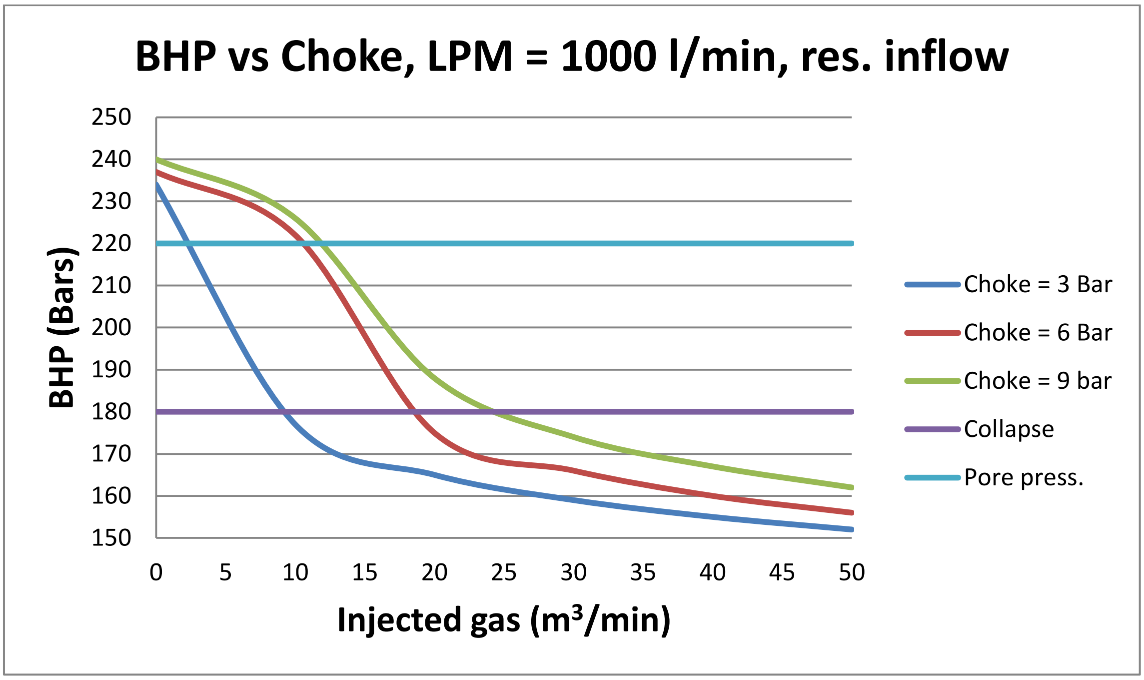
	Injected gas rate (m^3/min)					
Rate (l/min)	**0**	**10**	**20**	**30**	**40**	**50**
1000	234	177	165	159	155	152
1500	240	228	211	196	188	183
2000	247	238	231	224	218	211

Tabell 2: Variable LPM, Constant choke, with Reservoir inflow.

The sudden drop of the trend when the injection rate of water is low (LPM = 1000 l/min), is due to the production inflow. The production of gas downhole, has an effect of decreasing the BHP, especially when the injection rate of water is low. When the injection rate of water is low, there is less friction. So the system is not friction dominated, as it is the case when LPM is increased.

Again it is seen, that when injection rate of gas is increased, the trend is approaching steady state, and this is due to the friction dominion.

Case 3: Variable LPM, Variable choke, with Reservoir inflow

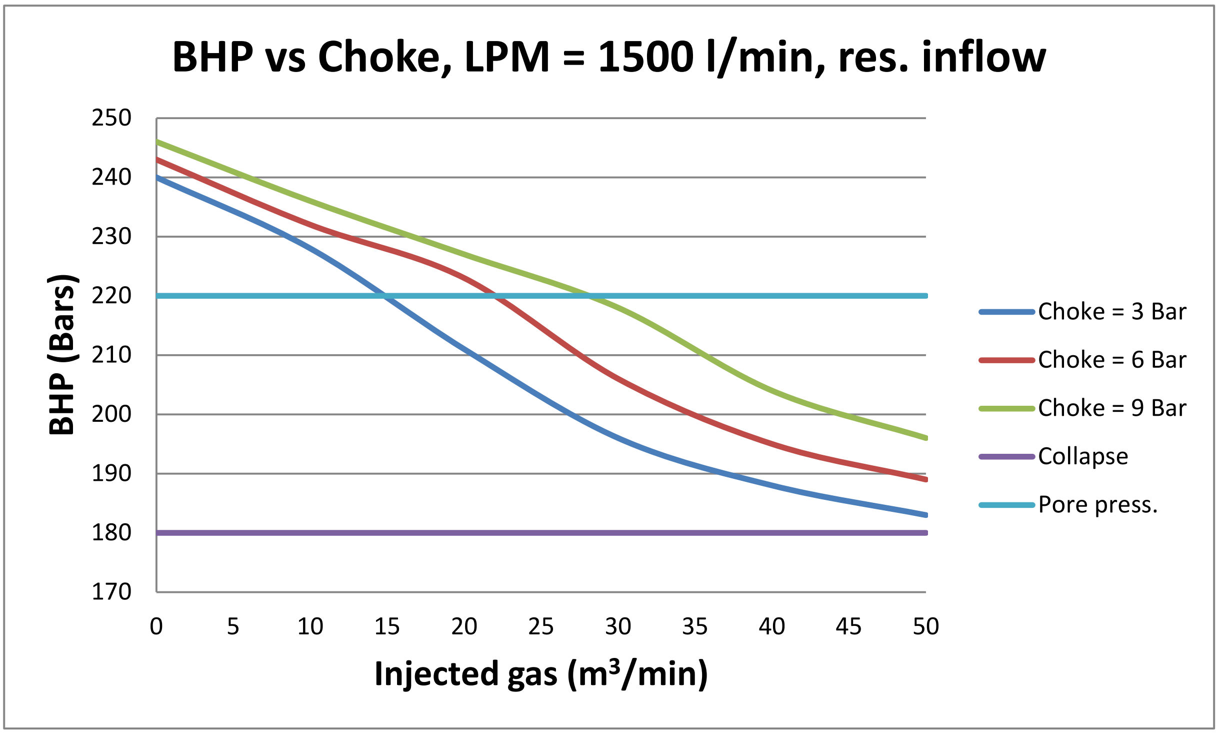


Figur 13: Variable LPM, Variable choke, with Reservoir inflow, LPM = 1000 l/min.

LPM = 1000 l/min				
		Choke pressure (Bar)		
Injected gas (m^3/min)		**3**	**6**	**9**
0		234	237	240
10		177	222	226
20		165	175	188
30		159	166	174
40		155	160	167
50		152	156	162

Tabell 3: Variable LPM, Variable choke, with Reservoir inflow, LPM = 1000 l/min.

The drop in pressure due to the reservoir inflow (production index) is visible on each choke pressure, but the drop is delayed when the choke pressure is increased. This is due to the effect choke pressure has on the BHP.

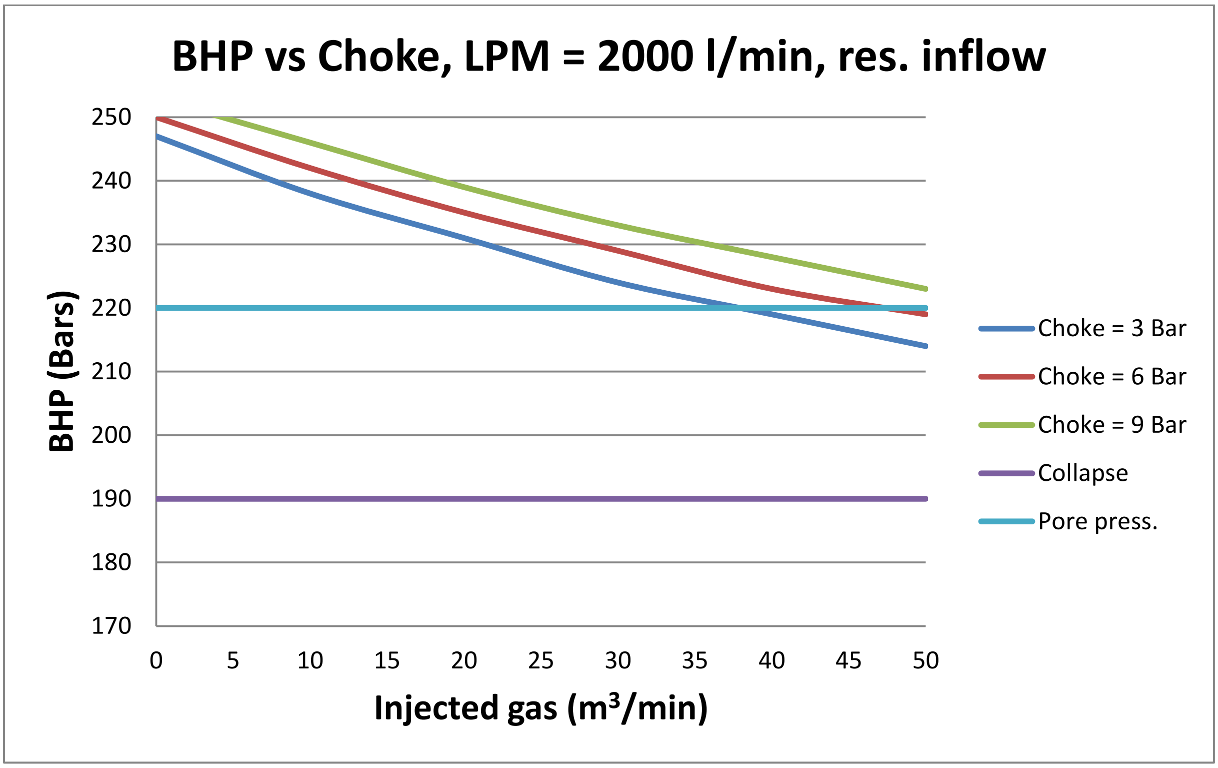


Figur 14: Variable LPM, Variable choke, with Reservoir inflow, LPM = 1500 l/min.

LPM = 1500 l/min				
		Choke pressure (Bar)		
Injected gas (m^3/min)		**3**	**6**	**9**
0		240	243	246
10		228	232	236
20		211	223	227
30		196	206	218
40		188	195	204
50		183	189	196

Tabell 4: Variable LPM, Variable choke, with Reservoir inflow, LPM = 1500 l/min.

The drop in pressure due to the reservoir inflow is less visible here than it was the case for LPM = 1000 l/min. This is due to the increased water injection rate, which increases the friction dominion in the well.



Figur 15: Variable LPM, Variable choke, with Reservoir inflow, LPM = 2000 l/min.

LPM = 2000 l/min				
Gas Influx (m^3/min)		Choke pressure (Bar)		
		3	6	9
0		247	250	253
10		238	242	246
20		231	235	239
30		224	229	233
40		219	223	228
50		214	218	223

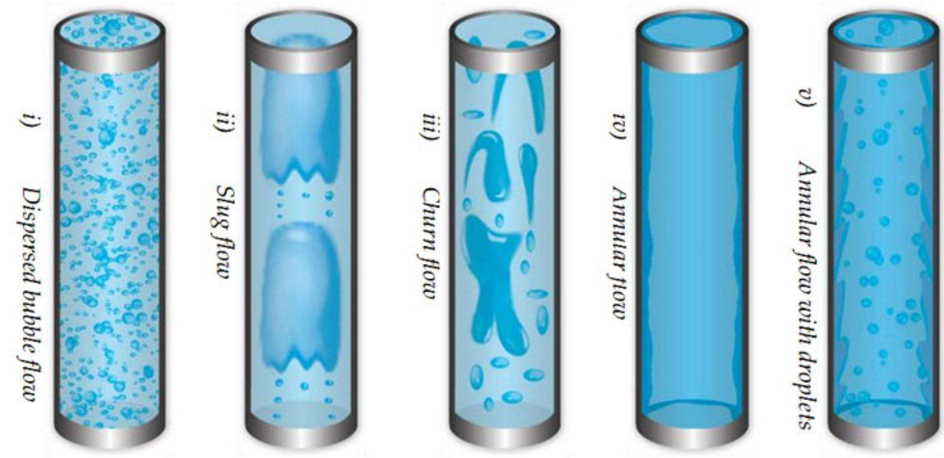
Tabell 5: Variable LPM, Variable choke, with Reservoir inflow, LPM = 2000 l/min.

The water injection rate is so high that the well is friction dominated all the way. This means that the effect reservoir inflow has on the BHP is negligible.

Discussion and conclusion

This model need improvements, that is evident. The main challenge in this thesis has been to go from the simple vertical well design to the more complex, and non-intuitive, horizontal well design. Why is the horizontal well non-intuitive, because it requires real well geometry data, with measured depth, inclination and azimuth. So, when in a planning phase of a underbalanced drilling operation, and a simulation wants to be done, all the data inputs like measured depth, inclination and azimuth has to be given. This can be challenging on the drilling engineers, because a planned well will not always follow the plan – the well geometry. But this is something the UBO simulation programs' dynamicity takes care of – the on-the-fly simulation. Where, by changing some variables, like the measured depth, inclination and azimuth, one is able to get new outputs.

The gas slippage parameters in this thesis has been constants. This is obviously not correct in a UDO, where there'll be different flow regimes in the well. In the science of multiphase flow, there are multiple flow regimes like; bubble flow, slug flow, churn flow and annular flow.



Figur 16: Gas-liquid flow regimes

One of the objectives of this thesis has been the attempt to implement the bubble flow model. But it was then realized, on a later stage, that the implementation of this model was too complicated to be worked on, with regards to the limited time there was.

But the approach for the implementation of the bubble flow model was somewhat clear. And that was to follow the example of the **itsolver.m**, where the numerical method of the "Bisection method" or the "Method of halving the interval" was used to find the correct BHP by looping. The variable that would be of importance in this respect would be the **S** in the formula for the gas velocity $\mathbf{V_g = KV_{mix} + S}$. This **S** would then be a function $\mathbf{f(\alpha_g)}$ of the gas fraction. The "Bisection method" would be used to find when $\mathbf{f(\alpha_g) = 0}$, just as it was used to find when $\mathbf{f(p_{bot}) = 0}$.

Another improvement that has to be done is the adjustment of the temperature gradient according to the flows interaction with the formation and also with the casing. Thou we have a constant flow in the well, the surrounding formation temperature will not have sufficient time to affect the flow with its temperature. So the flow will only get a fraction of the heat transfer from the formation temperature. So this heat transfer should be calculated according to the time of exposure that the fluid in the well has with the surrounding formations' temperature. Another case is when the annulus' outer diameter becomes the casing wall. Then a whole another approach has to be counted for. The heat transfer will not have the same effect as it had when the annulus outer diameter was the borehole, or the formation wall.

All of these variables have to be accounted for, so that an overall heat transfer gradient can be calculated.

In this thesis, as it has been given, the temperature gradient is very superficial. It is not something that would be accounted for in any UDO scenario, because it doesn't take into account for the above mentioned phenomenon.

The calculation of the true vertical depth for the deviated sections of the well has been done using simple trigonometry. This is also not a valid approach in this case, where the most suitable calculation method would be that of "The minimum curvature method". This method was studied, but when the realization of the need for real well geometry inputs like inclination, measured depth and azimuth was seen, the method was disregarded. Another important reason was the complications with regards to the discretization of the well into segments. The segments had to match the well inclinations, so that each segment would have its own measured depth and constant inclination. A variable inclination in one segment would create complexities surpassing the capacity of this thesis.

References

[1] Bill Rehm. *Practical Underbalanced Drilling and Workover.* Petroleum Extension Service: The University of Texas at Austin, 2002. ISBN 0886981980

[2] D.B. Bennion, SPE, F.B. Thomas, SPE, R.F. Bietz,* SPE, and D.W. Bennion, SPE, Hycal Energy Research Laboratories Ltd. *Underbalanced Drilling: Praises and Perils.* SPE 52889. December, 1998.

[3] C. Mykytiw, Blade Energy Partners, P. V. Suryanarayana, Blade Energy Partners, P.R. Brand, Blade Energy Partners, *Practical Use of a Multiphase Flow Simulator for Underbalanced Drilling Applications Design*, SPE 91598, October 2004.

[4] http://en.wikipedia.org/wiki/Bisection_method

[5] Antonio C.V.M. Lage, Petrobras; Rune W. Time, Stavanger University College, RF-Rogaland Research, *Mechanistic Model for Upward Two-Phase Flow in Annuli,* ISBN 978-1-55563-910-5.

[6] ”Applied Numerical Analysis”, Curtis F. Gerald, Patrick.O.Wheatley, Fourth Edition. Addison-Wesley Publishing Company. ISBN 0-201-52825-8, 1989.

[7] http://drbratland.com/PipeFlow2/chapter1.html

[8] Lecture notes from UBO course, Kjell Kåre Fjelde, University of Stavanger, 2007.

[9] Lecture notes from multiphase flow course (MPE 700), Rune W. Time, 2011.

[10] Lecture notes from petroleum production course (MPE 700), Aly Hamouda, 2011.

Figures

Figur 1: http://petroleumsupport.com/wp-content/uploads/2011/06/conventional-drilling.jpg

Figur 2: taken from IADC/SPE 122281

Figur 3: Bill Rehm (2002). Practical Underbalanced Drilling and Workover. Petroleum Extension Service: The University of Texas at Austin, page 12

Figur 4: Bill Rehm (2002). Practical Underbalanced Drilling and Workover. Petroleum Extension Service: The University of Texas at Austin, page 13

Figur 5: Bill Rehm (2002). Practical Underbalanced Drilling and Workover. Petroleum Extension Service: The University of Texas at Austin, page 23

Figur 6: http://petroleumsupport.com/wp-content/uploads/2011/06/underbalanced-drilling.jpg

Figur 7: http://upload.wikimedia.org/wikipedia/commons/d/d2/Differential_sticking.svg

Figur 8: http://bjbaonton.com/uploadfile/image/main/007.jpg

Figur 9: http://upload.wikimedia.org/wikipedia/commons/c/c2/Bisection_method.png

Figur 10: Authors work, using "Google Sketchup"

Figur 11: Authors work, using "Microsoft Excel 2010"

Figur 12: Authors work, using "Microsoft Excel 2010"

Figur 13: Authors work, using "Microsoft Excel 2010"

Figur 14: Authors work, using "Microsoft Excel 2010"

Figur 15: Authors work, using "Microsoft Excel 2010"

Figur 16: http://drbratland.com/PipeFlow2/images/Figure1_2_3.jpg

Appendix A: The matlab source code

Main.m

```
% A program developed for calculating  well pressures in a
% well where we have both liquid and gas flow. The model assumes that we
% have steady state conditions (constant flowrates at surface) and no time
% variations. The model is based on calculating the correct bottomhole
% pressure for certain gas and liquid flow rates and takes into account
% both the hydrostatic pressure and frictional pressures.

% All calculations are done using SI units (Pa for pressure),m3/s for
% rates.

clear;

 % Here we specify the measured and the vertical depth of the well.
 welldepthMD = 3500;
 welldepthTVD = 2204.5;
 % Then we specify the surface temperature in celsius degrees.
 tc = 15;
 % And then the number of boxes (segments) we want in our calulations.
 nobox = 35;
 % And from the number of boxes we get the number of points (nodes) which
 % will always be nobox+1, because of the known (guessed)
 % bottomhole pressure point (node).
 % nopoints is an index array keeping track of the end point of the boxes.
 nopoints = nobox+1;
 % And at last, the boxlength (segment length) is found and used in the
calculations.
 boxlength = welldepthMD/nobox;

  % Other initialisations like fluid properties and viscosties etc are done
  % deeper down in the code structure. Please note that you have the change
  % values there if you want to do changes in these routines. This is also
  % true for the inner/outer diameter of the annulus.

  % Now we will call a function that calculates the pressure along the well
  % for a given liquid flowrate and a gas rate. We call this function
  % solver because it is the zero point solver (e.g. regula falsi that
  % iterates until it finds the correct pressure. This solver routine again
  % calls upon a function "f(Pbottom)" called wellpressure.  The rotine
  % solver actually finds the correct bottomhole pressure that makes the
  % function wellpressure become zero "f(Pbottom) = 0". Then we have found
the correct
  % pressure profile.

  % INPUT variables
  % Rates are given in m3/s. We assume only liquid flow first.
  % Liquid rate is 1500 l/min. Convert to m3/s
  % Gas rate is in m3/min. Convert to m3/s

  liquidrate = 2000/1000/60;
  gasrate = 50/60;


  [pbot,error] =
itsolver(nopoints,boxlength,welldepthTVD,gasrate,liquidrate,tc);
```

Wellpressure.m

```
function f =
wellpressure(pbotguess,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,t
c)

% NB, At first stage we assume that our outlet pressure is 1 Bar (atm
% pressure). This is the physical boundary condtion that we have to ensure
% that out model reaches. If a choke is present. The surface pressure will
% be different. It measns that if the choke pressure is 300 000 Pa then the
variable below should be
% set to this. You change it her:

 prealsurface  = 900000;

%We now start by the deepest box with the pressure we assume: pbotguess and
% for each box, we calculate the pressure and flowrates. In the end, we end
up with some surface
% rates and a surface outlet pressure. The calculated outlet surface
% pressure should equal the physical outlet condition (now 100 000 Pa). We
% can therefore define our wellpressure(pbot)=pcalcsurface-prealsurface.
% The function will be zero if the correct bottomhole pressure is found.



%The surface temperature in kelvin is
  tk(1) = 273.5 + tc;

%This well has 3 flowareas

 diBHA = 0.15; %the OD of the BHA assembly
 doBHA = 0.2125; %the 8 1/2" borehole
 diDC  = 0.125; %the OD of the Drill Collars
 doDC  = 0.2125; %the 8 1/2" borehole
 diDP  = 0.0875; %the OD of the Drill Pipe
 doDPoh = 0.2125; %the 8 1/2" borehole
 doDPch = 0.1825; %the 9 5/8" casing ID, which is 8.9"

  flowareaBHA = pi()/4*(doBHA*doBHA-diBHA*diBHA);
  flowareaDC = pi()/4*(doDC*doDC-diDC*diDC);
  flowareaDPoh = pi()/4*(doDPoh*doDPoh-diDP*diDP);
  flowareaDPch = pi()/4*(doDPch*doDPch-diDP*diDP);

% Specify viscosities [Pa s]. In real life they depend on pressure and temp

  viscl = 0.001;
  viscg = 0.00001;


% Define gas slippage parameters.
  k = 1.2;
  s = 0.55;


% gas gravity

   g = 9.81;
```

```
 % The mass rate is the same at surface/atmosphere and at bottomhole since
we have steady state. This is later
 % used to find the rates at downhole conditions.

  liqmassratesurf = liquidrate*roliq(100000.0,tk);
  liqmassratebhp  = liqmassratesurf;

  gasmassratesurfinj = gasrate*rogas(100000.0,tk);
  gasmassratebhpinj  = gasmassratesurfinj;

   prodinx = 0.6/3600/1000;


   gasmassratesurfres = prodinx*(22000000-pbotguess);

 if (gasmassratesurfres <0)
     gasmassratesurfres = 0;
 end


   %gasmassratebhpinj  = gasmassratesurfinj*rogas(100000.0,tk);
   gasmassratebhpres  = gasmassratesurfres;

   gasmassratebhp = gasmassratebhpinj+gasmassratebhpres;


  % Now we loop from the bottom to surface and calculate accross all the
  % segments until we reach the outlet.

  % Define the variables needed. Initialise them first for comp efficiency.
  % vl - liquid vel, vg -gas velocity,
  % vgs,vls are superficial velocities.
  % eg-el - phase volume frac gas and gas
  % p - pressure., rhol liquid density, rhog gas density
  % fricgrad - friction gradient
  % hydgrad - hydrostatic gradient
  % temp - temperature
  % inc - inclination
  % TVD - True Vertical Depth

  vl = zeros(nopoints,1);
  vg = zeros(nopoints,1);
  vls = zeros(nopoints,1);
  vgs = zeros(nopoints,1);
  eg = zeros(nopoints,1);
  el = zeros(nopoints,1);
  p  = zeros(nopoints,1);
  fricgrad = zeros(nopoints-1,1);
  hydgrad = zeros(nopoints-1,1);
  temp = zeros(nopoints,1);
  inc = zeros(nopoints-1,1);
  TVD = zeros(nopoints-1,1);

  % Before we loop, we define  all variables at the inlet of the first
  % segment(at bottom). As starting point we use the fact that we know the
mass
  % rate of the different phases (same as on top of the well)
```

```
% First find the rates in m3/s (downhole)
liquidratebhp = liqmassratebhp /roliq(pbotguess,tk);
gasratebhp = gasmassratebhp/rogas(pbotguess,tk);

% Find the superficial velocities
vls(1) = liquidratebhp/flowareaBHA;
vgs(1) = gasratebhp/flowareaBHA;

% Find Phase velocities

vg(1) = k*(vls(1)+vgs(1))+s;
eg(1) = vgs(1)/vg(1);
el(1) = 1-eg(1);
vl(1) = vls(1)/el(1);

% Set pressure equal to guessed pressure
p(1) = pbotguess;
TVD(1) = welldepthTVD;
% Set BHP temperature in kelvin
%tk(1) = 333;

%Here we define the inclination values of the deviating parts of the well
inc(6) = cos(81*(pi/180));
inc(7) = cos(72*(pi/180));
inc(8) = cos(63*(pi/180));
inc(9) = cos(54*(pi/180));

inc(24) = cos(36*(pi/180));
inc(25) = cos(27*(pi/180));
inc(26) = cos(18*(pi/180));
inc(27) = cos(9*(pi/180));

% Now we loop across the segments.

sumfric = 0;
sumhyd = 0;

for i =1:nopoints-1
    %Here we define the inclination values for the vertical, deviated and
    %horizontal sections of our borehole

if i>=1 & i<6
inc(i) = cos(90*(pi/180));
elseif i>=10 & i <24
    inc(i) = cos(45*(pi/180));
elseif i>27 & i<36
    inc(i) = 1;
end

TVD(i+1)=TVD(i)-inc(i)*boxlength;
TVD(nopoints)=0;

tk = 273.5+(25/1000*TVD(i))+ tc;
temp(i) = tk;
temp(nopoints) = 273.5 + tc;
%The outer diameter and inner diameter in the flowarea is set, although
%we already have defined the flowareas above, we need to "repeat" this
%section for the fricgrad(i) method.
   if i>=1 & i<4
```

```
        do=doBHA;
        di=diBHA;
    elseif i>=4 & i<6
        di=diDC;
        do=doDC;
    elseif i>=6 & i<11
        di=diDP;
        do=doDPoh;
    else
        di=diDP;
        do=doDPch;
    end
 % use the inlet values for each seg. to calculate hydrostatic
 % and friction pressure across each segment.

  hydgrad(i) = (roliq(p(i),tk)*el(i)+rogas(p(i),tk)*eg(i))*g;
  fricgrad(i) = dpfric(vl(i),vg(i),el(i),eg(i),p(i),do,di,viscl,viscg);

  %the pressures here are calculated according to the TVD, thus we have
  %  boxlength*inc.
  p(i+1)=p(i)-hydgrad(i)*boxlength*inc(i)-fricgrad(i)*boxlength;

  %here we have the flowarea boundaries. The first shift is from
  %flowareaBHA -> flowareaDC, which happens at vls(3)

  if i>2 & i<4

vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDC/flowareaBHA;

vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDC/flowareaBHA;

  %here we have the flowarea boundary flowareaDC -> flowareaDPoh, which
  %happens at vls(5)
  elseif i>4 & i<6

vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDPoh/flowareaDC;

vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDPoh/flowareaDC;

  %here we have the flowarea boundary flowareaDPoh -> flowareaDPch, which
  %happens at vls(10)
  elseif i>9 & i<11

vls(i+1)=vls(i)*(roliq(p(i),tk)/roliq(p(i+1),tk))*flowareaDPch/flowareaDPoh
;

vgs(i+1)=vgs(i)*(rogas(p(i),tk)/rogas(p(i+1),tk))*flowareaDPch/flowareaDPoh
;

  else
   vls(i+1)=vls(i)*roliq(p(i),tk)/roliq(p(i+1),tk);
   vgs(i+1)=vgs(i)*rogas(p(i),tk)/rogas(p(i+1),tk);
  end


 vg(i+1) = k*(vls(i+1)+vgs(i+1))+s;
 eg(i+1) = vgs(i+1)/vg(i+1);
 el(i+1) = 1-eg(i+1);
 vl(i+1) = vls(i+1)/el(i+1);
```

```
 sumfric = sumfric+fricgrad(i)*boxlength;
 sumhyd = sumhyd+hydgrad(i)*boxlength*inc(i);
 end

 pout = p(nopoints);
 f = pout-prealsurface;

sumfric
sumhyd
pout
p
```

Itsolver.m

```
function [pbot,error] =
itsolver(nopoints,boxlength,welldepthTVD,gasrate,liquidrate,tc)

% The numerical solver implementeted here for solving the equation f(x)= 0
% "wellpressure(pbot)= 0" is called the
% Method of Halving the Interval (Bisection Method)

% You will not find exact match for f(x)= 0. Maybe f(x) = 0.0001. By using
% ftol we say that if f(x)<ftol, we are satisfied. Since our function
% gives results in Pascal, we say that ftol = 1000 Pa gives us a quite good
% answer.

 ftol = 1000;

 % Specify the search interval". xguess is the pressure you guess for the
 % bottomhole. We here use hydrostic pressure of liquid in the well as our
 % initial guess. This is of course not nes. correct since we have gas and
 % friction effects in addtion. But it might be a good starting point for
 % the iteration. (Remember x is in Pa). 1 Bar = 100 000 Pa.

 % Set number of iterations to zero

  noit = 0;


 xguess = 1000*9.81*welldepthTVD;
 xint = 400000000;
 x1 = xguess-xint/2.0;
 x2 = xguess+xint/2.0;

 f1 =
wellpressure(x1,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);
 f2 =
wellpressure(x2,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);

 % First include a check on whether f1xf2<0. If not you must adjust your
 % initial search intervall. If error is 1 and zero pbot, then you must
 % adjust the intervall here.


 if (f1*f2)>=0
     error = 1;
     pbot = 0;
 else
 % start iterating, we are now on the track.
     x3 = (x1+x2)/2.0;
     f3 =
wellpressure(x3,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);
     while (f3>ftol | f3 < -ftol)
        noit = noit +1 ;

        if (f3*f1) < 0
           x2 = x3;
        else
           x1 = x3;
        end
```

```
        x3 = (x1+x2)/2.0;
        f3 =
wellpressure(x3,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);
        f1 =
wellpressure(x1,gasrate,liquidrate,nopoints,boxlength,welldepthTVD,tc);

     end
     error = 0;
     pbot = x3
     noit
 end
```

Dpfric.m

```
function friclossgrad = dpfric(vl,vg,el,eg,pressure,do,di,viscl,viscg)

 % Works for two phase flow. The one phase flow model is used but mixture
 % values are introduced.

 t = 333;

 rhol = roliq(pressure,t);
 rhog = rogas(pressure,t);
 romix = rhol*el+rhog*eg;
 viscmix = viscl*el+viscg*eg;
 vmix = vg*eg+vl*el;

 % Calculate mix reynolds number

 re = romix*vmix*(do-di)/viscmix;

 % Calculate friction factor. For re > 3000, the flow is turbulent.
 % For re < 2000, the flow is laminar. Interpolate in between.

 if (re >= 3000)
  fricfactor = 0.052*re^(-0.19);
 elseif ( (re<3000) & (re > 2000))
  f1 = 24/re;
  f2 = 0.052*re^(-0.19);
  xint = (re-2000)/1000.0;
  fricfactor = (1.0-xint)*f1+xint*f2;
 else
  fricfactor = 24/re;
 end

 % fricfactor

 % calculate friction loss gradient (Pa/m)

  friclossgrad = 2*fricfactor*romix*vmix*abs(vmix)/(do-di);
```

Roliq.m

```
  function rhol = roliq(p,t)

% Model for water density
% Reference: http://www.engineeringtoolbox.com/fluid-density-temperature-
pressure-d_309.html
% p is in Pascal
% tk is in Kelvin, t in Celsius
% vanntetthet i kg/m3
% water is heaviest at 4 C, 1000 kg/m3. use this and 1 bar as refernce
density

tk = t-273.5;
% E -bulk modulus fluid elasticity (N/m2)
E = 2.15*10^9;
% B Volumetric temperature expansion coefficient. (1/V)
B = 0.0002;

%rhol =rot*factor;

rhol = 1000/(1-(p-101325)/E)/(1+(tk-4)*B);
```

Rogas.m

```
function rhog = rogas(p,t)

% p is given i pascal
% tk is given i Kelvin

  rhog = 0.003368*p/t;
```