Universitetet i Stavanger

Faculty of Science and Technology

MASTER'S THESIS

Study program/specialization: **Petroleum Technology / Drilling & Well Technology**	Fall semester, 2013 **Open**
Writer: **Mohamed Agharbi**	……
Faculty Supervisor: **Kjell K. Fjelde**	
Title of thesis: **MPD and use of the AUSMV model to simulate and analyze propagation of pressure pulses.**	
Credits (ECTS): **30**	
Key words: **MPD Tools** **Rotating Control Device** **Choke Manifold** **Drift Flux Model** **AUSMV Scheme** **Pressure Pulse Propagation**	Pages: **93** + enclosure: **26** Stavanger, **02.01.2014** Date/year

DEDICATION

This work is dedicated to my parents, my sisters and my extended family to which I owe much for my achievements. This work is also dedicated to my nephews and my friends, in particular Rashid Elmi R.I.P.

ACKNOWLEDGEMENTS

I would like to thank and show my sincere gratitude to my supervisor Kjell K. Fjelde for his continuous help and support. I am thankful for his encouragement, patience and guidance throughout the process in writing this thesis. Without him this work would have been much more difficult to complete.

I would like to thank Professor Mesfin Belayneh for his additional support and help even though he didn't have to.

I would like to take this opportunity to again thank my family and friends for their prayers and support throughout the years.

Abstract

Offshore drilling is one of our era's most challenging operations in the petroleum industry. Conventional drilling methods face difficulty in certain prospects, such as drilling in deep water reservoirs and depleted offshore reservoirs. This is due to the narrow drilling window present in these types of reservoirs. There is a small margin between the pore pressure gradient and the fracture gradient which causes difficulty to drill with conventional methods. Managed pressure drilling (MPD) has emerged as a solution to many of the conventional drilling problems like kick scenarios, lost circulation, stuck pipe problems and drilling in depleted reservoirs. MPD provides us with precise control in these narrow margin wells by allowing drilling close to the pore-pressure gradient. The conventional drilling problems are either reduced or eliminated completely by using MPD technology.

Most of MPD is executed by drilling in a closed well loop utilizing a Rotating Control Device (RCD) with a minimum of one drill string Non-Return Valve, and a Drilling Choke Manifold. The choke can be controlled manually or automatically. In today's world the most common type is the automatic choke regulation. It requires PID controllers as a regulation technique, this is based on the difference between measured pressure vs required pressure. In addition to that, the automation process is supported by hydraulic simulator models. Wellbore control is very precise in MPD, assuming that the wellbore is sealed and able to contain the pressure. If this is the case, it is then possible to monitor the pressure throughout the wellbore in real time at the surface. Pressure changes are seen immediately in a closed system. MPD offers more precise control of the annular wellbore pressure profiles; hence influxes and losses are detected immediately. Safety of personnel and equipment during drilling is improved. Drilling economics is improved in MPD due to the reduction of drilling mud costs and reduction in non-productive time.

The Drift Flux model is used in the petroleum industry among other things to evaluate transient flow responses of drilling operations. It has its roots in the laws of conservation for two phase flow, and its goal is to describe the characteristics of flow in pipes or wells. The AUSMV (Advection Upstream Splitting Method) scheme is a hybrid flux-vector splitting scheme. The AUSMV scheme is used in this thesis to simulate different scenarios concerning propagation of pressure pulses in a well.

In the simulations we studied the pressure pulse propagation caused by pump start up and choke valve adjustments. It was demonstrated what effect friction has on the pulses and if the differences between having a one-phase and two-phase flow system. In the latter case, we also show how the gas volume content in the well affects the propagation velocities.

Pressure pulse propagation caused by pump start up and choke valve adjustments is studied in the simulations with the help of the AUSMV scheme. Studying the propagation of pressure pulses generated by adjusting the choke, is of importance in MPD because situations arise

where we have to increase the choke pressure to avoid kick. In a MPD system, presence of gas might easily occur since the system is designed for taking small gas kicks while drilling. Hence, in a long extended reach well, this is maybe an effect that one has to consider when working with an automated choke regulation system. After a given choke adjustment, one must give the well time to respond before an additional choke adjustment is introduced.

The results of the simulations show that the sonic velocity depends both on gas fraction and pressure. If we operate with gas in a well, typically we find that the sonic velocity is reduced most at the top of the well. As the pressure increases with well depth, the sonic wave propagation velocity increases. Therefore, if we adjust the choke by making fast updates based on frequent measurements in long wells, there is a possibility that this “time lag” is a factor which must be taken into account. This will particularly apply to underbalanced drilling systems where we know there will be significant volumes of gas in the well.

Table of Contents

List of Figures:

Nomenclature:

MPD: Managed Pressure Drilling

UBD: Underbalanced Drilling

BHP: Bottom Hole Pressure

NPT: Non-Productive Time

BOP: Blow Out Preventer

RCD: Rotating Control Device

ROP: Rate of Penetration

ECD: Equivalent Circulating Density

CCS: Continuous Circulation System

NRV: Non-Return Valve

AUSMV: Advection Upstream Splitting Method

DG: Dual Gradient

PMCD: Pressurized Mud Cap Drilling

CBHP: Constant Bottom Hole Pressure

PLC: Programmable Logic Controller

1. Introduction

In offshore drilling we often deal with narrow drilling windows between the pore-pressure and the fracture-pressure gradients. These scenarios typically occur in deep water drilling, where most of the overburden pressure is seawater. These narrow pressure operating windows also occur in fields which used to have more generous drilling windows, but with time, because of production and depletion the pressure gradients change. MPD provides us with precise control in these narrow margin wells by allowing drilling close to the pore-pressure gradient [12]. Managed pressure drilling (MPD) has emerged as a solution to many of the conventional drilling problems like kick scenarios, lost circulation, stuck pipe problems and drilling in depleted reservoirs.

In this work, the focus will firstly be on explaining the commonalities and differences between conventional drilling, UBD and MPD. The focus will then shift to MPD technology and MPD tools. In most MPD systems is the pressure controlled with a choke manifold and a backpressure system. In today's world the most common choke type is the automatic choke regulation. It requires PID controllers as a regulation technique, this is based on the difference between measured pressure vs required pressure. In addition to that, the automation process is supported by hydraulic simulator models.

The Transient flow models are explained in this work. They can be used for describing fluid flow in wells. The main aim of this project is to take into use a previously developed code for two phase flow and apply it for simulating some transient flow scenarios. Special focus will be on examples from underbalanced drilling and managed pressure drilling. Of special interest is to study the effect of sonic wave propagation. When the outlet choke is adjusted, how long time will it take before the adjustment is felt at the bottom and what effect will friction and one-phase/two phase flow conditions have on this. For extended reach wells the pressure pulses will use long time before reaching the pressure sensors at the bottom, will that have any implications for the control engineering system often used in controlling pressure in MPD systems ??

1.1 Conventional drilling

In conventional drilling, the circulation flow path starts with drilling fluid (mud) that is pumped downhole through the drillstring and drillbit. The mud then flows up the annulus and through a bell nipple at the top of the wellbore which lead the mud to a mud-gas separator and solid control equipment and finally back to the mud pit. Conventional wells are drilled in a state of overbalance. Overbalance is when exerted wellbore pressure is greater than the pore pressure throughout the exposed formation. Annular pressure in these cases is controlled by mud density and mud pump flowrates [7].

In static conditions, (Fig 1) we have that the bottomhole pressure (P_{BH}) is a function of hydrostatic pressure (P_{Hyd}) [7]:

$$P_{BH} = P_{Hyd}$$

In dynamic conditions, when the hole is being circulated, P_{BH} is equal to P_{Hyd} plus annular friction pressure P_{AF} (Fig 1):

$$P_{BH} = P_{Hyd} + P_{AF}$$

Problems associated with conventional drilling operations, are problems like lost circulation, kick and stuckpipe scenarios. These problems increase Non-Productive Time (NPT) significantly, which leads to economic losses and other hazards to personnel and environment. Well control monitoring in an open vessel is often problematic. Higher flow rates from the wellbore is used as an indicator to spot a bigger problem. Inner bushings are often pulled to check for flow. In a short time period a small influx can lead to a large kick. To monitor the pressures reasonably the well is shut-in and behaves like a closed vessel [7].

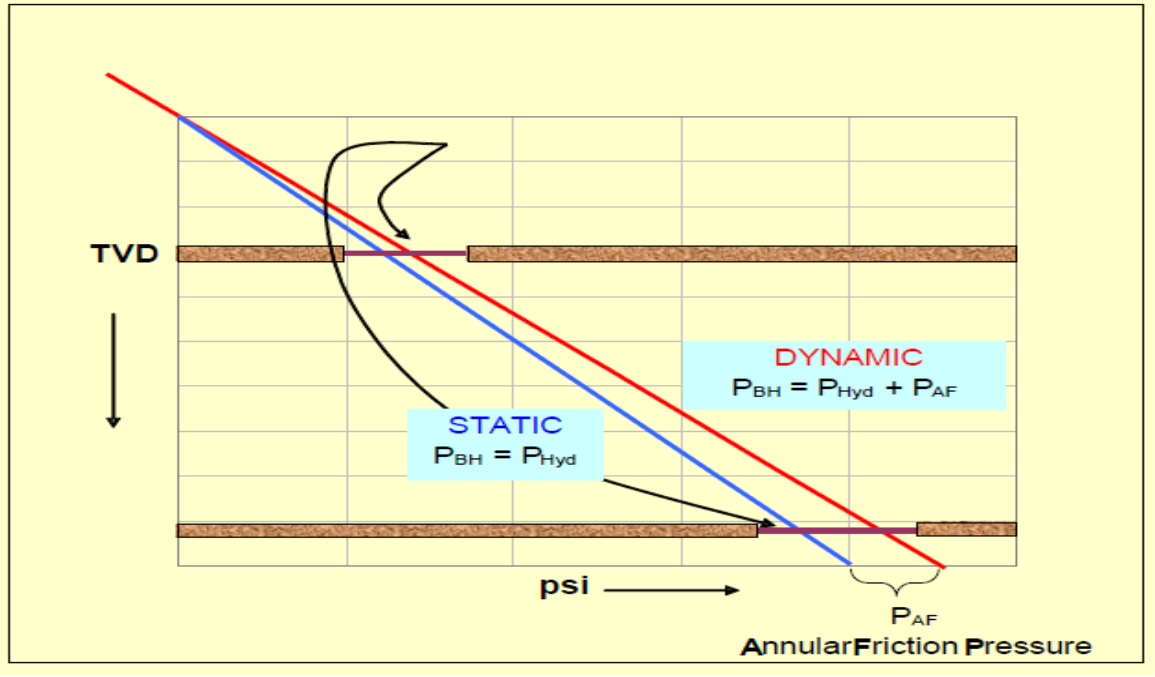


Figure 1: Static and dynamic pressure [7]

1.2 Underbalanced Drilling

Underbalanced drilling operations and techniques are used primarily to enhance reservoir productivity. Underbalance is the condition where the exerted pressure inside the wellbore is intentionally lower than the pore pressure throughout the exposed formation. The intention is to bring formation fluid to the surface; we have that [7]:

$P_{BH} < P_{Pore}$

Underbalanced drilling (UBD) improves rate of penetration (ROP). The main objectives though, are to protect, characterize and preserve the reservoir while drilling. Thus the well potential is not compromised. To achieve this objective, the well is allowed to produce and the well is controlled by three surface containment devices [7].

- Rotating Control Device (RCD)
- Drilling Choke Manifold
- Multiple Phase Separator

If the well is produced while drilling, then the gas is flared, recirculated or sold. Oil is typically stored in stock tanks in land- based drilling wells.

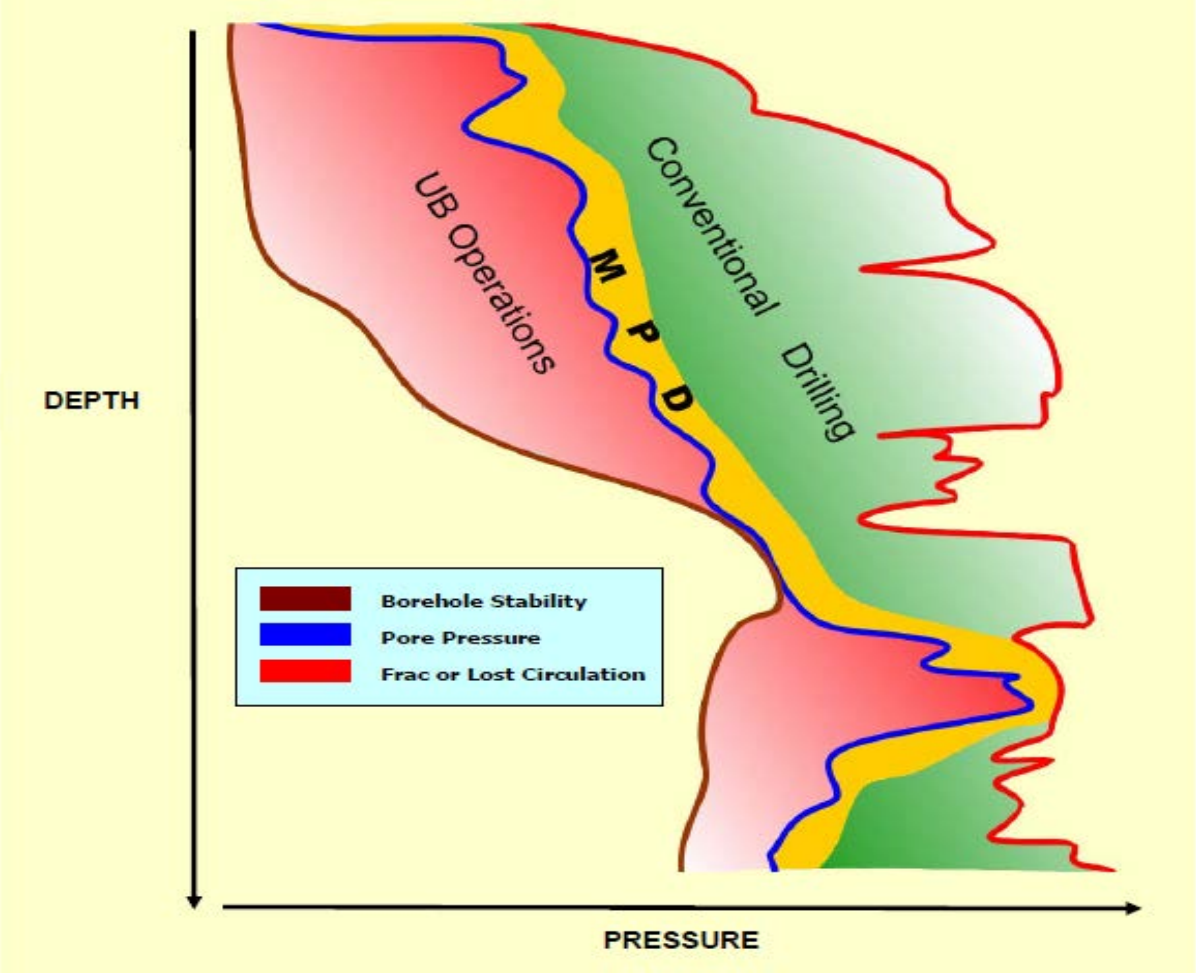


Figure 2: Drilling Windows for Conventional, MPD and Underbalanced Drilling Operations. [7]

UBD is a technique where the hydrostatic pressure in the circulating downhole fluid system is kept below the pore pressure while drilling the well. This allows hydrocarbon production during drilling. This condition can occur naturally if the pressure in the formation is high, and low density fluids are used. This technique is called flow drilling. In many cases the underbalanced condition must be achieved artificially. This is done by injecting a non-condensable gas to reduce effective hydrostatic density. Typically, the gas used is nitrogen, because it`s easily accessible and easy to transport. UBD techniques are often applied for horizontal wells due to formation damage concerns. In horizontal wells there are longer fluid contact time and openhole completions relative to vertical wells. UBD is used because even small invasive damage to an openhole horizontal well can reduce the productivity of the well. A cased and perforated vertical well doesn't have the same need for UBD as a horizontal well. [8]

Artificially generated underbalanced condition is often accomplished through drillstring injection. This is when the nitrogen gas is injected directly through the drillstring, and it reduces the density of the whole circulating fluid system, inside the drillstring and in the return flow outside the drillstring in the annulus. Special surface equipment for pressurized flow, well control, solids separation and cuttings sampling are required Figure 3. [8]

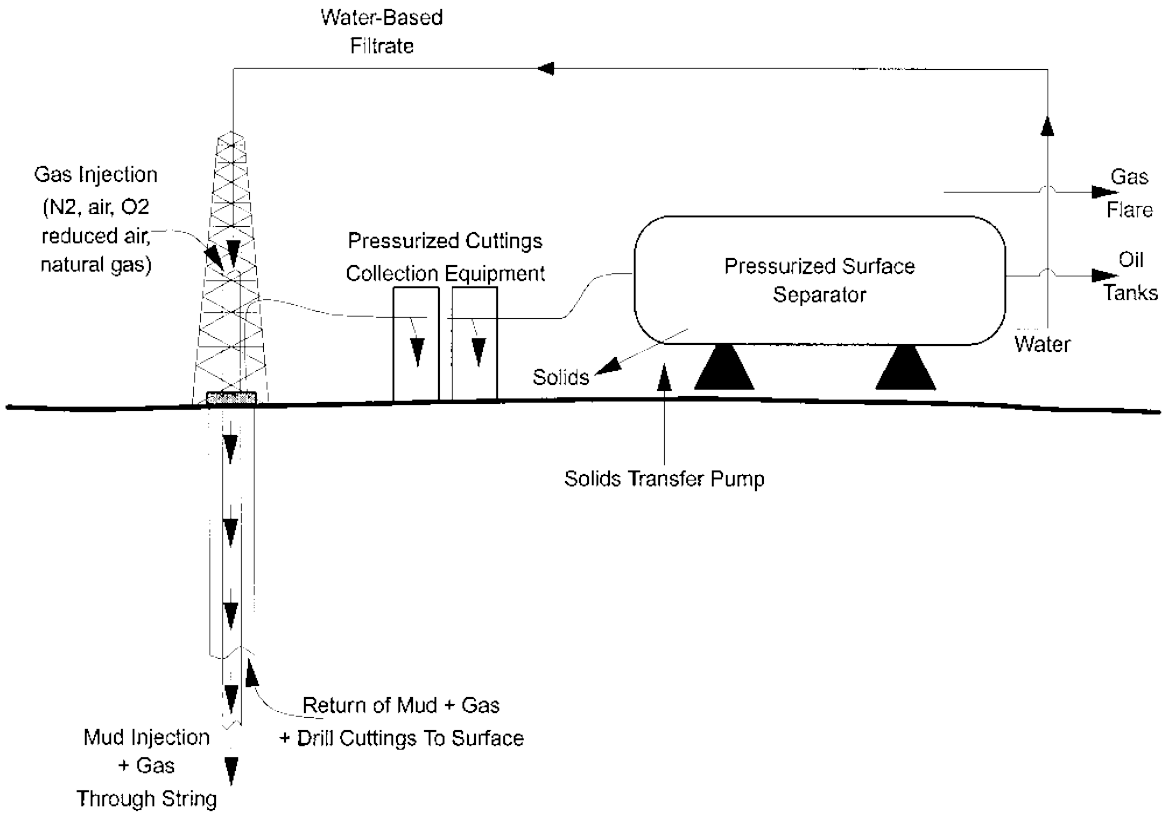


Figure 3: Typical UBD arrangement. [8]

1.2.1 Advantages of UBD

We will in this section highlight some of the advantages of UBD.

Reduction in invasive formation damage:
During conventional overbalanced drilling operations many different types of formation damages occur, UBD reduces the risk of these types of formation damages [8].

- The migration of in-situ fines and clays because of high fluid leak-off velocities in overbalanced conditions.
- Invasion of solids found in the mud into the formation matrix.
- Invasion due to a poorly designed sealing filter cake.
- High permeability- zones present a threat of severe fluid loss to the formation. For example in high permeability sands.
- Potential of an unfortunate reaction between the invaded filtrate and the formation.
- Potential of an unfortunate reaction between the invaded filtrate and in-situ fluids.

Increased ROP (Rate of Penetration): In many UBD operation, the ROP is greater than in conventional drilling operations. The effect is reduced drilling time in horizontal sections, longer bit life and reduced costs. Many problems are avoided with UBD, especially problems with differential sticking [8].

UBD gives a quick indication of productive reservoir zones: In an underbalanced operation the hydrostatic pressure of the circulating system is usually less than the formation pressure. This should provide a net outflow of formation fluids, given sufficient formation pressure and in-situ permeability. Good production flow monitoring can give a good indication of a production zone in the reservoir. During drilling any significant production of oil can provide an early cash netback to pay off some additional costs which might occur during the UBD operation [8].

LWD/MWD with EMT tools (Electromagnetic Telemetry): UBD operations used to have a major disadvantage in that MWD and geo-steering was impossible when gas-charged fluid systems where used. With new technology and development of EMT tools, it is now possibility. EMT transmits in real time downhole information to the surface during drilling, even in underbalanced conditions. This technology has proven very useful in UBD operations [8].

Ability to flow/well -test during drilling: The advantages of the flowing conditions occurring during UBD operations, have led many operators to conduct drawdown tests for evaluation of the productive capacity of the formation and formation properties during drilling operations [8].

1.2.2 Disadvantages of UBD

In this section we will discuss the adverse effects associated with UBD. The main reason for using UBD has to be economically driven, in that the operator sees the advantages exceeding the disadvantages. Some of the disadvantages will be discussed below.

Expense: UBD is more often than not associated with higher cost than conventional drilling, especially offshore and in more remote places. Additional equipment and cost are required when drilling in underbalanced conditions. UBD is expensive when the well is not completed in an underbalanced fashion. Expensive and extensive completion and stimulation treatments are needed in different parts of the operation. These costs can be avoided if the well is drilled in a truly underbalanced fashion. The primary reason for using UBD is to increase the productivity of the well compared to a conventional overbalanced completion. Therefore, if the well is drilled properly, the increased productivity will more than likely make up for the increased drilling costs [8].

Safety Concerns: Many improvements and safety features have been introduced in UBD operations. These have increased the reliability of many UBD operations. But because the wells have to be drilled and completed in a flowing mode, it will always be a source of safety and technical concerns. UBD uses air or processed flue gas as the injected gas. This can cause problems with flammability and corrosion problems, even though it is effective in reducing costs of the operations [8].

Wellbore Stability Issues: Wellbore stability issues have been a major concern in UBD for a long time, especially in poorly consolidated and highly depleted formations [8]. Not being able to keep a continuously underbalanced condition while drilling, completing and during formation damage has also been a big concern in UBD. This has been the main reason for many unsuccessful UBD operations in the past. There have been severe damages associated with the failure to keep 100% underbalanced condition during drilling and completion. Figure 4: a-d illustrates the mechanism of these types of damages. The main idea is that in an underbalanced condition where the formation pressure is greater than the circulating fluid pressure, produces a lack of incentive for the formation of a classic sealing filter cake on the rock surface. Even though this method has many advantages like reduction in invasive formation damage and with respect to differential sticking concerns, there is still a disadvantage when the sealing filter cake is missing. The result of a missing filter cake is that the barrier protection from fluid and solids invasion is gone. An underbalanced formation which is exposed to an overbalanced pressure, either abruptly or gradually, will have a quick and severe invasion of filtrate and solids [8].

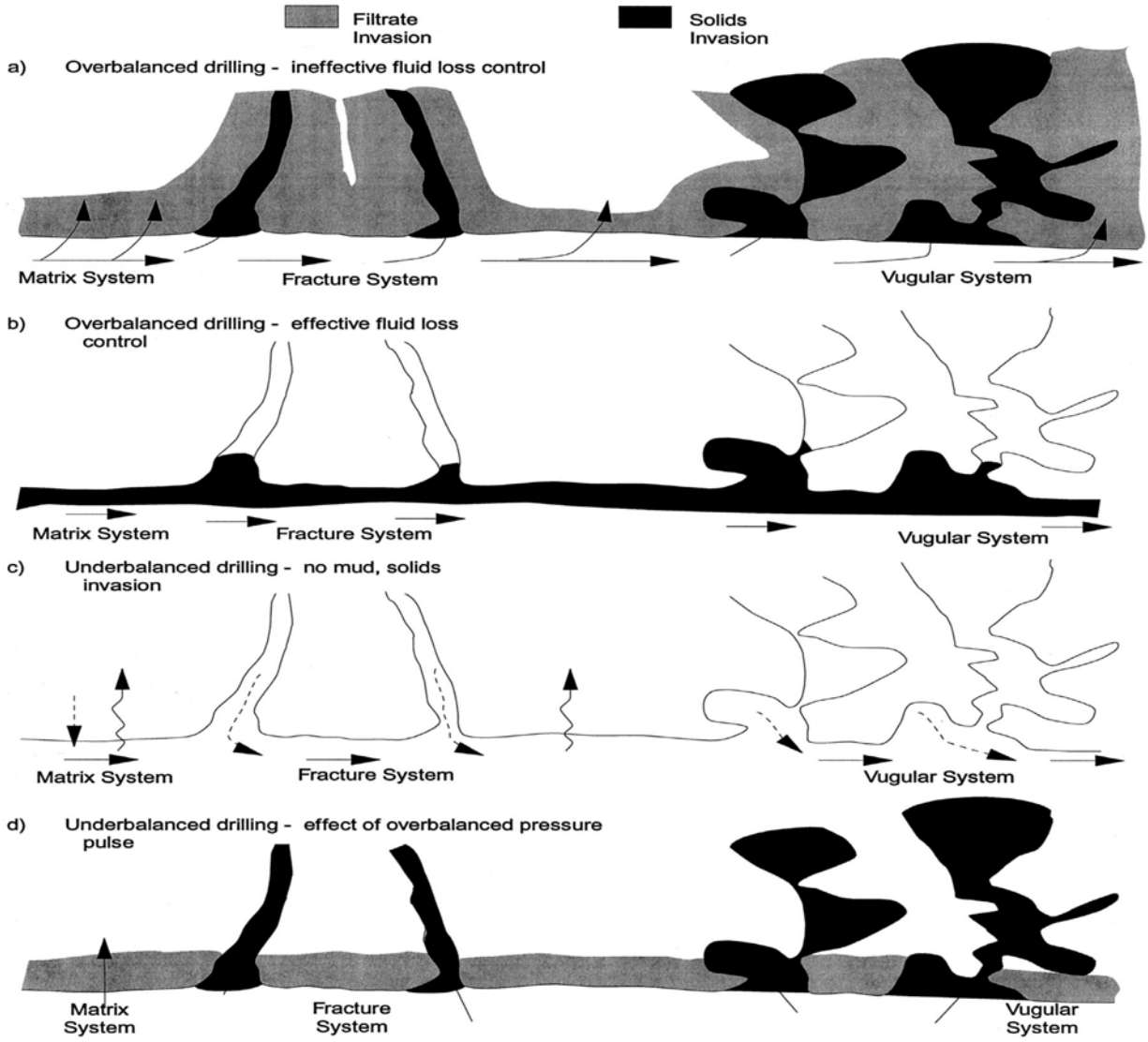


Figure 4: Fluid and solids loss overbalanced and underbalanced drilling operations. [8]

There can be many reasons why the underbalanced condition is lost during drilling. When a rotary rig is utilized, the UB condition is under threat every time gas injection is terminated to make a pipe connection (see Fig 5a). To minimize the effect of these overburden pulses during pipe connections, it is recommended to speed up the connection process and to circulate out to pure gas before every pipe connection (see Fig 5b). Periodic kill jobs for tripping the bit can compromise the underbalanced condition. Hydrostatic kill jobs for conventional mud pulsed logging programs for MWD purposes, are another threat which compromises the UB condition. A human error or having a lack of knowledge when it comes to the original reservoir pressure, can lead to operations in overbalanced conditions. An area of intersection, where multiple reservoir zones at different pressures connect, can lead to crossflow between the reservoir zones. This can lead to underbalanced condition in some high pressure zones, and overbalanced condition in low pressure zones. It is especially true in the presence of permeability barriers [8].

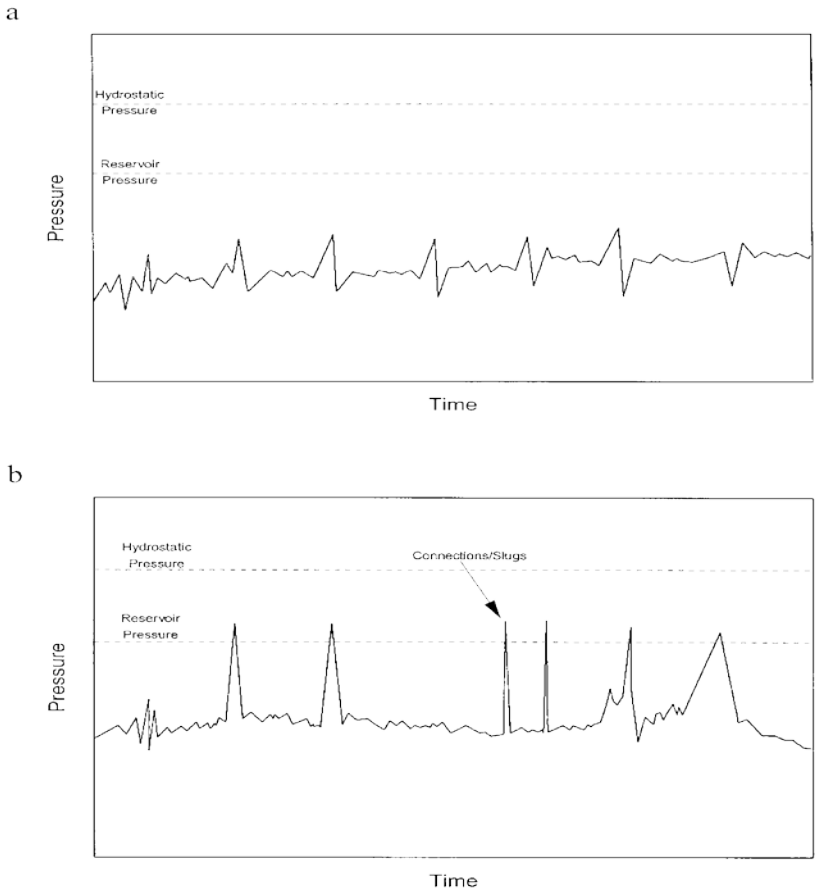


Figure 5: UBD operations. a) Typical BHP survey (N2 circulation before connection). b) Typical BHP survey. [8]

1.3 Managed Pressure Drilling Definition

The International Association of Drilling Contractors has defined MPD as [7]:

"Managed Pressure Drilling is an adaptive drilling process used to precisely control the annular pressure profile throughout the wellbore. The objectives are to ascertain the downhole pressure environment limits and to manage the annular hydraulic pressure profile accordingly. The intention of MPD is to avoid continuous influx of formation fluids to the surface. Any influx incidental to the operation will be safely contained using an appropriate process."

- *"MPD process employs a collection of tools and techniques which may mitigate the risks and costs associated with drilling wells that have narrow downhole environmental limits, by proactively managing the annular hydraulic pressure profile."*
- *"MPD may include control of back pressure, fluid density, fluid rheology, annular fluid level, circulating friction, and hole geometry, or combinations thereof."*
- *"MPD may allow faster corrective action to deal with pressure variations. The ability to dynamically control annular pressures facilitates drilling of what might otherwise be economically unattainable prospects."*

The intent of MPD is not to allow continuous influx of formation fluids to the surface. This intent differs with underbalanced drilling in that in UBD operations the intent is to allow influx to the surface. MPD and UBD share some similarities in equipment and in training for personnel, but both methods are application specific. The drilling engineer will choose the best method to different drilling problems. Contrary to UBD, MPD does not require flares and four-phase separation units. The equipment footprint and outlay for MPD operations is normally not as vast as UBD, however extra training for rig personnel is recommended [7].

1.4 Managed pressure drilling

From the MPD definition we gather that the key word used is "precise control". In offshore drilling we often deal with narrow drilling windows between the pore-pressure and the fracture-pressure gradients. These scenarios typically occur in deep water drilling, where most of the overburden pressure is seawater (fig. 3). These narrow pressure operating windows also occur in fields which used to have more generous drilling windows, but with time, because of production and depletion the pressure gradients change. MPD provides us with precise control in these narrow margin wells by allowing drilling close to the pore-pressure gradient (see fig. 2) [12].

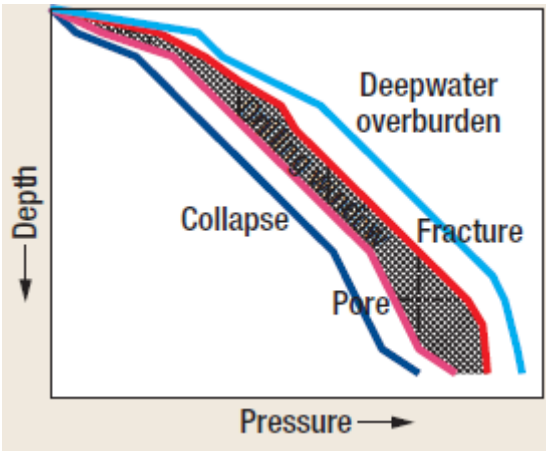


Figure 6: Deepwater Drilling has Narrow Drilling Windows [12]

Managed pressure drilling (MPD) utilizes specific techniques to drill problem wells. Where conventional drilling methods fail, MPD is utilized. The problem wells can be impossible or economically problematic with conventional methods of drilling. MPD reduces the probability of problems arising, such as [7]:

- Lost circulation
- Stuck pipe
- Wellbore instability
- Well Control incidents

To be able to control the annular hydraulic pressure profile of the exposed wellbore, many techniques are used. Proactive control through equivalent mudweight, allows for setting casing at greater depths than in conventional overbalanced drilling. This reduces NPT significantly [7].

2. MPD Technology

Most of MPD is executed by drilling in a closed well loop utilizing a Rotating Control Device (RCD) with a minimum of one drill string Non-Return Valve, and a Drilling Choke Manifold. Manual controlled chokes and microprocessor chokes can also be used depending on the application. Wellbore control is very precise in MPD, assuming that the wellbore is sealed and able to contain the pressure. If this is the case, it is then possible to monitor the pressure throughout the wellbore in real time at the surface. Pressure changes are seen immediately in a closed system. MPD offers more precise control of the annular wellbore pressure profiles; hence influxes and losses are detected immediately. Safety of personnel and equipment during drilling is improved. Drilling economics is improved in MPD due to the reduction of drilling mud costs and reduction in non-productive time [7].

2.1 Approaches

There are two basic approaches in MPD, reactive and proactive [10]:

1. Reactive MPD: uses MPD methods/equipment as a contingency plan to reduce the effect of drilling problems after they happen. The well is planned using conventional drilling methods and MPD methods are activated after unforeseen events occur.
2. Proactive MPD: when the well is designed using MPD methods and equipment to manage the annular pressure profile throughout the wellbore. It is a proactive way to avoid and deal with unforeseen drilling problems that might occur. Proactive MPD can drill the operationally challenging, the economically challenging, and the "undrillable".

There are many proactive variations in manipulating the wellbore pressure profile to reduce or eliminate the effect of drilling problems. Some variations are the PMCD (Pressurized Mud Cap Drilling), CBHP (Constant Bottomhole Pressure), HSE (Returns Flow Control), RC (Reverse Circulation), Casing Drilling, ECD Reduction and the Dual Gradient (DG) Method [7][14].

2.2 MPD: Annular Backpressure System

A problem associated with drilling in small margins is related to loss of circulation during connections. If we recall from conventional drilling, the bottomhole pressure in dynamic conditions, when the hole is being circulated, P_{BH} is equal to P_{Hyd} plus annular friction pressure P_{AF} ($P_{BH} = P_{Hyd} + P_{AF}$). During connections and in static conditions, the equation is reduced to $P_{BH} = P_{Hyd}$. There is a pressure loss for every new connection in a well. The variations in pressure before and during connections can lead to kicks or loss scenarios. MPD technology makes it possible to reduce the effect of these pressure variations and it is possible to drill longer as a result. In deep water drilling, there can be problems with narrow margin between the pore pressure and the fracture pressure. In these circumstances, one can use all the casing sizes, and never reach the target [9].

There are two ways to control the pressure in the well, either by changing the mud weight or mud rheology. The first method is time consuming. The second option is to increase or decrease the drillstring pump rate [9]. MPD introduces pressure control in the well during connections by first closing the annulus with a Rotating Control Device (RCD). The RCD not only closes the system, but also leads the flow to the choke. The choke manifold system or annular backpressure system is then used to control pressure in annulus. By adjusting the choke settings we can achieve the desired BHP [10]. In MPD with backpressure pump, dynamic conditions [9]:

MPD: $P_{BH} = P_{Hyd} + P_{AF} + P_{Choke}$

In static conditions, e.g. during connections [12][9]:

MPD: $P_{BH} = P_{Hyd} + P_{Choke}$

The annular friction pressure term, Is dependent on flow in the well. P_{Choke} depends on the choke opening and the flowrate. During connections, in static condition P_{AF} in the equation disappears. Unlike conventional drilling in MPD the backpressure pump is started to provide sufficient flow across the choke during connections. The backpressure pump compensates for the turning off of the main pump. The choke opening is adjusted to help maintain a stable BHP during connections. A schematic MPD flow overview is given in the figure below [11]:

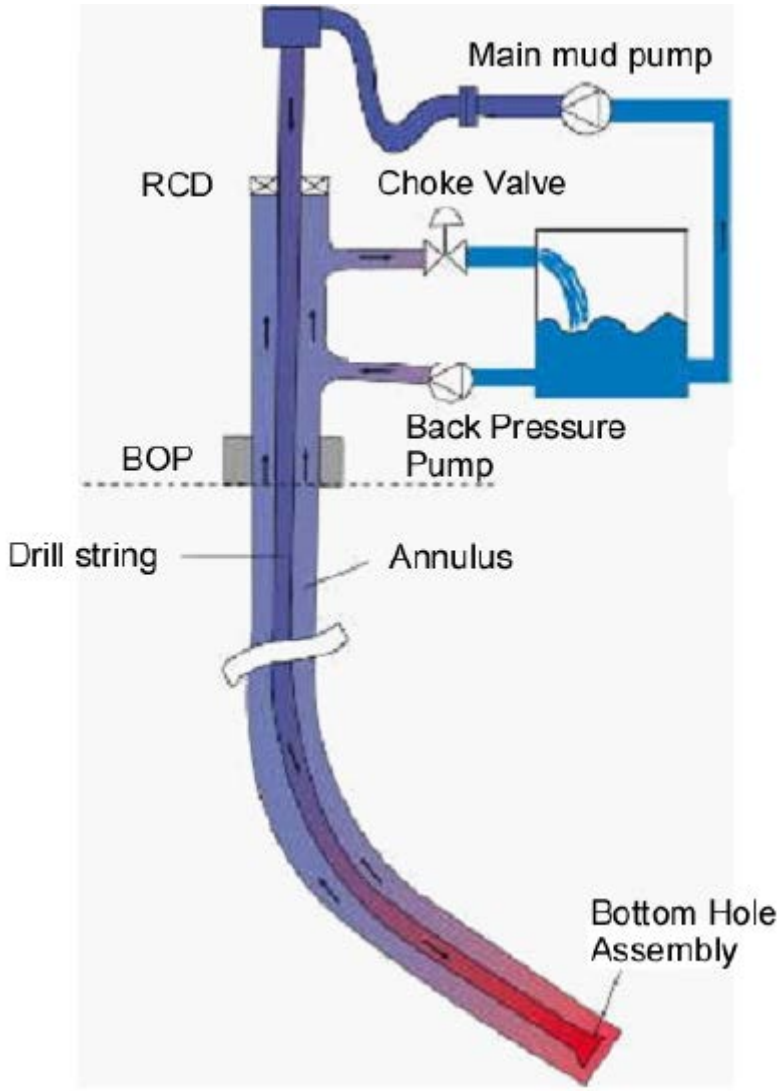


Figure 7: Schematic MPD flow system. [5]

The choke can be controlled manually or automatically. In today's world the most common type is the automatic choke regulation. It requires PID controllers as a regulation technique, this is based on the difference between measured pressure vs required pressure. In addition to that, the automation process is supported by hydraulic simulator models [9].

3. Barriers – Conventional & MPD

In this chapter, a presentation of the comparison of the well barriers in conventional drilling and in MPD is given. Information given in this chapter is taken from NORSOK-D010 [13]. First some definitions:

According to NORSOK a **well barrier** is [13]:
"An envelope of one or several dependent barrier elements preventing fluids or gases from flowing unintentionally from the formation, into another formation or to surface"

Well barrier element [13]:
"Object that alone cannot prevent flow from one side to the other side of itself"

Primary well barrier [13]:
"This is the first object that prevents flow from a source"

Secondary well barrier [13]:
"This is the second object that prevents flow from a source."

Well barriers are essential in drilling operations. They reduce the effect of unforeseen failures and complications. In MPD there are different types of barrier elements compared to conventional drilling. But the purpose is the same. Proper functioning barriers are important in every drilling operation.

3.1 Conventional Drilling – Barriers & Requirements

The conventional drilling barrier elements can be found in [13] and [19].

- **Primary barrier element:**
  - Fluid column
- **Secondary barrier elements:**
  - Formation integrity (kick margin)
  - Casing, casing/cement
  - Wellhead (seal assembly)
  - High pressure riser
  - Drilling BOP
- **Requirement to overbalance [19]:**
  - *"The hydrostatic pressure shall at all times be equal to the estimated or measured pore/reservoir pressure plus a defined safety margin"*

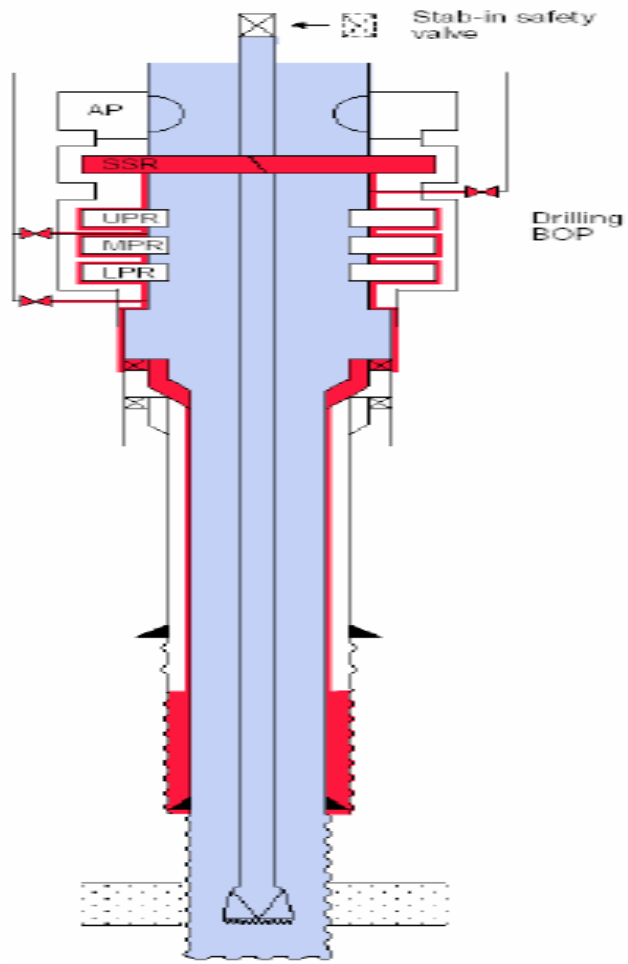


Figure 8: Sharable drillstring- Drilling, coring or tripping. [13]

3.2 MPD - Barriers

The MPD barrier elements can be found in [13] and [19]. There are several common elements in the primary and secondary barriers. If one of the secondary barrier elements fails, there is an immediate serious well control situation [19].

- **Primary barrier elements:**
  - Fluid column
  - RCD
  - Choke system
  - NRV (Non Return Valves)
  - Drill pipe
  - All elements found in secondary barrier
- **Secondary barrier elements:**
  - Formation integrity (kick margin)
  - Casing, casing/cement
  - Wellhead (seal assembly)
  - High pressure riser
  - Drilling BOP

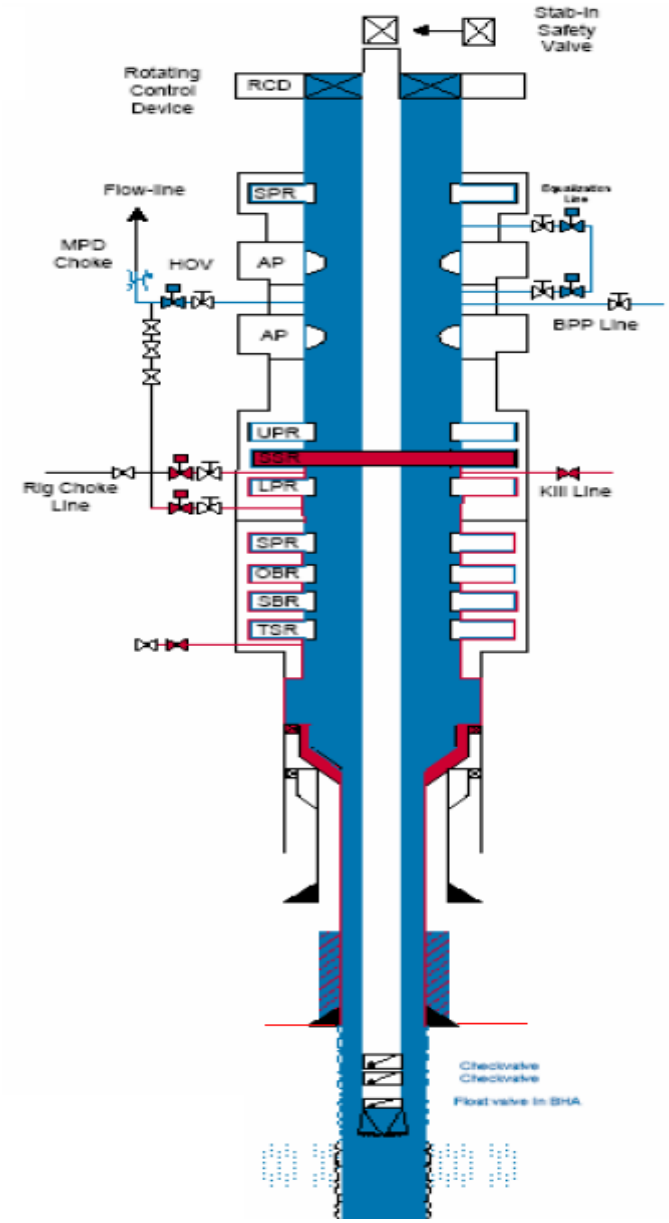


Figure 9: Drilling and tripping of MPD work string in UB fluid [19]

In MPD one finds many of the same well barrier elements as in conventional drilling. One sees that MPD has all the secondary barrier elements found in conventional drilling included in its primary barrier. The well barrier elements unique to MPD are the RCD, choke system and the non-return valves (NRV).

4. MPD Tools

In this section, some important MPD tools will be discussed in some detail. These tools are of importance in making MPD operations possible. In this thesis of propagation of pressure pulses it is important to present the tools essential for pressure control. The key MPD equipment common for most MPD variations can be categorized as [14]:

RCD - Rotating Control Device for floating Rigs:

- External Riser RCD
- Subsea RCD
- Internal Riser RCD (IRRCH)

RCD - Rotating Control Device for fixed Rigs:

- Passive & Active annular seal design
- Marine Diverter Converter RCD
- Bell Nipple Insert RCD

NRV's – Non Return Valves

Choke Manifold Options:

- Manual
- Semi-automatic
- Automatic

4.1 RCD – Rotating Control Device

RCD is the most important tool in MPD. It is the back bone of pressure control in MPD operations. The RCD main function is to maintain a pressure tight barrier in the annulus, and to divert flow coming from the well through the choke manifold to separation equipment. The RCD can be located in different places, depending on the MPD variation and rig type. In production rigs, the RCD is installed on top of the BOP subsea. On the mobile rigs RCD is installed on top of the riser at the surface. The most important feature of the RCD is that it allows for the drill pipe to rotate and still keeping the annulus sealed. Different types of RCD can be categorized into two different categories, the passive rotating devices and active rotating annular preventers [9][14].

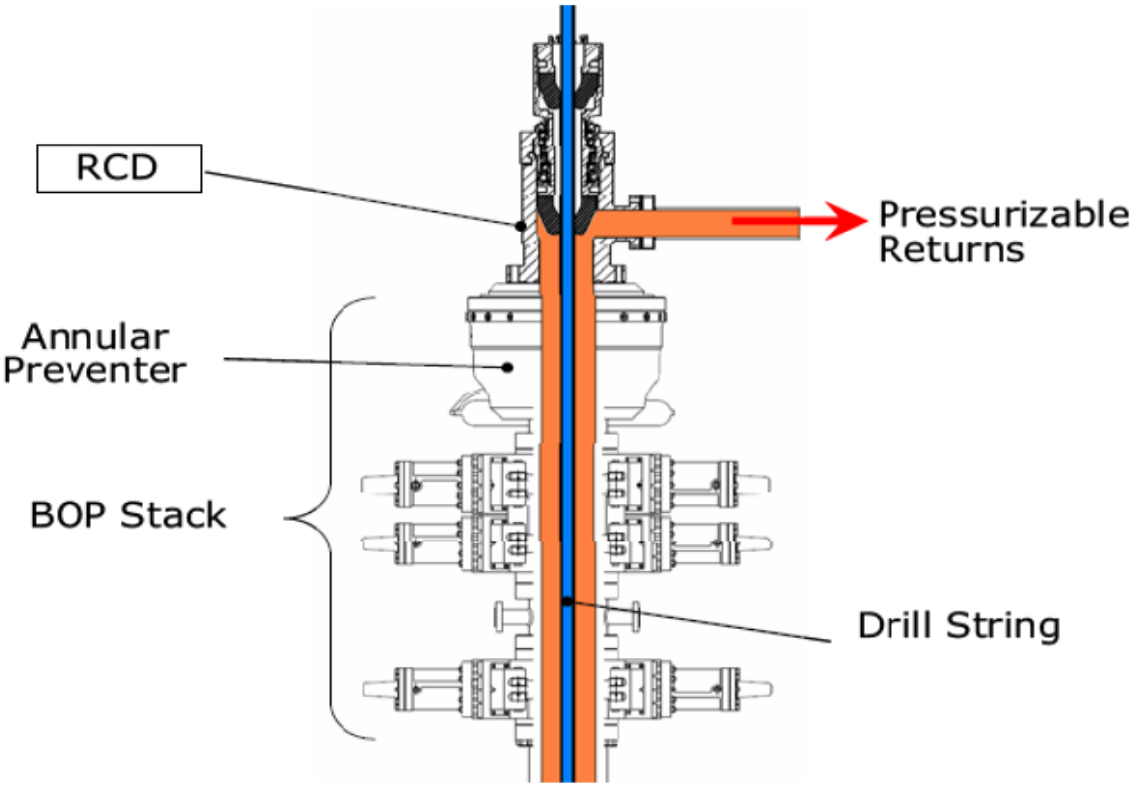


Figure 10: Rotating Control Device [20]

The passive system is the most common in MPD operations. The passive rotating devices are rotating packers with undersized seal elements or stripping rubber. The stripping rubber forms a seal against the drillpipe under zero pressure. When the seal is exposed to the annulus pressures, it is made stronger as a result. This is illustrated in figure 11, where the annulus pressure in red is exerting force to strengthen the seal. The drillpipe can move vertically or rotate in the RCD without compromising the seal. The rubber element in the RCD needs to be changed periodically, due to wear and tear. Depending on the surface pressure, RPM (rotation per minute) and condition of the drillpipe, the decision has to be made whether or not to replace the rubber element. Condition of the drillpipe is important to analyze to get the maximum lifetime out of the rubber [15][17].

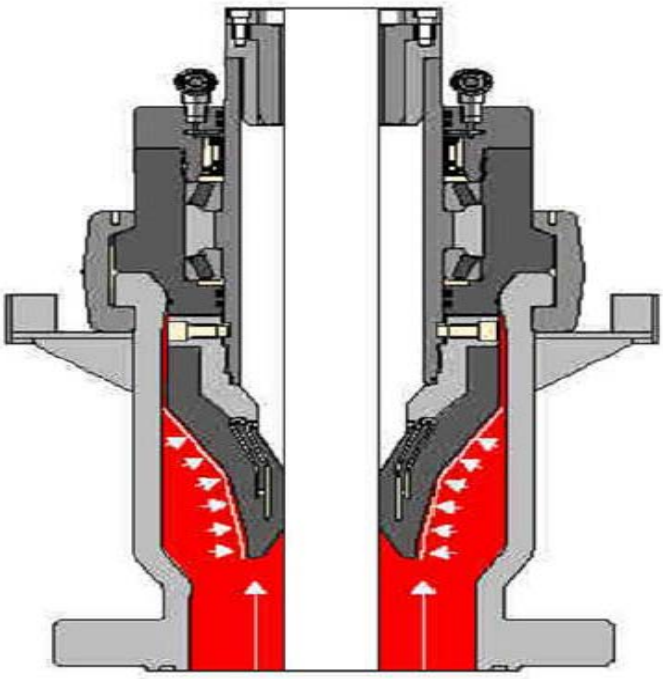


Figure 11: RCD, annulus pressure strengthening the seal. [18]

The active rotating annular preventer is not used much in MPD because it is larger in size and requires more vertical space. It seals against the drillpipe by using hydraulic power, in contrast to the passive RCD system.

4.1.1 RCD – External riser

This tool is used on floating vessels in motion, due to waves. Depending on the maximum possible wave heave and maximum return flow, length and size of the flexible flow line is determined. The pressurized mud cap variation of MPD is possible, partly because the external riser is a part of the riser cap. To create a mud cap situation, we need riser cap to pump high viscous fluids down the annulus. A figure of an external riser is shown below [21].

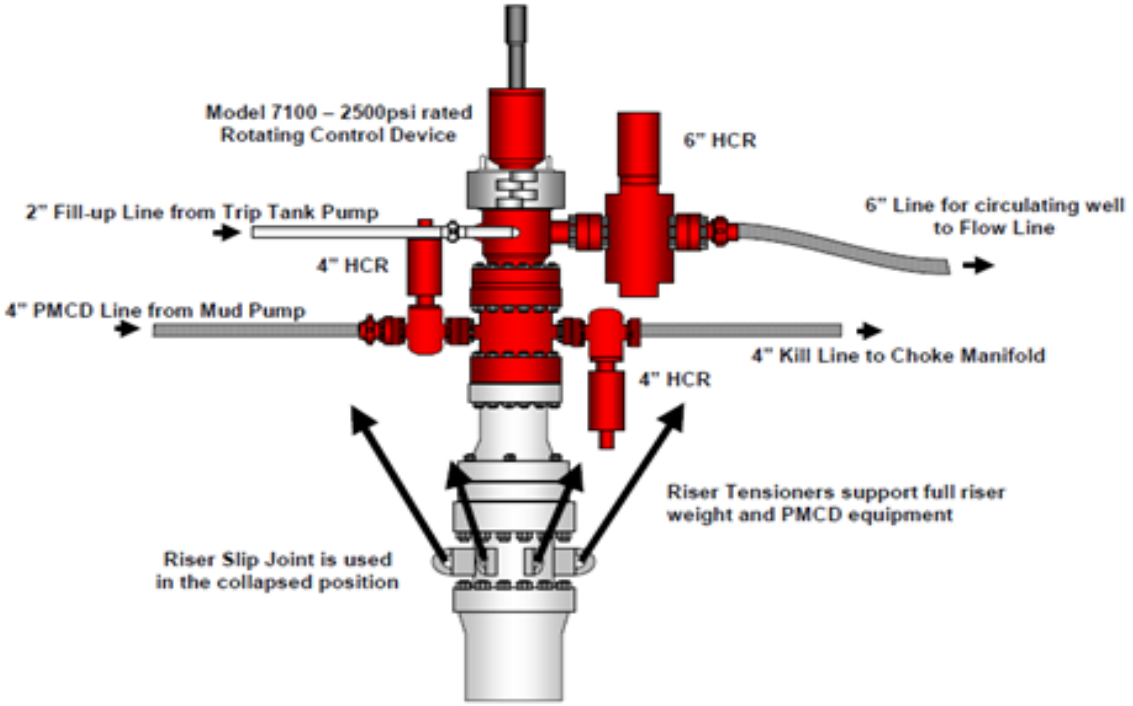


Figure 12: External riser – Floating rigs [14]

4.1.2 RCD – Subsea

The subsea RCD is commonly used in dual gradient drilling variations of MPD with a marine riser, or in riserless drilling [21].

Figure 13: Subsea RCD unit [14]

4.1.3 RCD – Marine Diverter Converter

The marine diverter converter converts a typical marine diverter to rotating diverter [14]. The tool is used when there is no motion between the rig and drillstring. The housing of the marine diverter converter clamps down on RCD. Since a RCD unit is installed inside the marine diverter, it diverts any influx that may occur during drilling to the choke manifold effectively [21].

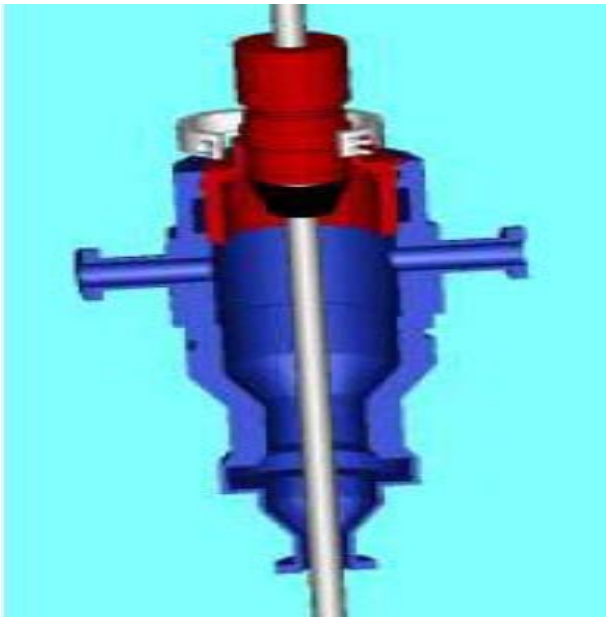


Figure 14: Marine Diverter Converter [14]

4.2 MPD Choke Manifold

The choke manifold system consists of a choke, pressure gauges, a Coriolis flow meter, an advanced control system and a backpressure pump. Not all the equipment cited need necessarily be installed in the manifold system, depending on the vendor of the choke manifold system, different equipment can be subtracted [18]. The chokes are valves with a main purpose to precisely control the flow. In MPD the choke manifold is used to control annular backpressure by adjusting the choke opening. To maintain the backpressure, a sufficient volume of mud flow must flow through the choke. When this flow decreases, the coke opening is reduced to keep the same pressure. The reduction in flow area makes the fluid velocity to increase. When there is a need to reduce the pressure, the choke is operated the opposite way, hence the choke opening is increased. If there is no flow, the choke closes to trap the pressure [22].

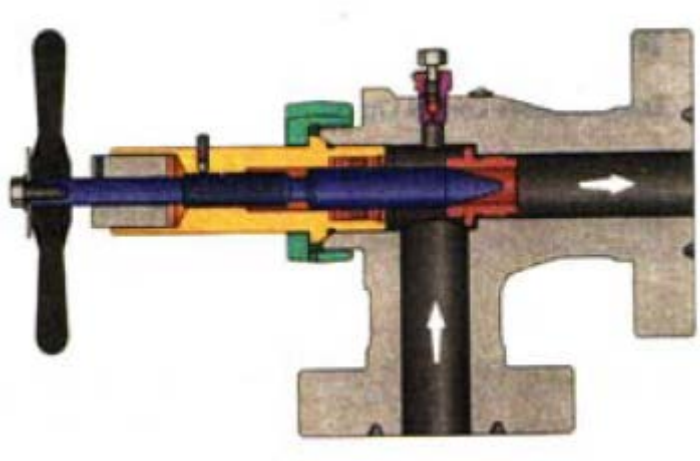


Figure 15: Illustration of choke's functionality [9]

The chokes are used as secondary well control equipment in conventional drilling methods. In MPD is the purpose not to allow continuous influx from the formation to the surface, thus the functionality of the MPD chokes is more for pressure control and not so much for flow control. Since the MPD choke system is used continuously throughout the drilling operations, it is more a part of the drilling system than the well control system [23].

The choke comes in three types: Manually controlled choke, semi-automatic or fully automatic. There are significant advantages in using automatic choke control compared to the manually controlled choke. During connections, the choke opening is to be gradually reduced at the same time as the pump rate is gradually reduced. This is done to maintain constant pressure. As the choke gradually closes, the backpressure increases with BHP. Meanwhile the reduction of the pump rate decreases the BHP. The idea is to keep a constant BHP and reduce the BHP fluctuations. The figure below shows an example of such a procedure. It stands to reason that to try to keep a constant BHP by manually controlling the choke is a difficult task. Human errors are a real challenge [24]. The first wells drilled in the NCS were drilled with manual chokes. The choke was controlled by the choke operator who communicated with the driller [9].

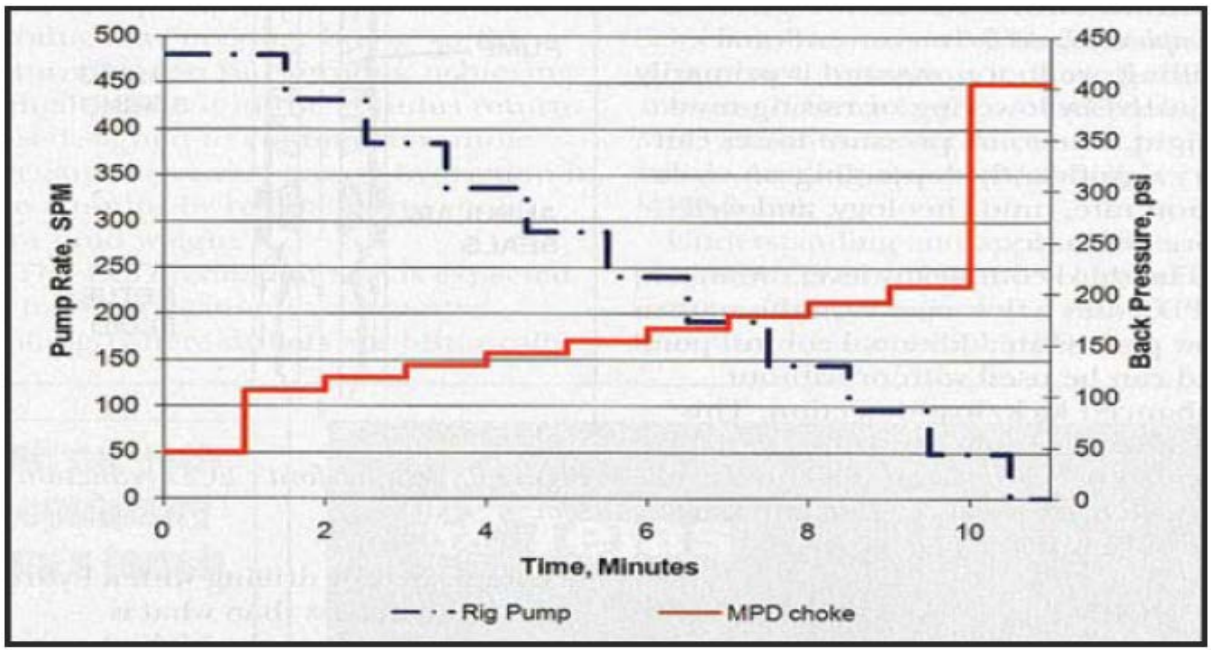


Figure 16: Plot of Pump rate and Backpressure, Manual choke, MPD connection. [24]

The semi-automatic choke can be categorized as a manual choke, but incorporates in addition automatic control of the backpressure [21]. Example of a semi-automatic choke manifold can be seen on the figure below.

Figure 17: Semi-Automatic Choke Manifold [21]

4.2.1 Choke Control System

The fully automatic chokes are hydraulically controlled with a PLC (Programmable Logic Controller) system. PLC automatically controls the choke opening to computed setpoints with the help of a dynamic hydraulic flow model. Both choke opening and pumprate can be controlled by the system. The PLC system has real-time control system software and gets real-time data such as flow rate data, surface pressure, temperature and data from MWD. The control software uses these real-time data to automatically control the choke manifold to maintain a certain choke set point. The software signals the PLC who controls a mechanical device which in turn adjusts the choke [24]. The hydraulic model assisting the PLC, continuously updates the calculations as new measurements and data become available. The new calculations lead to new choke setpoints, and in this way is the BHP kept approximately constant.

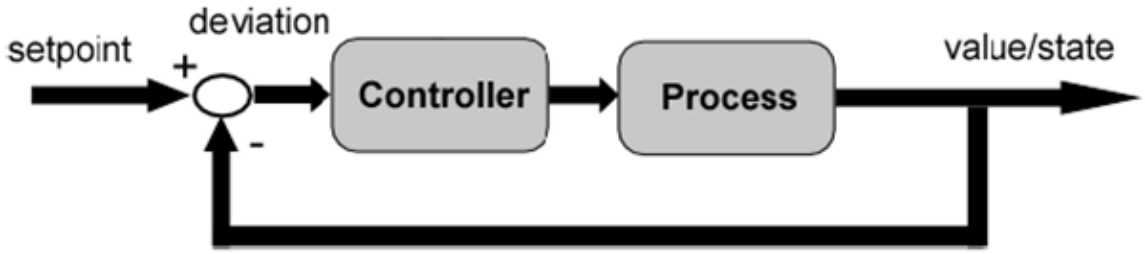


Figure 18: "The closed loop control concept. The driller feeds the automation system with a setpoint and the closed loop algorithm compensates for the deviation from this setpoint."[26]

It is important to calibrate the dynamic hydraulic flow model with measured BHP to make sure it is accurate. The pressure measurements are transferred to the surface with mud pulse telemetry usually. There can be a significant delay in the response time for the mud pulse, depending on length, gas fraction and pump rate. This "time lag" can create complications for the hydraulic model, moreover it can produce inaccuracy in the output. Because of inaccuracies of the input data and the limitations set by the processing capacity of the computers, the control systems need to have a manual override function. The MPD operator must have access to intervene manually in case of emergencies [22].

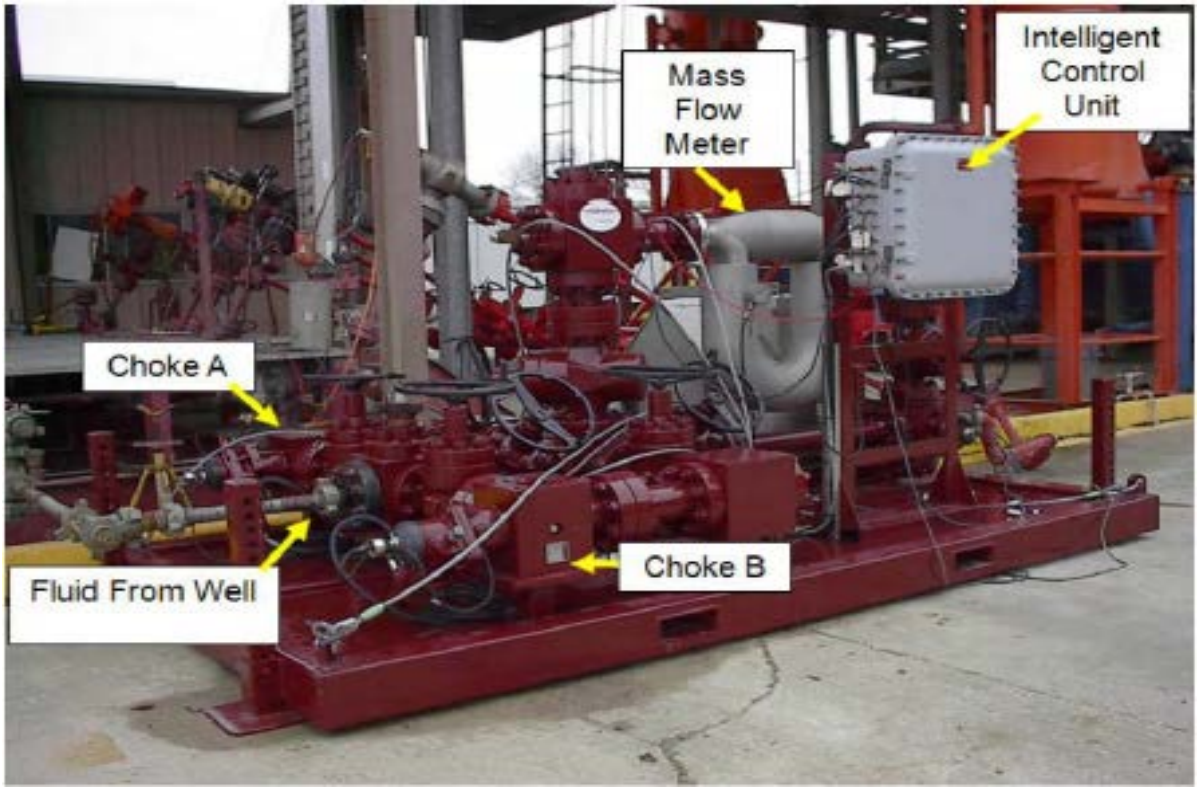


Figure 19: Fully Automatic Choke Manifold [21]

4.3 ECD reduction tool

The ECD reduction tool is used in wells where friction or ECD is a problem. It is common in deepwater drilling operations. The idea is to place a motor in the well that pumps fluid upwards. Operation of the system creates dual gradients in the annulus [9].

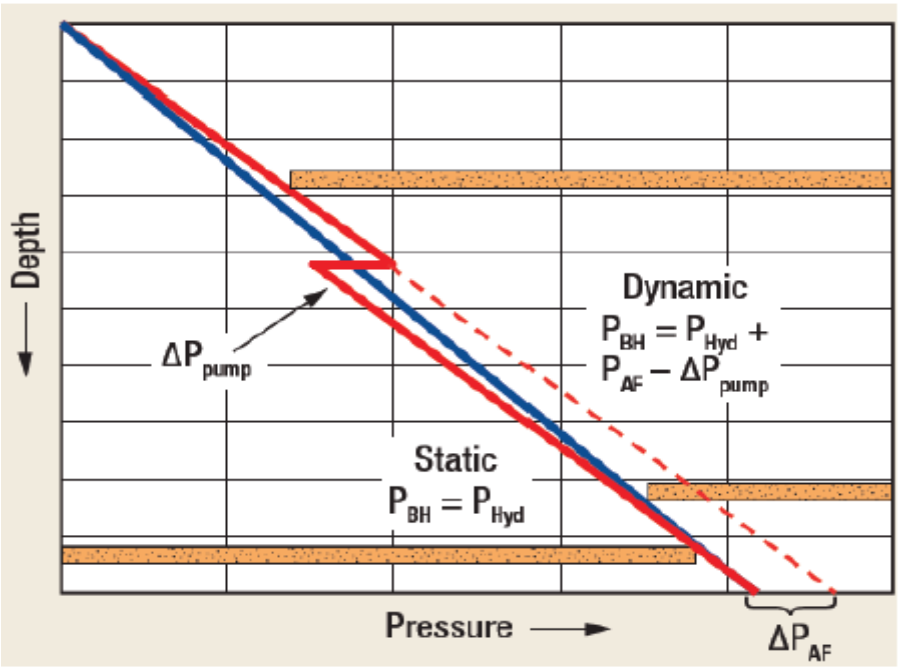


Figure 20: ECD Reduction [12]

In deepwater drilling many casings have to be run in the well in order to reach the intended target. To run many casings reduce the hole size and can make the frictional pressure a problem. The ECD reduction tool is also effective in long reach wells where the frictional pressure loss is high. If one considers using a single density drilling fluid, the downhole motor can pump fluid upwards creating energy that can create a change in the annular pressure profile. Backpressure is then not needed in static conditions. Moreover, the BHP will remain constant [21].

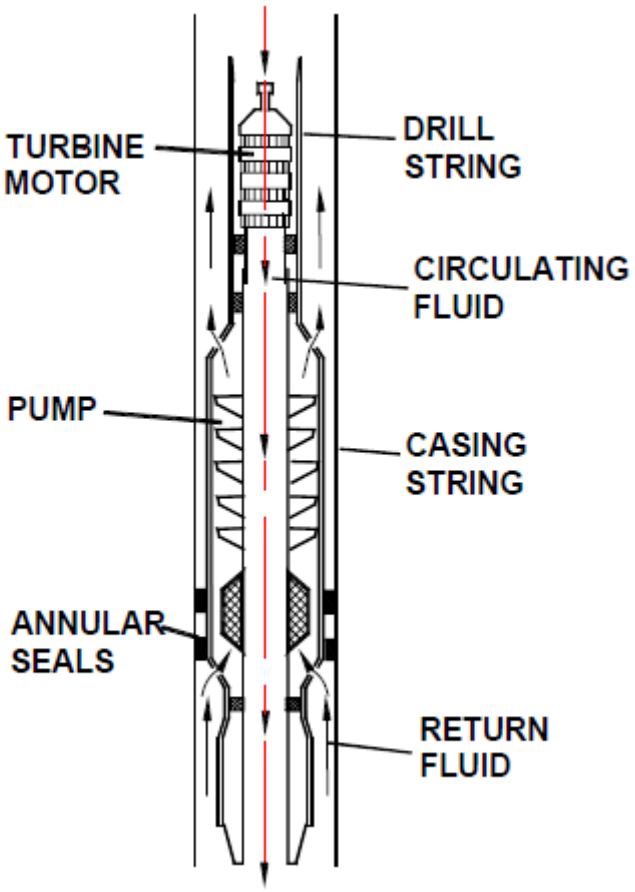


Figure 21: Schematic of ECD reduction tool [16]

4.4 CCS – Continuous Circulation System

The continuous circulation system makes it possible to control the annular pressure profile during connections by maintaining the ECD. It is also possible to maintain circulation during connections. As soon as the CCS is installed it is calibrated and tuned to the rig. In HPHT wells it is common to have large and unpredictable temperature changes downhole. In these conditions there is a possibility for mud to become over heated and as a result change the mud properties. With changed mud properties it can be difficult to interpret trends in other parameters. It is also difficult to accurately control the choke as a result for the BHP set points. Meanwhile, if the circulation in the well is maintained, the temperature changes will not affect the mud as much, thus making it easier to control choke. With continuous circulation will the ECD never vanish, thus reducing the pressure fluctuations during

connections. Another advantage with continuous circulation is improved hole cleaning and reduces connection gas, resulting in a larger drilling window because the mud weight can be adjusted down equivalent to initial pore pressure [25].

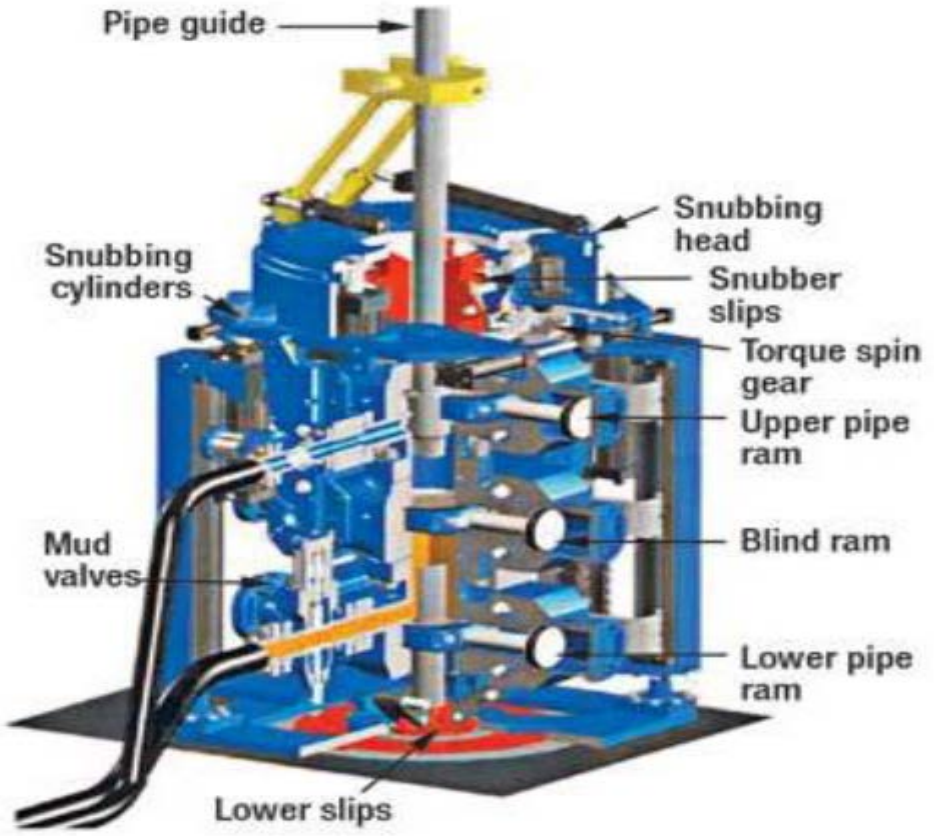


Figure 22: CCS- Continues circulating system [12]

4.5 NRV – Non- Return Valves

The NRV's or the "drillstring float valves" are important MPD equipment. NRV's are installed to avoid U-tubing and to only allow flow in the desired direction. When one applies backpressure during connections, a U-tube effect might occur between the drillstring and annulus. U-tubing might lead to drilling fluid flowing back up the drillstring, which can carry with it cuttings plugging MWD tools or in worst case blow out the drill pipe. NRV's are installed in the BHA, usually right above the mud motor. Two or more NRV's are installed for redundancy. To avoid tripping the whole drillstring when a NRV is to be replaced, using wireline retrievable floats is back-up solution [17].

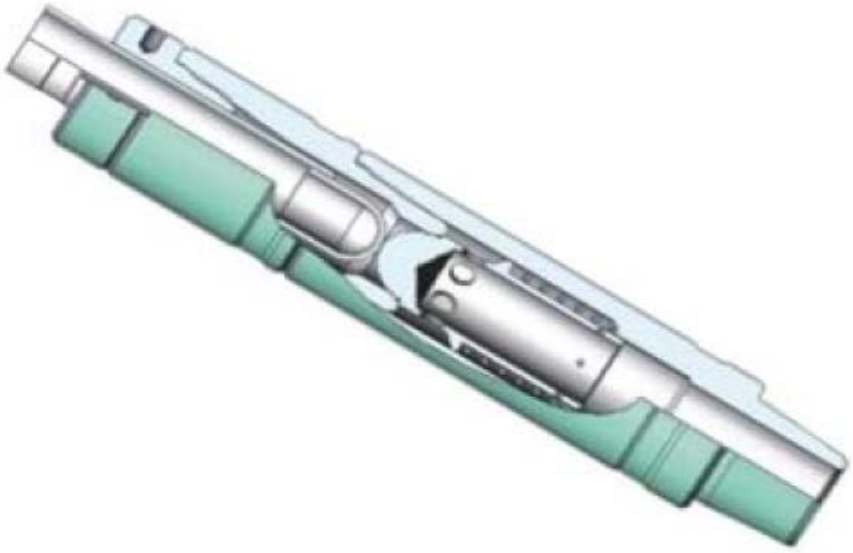


Figure 23: NRV (inside BOP) [23]

4.6 Coriolis Flow Meter

In MPD operations it is important to measure the flow, and the Coriolis flowmeter does just that. It is a high accuracy mass flow meter, put together with a transducer and a transmitter. The transducer measures the fluid moving through the flowmeter, while the transmitter transmits the information to the control system. The measurements include mass flow, volumetric flow, density and temperature data [9][18].

Figure 24: Coriolis flow meter [9]

5. The Transient Drift Flux Model

5.1 Conservation Laws

Two-phase flow models are used in the petroleum industry among other things to evaluate transient flow responses of drilling operations. Because the two-fluid models are quite complex, other simpler drift flux models are used. The Drift Flux model has its roots in the laws of conservation for two phase flow, and its goal is to describe the characteristics of flow in pipes or wells. The Drift Flux model is derived from the Two-phase flow model with the addition of liquid and gas momentum equations and the energy equations. When the addition is computed, the results are mixture momentum (liquid and gas) and energy equations. Assuming there is no significant temperature change (isothermal flow), eliminates the energy component in the equations. The drift flux model for two phase flow, where the flow area is constant and in isothermal conditions, is given below [1]:

$$\partial_t [\alpha_g \rho_g] + \partial_x [\alpha_g \rho_g v_g] = \Gamma_g$$

$$\partial_t [\alpha_l \rho_l] + \partial_x [\alpha_l \rho_l v_l] = \Gamma_l$$

$$\partial_t [\alpha_g \rho_g v_g + \alpha_l \rho_l v_l] + \partial_x [\alpha_g \rho_g v_g^2 + \alpha_l \rho_l v_l^2 + p] = -q$$

Where the nomenclature is [1]:

- α_l and α_g = phase volume fractions, ($\alpha_l + \alpha_g$) = 1.
- ρ_l and ρ_g = phase densities of liquid and gas.
- v_l and v_g = phase velocities of liquid and gas.
- Γ_l and Γ_g = interphase mass transfer (liquid and gas).
- p = common pressure.

The right hand side of the equation: q is a source term, ($q = F_w + F_g$), representing the external forces acting on the fluids. F_w is the wall friction. F_g represents the gravitational forces, $F_g = g(\rho_l \alpha_l + \rho_g \alpha_g)\cos\theta$, where g is the gravity and θ is the inclination.

Assuming there is no mass transfer between the two phases,

$$\Gamma_l = \Gamma_g = 0$$

The equation system takes the following conservative vector form [1]:

$$\partial_t w + \partial_x F(w) = G(w)$$

Where,

$$w = \begin{pmatrix} \alpha_l \rho_l \\ \alpha_g \rho_g \\ \alpha_l \rho_l v_l + \alpha_g \rho_g v_g \end{pmatrix}, F(w) = \begin{pmatrix} \alpha_l \rho_l v_l \\ \alpha_g \rho_g v_g \\ \alpha_l \rho_l v_l^2 + \alpha_g \rho_g v_g^2 + p \end{pmatrix}, G(w) = \begin{pmatrix} 0 \\ 0 \\ -q \end{pmatrix}.$$

It can also be written in the form:

$$\partial_t \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} + \partial_x \begin{pmatrix} v_l w_1 \\ v_g w_2 \\ v_l^2 w_1 + v_g^2 w_2 + p(w_1, w_2) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -q \end{pmatrix},$$

Where $w_1 = \alpha_l \rho_l$, $w_2 = \alpha_g \rho_g$ *and* $w_3 = \alpha_l \rho_l v_l + \alpha_g \rho_g v_g$ are the conservative variables. Pressure is a passive variable derived from the conservative variables w_1 and w_2.

5.2 Closure Laws

For computational purposes certain closure laws are required to be able to solve the drift model. These can be quite complex when real flow conditions should be simulated. These laws can are derived through experiments or through theoretical knowledge. Closure laws are used to close a model of certain unknowns. A system with certain unknowns should have the same number of corresponding equations so that the system can be solved [1].

In the drift flux model for two-phase flow certain closure laws are assumed, like slip, density and friction.

Slip law:

$$v_g = K v_{mix} + S$$

Where $v_{mix} = \alpha_l v_l + \alpha_g v_g$ is the mixture average velocity. K and S are flow-dependent parameters. K = 1.2 and S = 0.55 are example values.

Liquid and gas densities:

Assuming the liquid density is given by the equation [1]:

$$\rho_l = \rho_{l,0} + \frac{p - p_{l,0}}{a_l^2}$$

Where a_l = 1500 m/s is the velocity of sound in liquid phase. Assuming $\rho_{l,0}$ = 1000kg/m^3 and $p_{l,0}$ = 1.0 bar. The original a_l was given as 1000 m/s but it is changed to 1500 m/s which is more realistic for water in MPD calculations.

The gas density is given the assumed form:

$$\rho_g = \frac{p}{a_g^2}$$

Where a_g = 316 m/s is the velocity of sound in the gas phase. P has the unit pascal.

Volume phase fractions:

The volume fractions are given by the relation [1]:

$(\alpha_l + \alpha_g) = 1$

Source term q:

q is a source term, $(q = F_w + F_g)$, representing the external forces acting on the fluids. F_w is the frictional pressure loss. F_g represents the gravitational forces, $F_g = g(\rho_l \alpha_l + \rho_g \alpha_g)\cos\theta$, where g is the gravity and θ is the inclination. In the AUSMV scheme a more complex frictional pressure loss model is used. The frictional pressure loss model used is [3]:

$$F_w = \frac{2 f \rho_{mix} v_{mix} abs(v_{mix})}{(d_{out} - d_{in})}$$

Where, d_{out} and d_{in} are the outer and inner diameter of the annular flow area. f is the friction factor. The friction factor is determined based upon the type of flow. Type of flow can be laminar or turbulent flow. From the Reynolds number, one can determine the transition between the two flow regimes, through the relation [3]:

$$\mathrm{Re} = \rho_{mix}\, abs(v_{mix})(d_{out} - d_{in}) / \mu_{mix}$$

Laminar flow occurs when the Reynolds number is below 2000. The friction factor for laminar flow is f = 24/Re where Re<2000. For turbulent flow the friction factor is given by $f = 0.052\,\mathrm{Re}^{-0.19}$ where Re > 3000.

5.3 Eigenvalues

The drift flux model is a hyperbolic system of conservation laws. It describes propagation of mass waves and pressure pulses. A mass wave could e.g. be a kick migrating upwards. Sonic waves are generated when flow rates are changed or if we change valve openings in the flow path. Eigenvalues and eigenvectors can be of to define the conditions at the flow boundaries [1].

For this two-phase model is it worthy to note some of the features. Under the condition $\alpha_g \rho_g << \alpha_l \rho_l$ and assuming that the liquid is incompressible, it can be derived that an approximate sound velocity for two-phase regions where $\alpha_g \in (0,1)$ is given as [1]:

$$\omega^2 = \frac{p}{\alpha_g \rho_l (1 - K\alpha_g)}$$

The corresponding eigenvalues are:

$$\lambda_1 = v_l - \omega, \quad \lambda_2 = v_g, \quad \lambda_3 = v_l + \omega$$

The first and the third eigenvalues correspond to the pressure pulses propagating downstream and upstream. The second eigenvalue is there to describe the wave speed of the linear gas volume wave traveling downstream. For pure liquid regions ($\alpha_g = 0$), the eigenvalues are [1]:

$$\lambda_1 = v_l - a_l, \quad \lambda_3 = v_l + a_l$$

Where a_l is the velocity of sound in the liquid phase. These eigenvalues correspond to the pressure pulses propagating upstream and downstream. It is similar for pure gas regions ($\alpha_g = 1$):

$$\lambda_1 = v_g - a_g, \quad \lambda_3 = v_g + a_g$$

Where a_g is the velocity of sound in the gas phase. Pressure pulses propagate in a well when there is a change in the flow rates or change in the choke valve openings. These pulses will move with the same velocity as the eigenvalue values [1].

5.4 Discretization

For one to apply the conservation laws and closure laws given in previous sections, it is then paramount to first divide the well into a N number of segments (1,…, i -1, i, i +1,…, N**)** (fig 2). This process is discretization. In the beginning of the calculation there is a known initial stage where all flow variables are known. Then each cell is updated in time using the equations shown in (fig 3). One solves the equations for each segment. The flow variables are considered to be constant in each segment. The discretization makes sure that local variations in pressure and temperature are being reflected in the density calculations, which again is used to derive the total hydrostatic pressure in the well. Friction is also calculated locally for each segment in the discretization process [3].

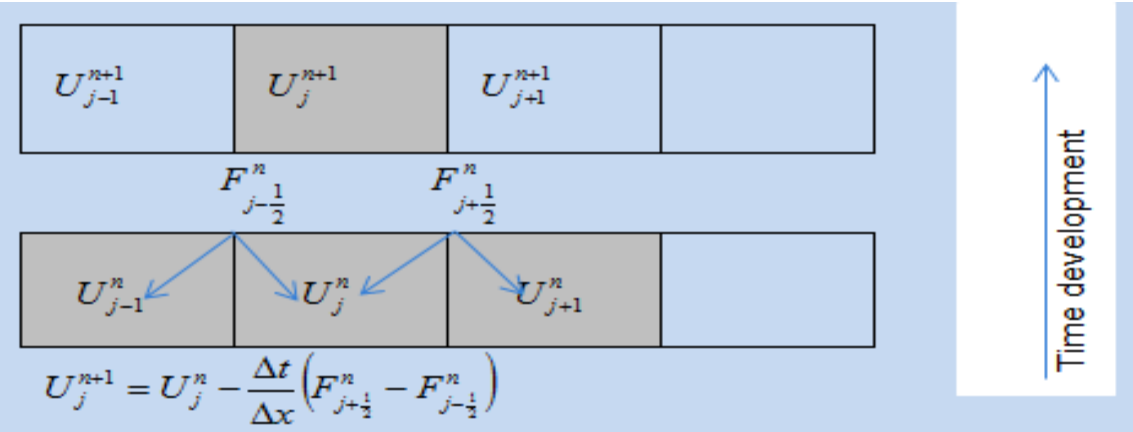


Figure 25: Stage wise discretization in an explicit AUSMV scheme [3]

When solving the drift flux equations, one has to decide which numerical strategy to use. One can use either implicit or explicit numerical schemes for discretization [2]. An implicit scheme finds the variables based on "new" values [4]. An implicit scheme is faster, but it is more complex to implement. It`s faster since the time steps (Δt) are only limited by the length of the segments (Δx) and the maximum velocity of the fluid (v_{max}) [2].

$$\Delta t = \frac{\Delta x}{v_{\max}}$$

The explicit scheme solves the equations by calculating new values based on old values [4]. For an explicit scheme are the time steps (Δt) limited by length (Δx), the maximum velocity (v_{max}), and the speed of sound (c) [2].

$$\Delta t < \frac{\Delta x}{v_{\max} + c}$$

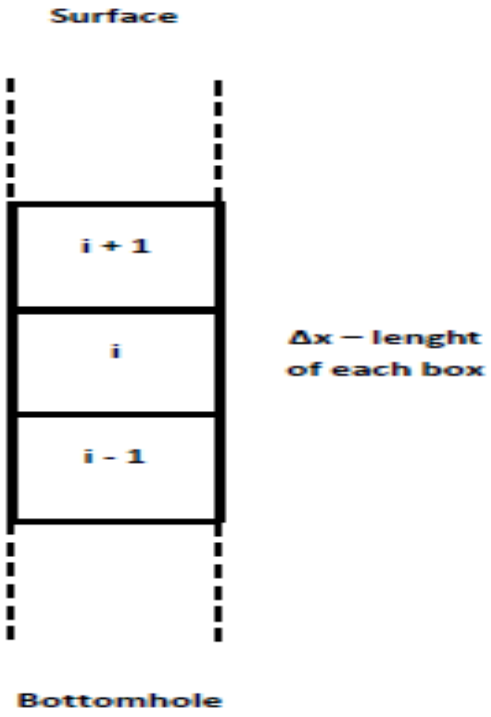


Figure 26: Well discretization [4]

6. AUSMV Scheme

6.1 AUSMV Scheme

The AUSMV scheme is a hybrid flux-vector splitting scheme (FVS) and stands for Advection Upstream Splitting Method, the V in AUSMV stands for the velocity splitting functions ($\tilde{V}^{\pm}$). The scheme is used to analyze two-phase flow in pipes or wells [1]. The AUSMV scheme is based on the hybrid flux-vector splitting scheme (FVS). The numerical FVS flux is given as [5]:

$$F^{FVS}_{j+1/2}(w_L, w_R) = (\alpha_l \rho_l)_L \Psi^{+}_{l,L} + (\alpha_l \rho_l)_R \Psi^{-}_{l,R} + (\alpha_g \rho_g)_L \Psi^{+}_{g,L} + (\alpha_g \rho_g)_R \Psi^{-}_{g,R} + (F)_{j+1/2}$$

Where $F_P = (0,0,p)^T$, and for liquid phase we have [5]:

$$\Psi^{+}_{l,L} = \Psi^{+}_{l}(v_{l,L}, c_{j+1/2}), \qquad \Psi^{-}_{l,R} = \Psi^{-}_{l}(v_{l,R}, c_{j+1/2})$$

Where

$$\Psi^{+}_{l}(v,c) = V^{+}(v, c_j)\begin{pmatrix}1\\0\\v\end{pmatrix}, \qquad \Psi^{-}_{l}(v,c) = V^{-}(v, c_j)\begin{pmatrix}1\\0\\v\end{pmatrix}$$

Similarily for gas phase, we have [5]:

$$\Psi_{g,L}^{+} = \Psi_{g}^{+}(v_{g,L}, c_{j+1/2}), \qquad \Psi_{g,R}^{-} = \Psi_{g}^{-}(v_{g,R}, c_{j+1/2})$$

Where

$$\Psi_{g}^{+}(v,c) = V^{+}(v,c_j)\begin{pmatrix}0\\1\\v\end{pmatrix}, \qquad \Psi_{g}^{-}(v,c) = V^{-}(v,c_j)\begin{pmatrix}0\\1\\v\end{pmatrix}$$

The velocity-splitting functions ($V^{\pm}$) are given as [5]:

$$V^{\pm}(v,c) = \begin{cases} \pm\dfrac{1}{4c}(v \pm c)^2 & if\ |v| \le c \\ \dfrac{1}{2}(v \pm |v|) & Otherwise. \end{cases}$$

The velocity-splitting functions in FVS schemes ($V^{\pm}$) is replaced in AUSMV by the more general ($\tilde{V}^{\pm}$). To be more precise, for AUSMV is $\tilde{V}^{\pm}$ given as [5]:

$$\tilde{V}^{\pm}(v,c,\chi) = \begin{cases} \chi V^{\pm}(v,c) + (1-\chi)^{\frac{v \pm |v|}{2}} & |v| \le c \\ \dfrac{1}{2}(v \pm |v|) & Otherwise \end{cases}$$

The pressure term F_P is treated as for FVS. Here is v fluid velocity, c is the velocity of sound, and the parameter χ is a constant that is chosen to get "good" numerical fluxes. Assuming [5]:

$$\chi_L = \alpha_R, \qquad \chi_R = \alpha_L$$

The AUSMV scheme or FVS gives an explicit solution to the drift-flux laws, hence the variables are found based on "old" values. When the discretization is done, and the well is divided into small segments, By replacing ($V^{\pm}$) with the more general ($\tilde{V}^{\pm}$) in AUSMV, together with weighting functions χ_L and χ_R ,one finds the variables at the new time step (n+1) in the following way [5]:

$$w_{i,j}^{n+1} = w_{i.j}^{n} - \frac{\Delta t}{\Delta x}(F_{j+\frac{1}{2}}^{AUSMV} - F_{j-\frac{1}{2}}^{AUSMV}) - \Delta t\, q_i^n$$

Where q_i is the sum of the external forces, and $w_{i,j}$ is the mass conservative variable and momentum conservative variable for I = 1, 2, 3.

F^{AUSMV}, the mass flux can be represented by [1]:

$$F^{AUSMV}_{j+\frac{1}{2}} = v\,\rho(p)\frac{\alpha_L + \alpha_R}{2} - \frac{1}{2}|v|\,\rho(p)[\alpha_R - \alpha_L]$$

Were $_R$ and $_L$ are representing the right and left boundary of the flux boundaries. Since the fluxes are treated explicit in the AUSMV, one has a restriction on which time step is allowed to be used in the calculations. The time steps are limited by the CFL criterion (Courant-Friedrichs-Lewy) [1]:

$$\Delta t = CFL\frac{\Delta x}{\max(|\lambda_1|,|\lambda_2|,|\lambda_3|)}$$

Where λ_i are the eigenvalues. When one calculates fluxes using the AUSMV scheme it is important to note that it is required to have information of the sound velocity.

When dealing with wells and pipelines, flow area might change. The flow area discontinuity can also be taken into account in the AUSMV scheme. The flow area discontinuity is put in the center of the numerical segment (box) and mass conservation is used [5]:

$$(A\alpha_l\rho_l v_l)_{Left} = (A\alpha_l\rho_l v_l)_{Right}$$

$$(A\alpha_g\rho_g v_g)_{Left} = (A\alpha_g\rho_g v_g)_{Right}$$

If one assumes the pressure, average phase volume fractions and corresponding densities are constant across the discontinuity, one can derive a modified form of the numerical scheme, and it is as follows [5]:

$$w^{n+1}_{i,j} = w^n_{i.j} - \frac{\Delta t}{\Delta x}(A_R F_{m,j+\frac{1}{2}} - A_L F_{m,j-\frac{1}{2}} + A_{avg}(F_{p,j+\frac{1}{2}} - F_{p,j-\frac{1}{2}})) - \Delta t\, q^n_i$$

Where F_m refer to the convective fluxes and F_p refer to the pressure fluxes. A_L, A_R and A_{avg} refer to the work flow area changes in a segment. The average work flow area changes are given as $A_{avg} = 0.5\,(A_L + A_R)$ [5].

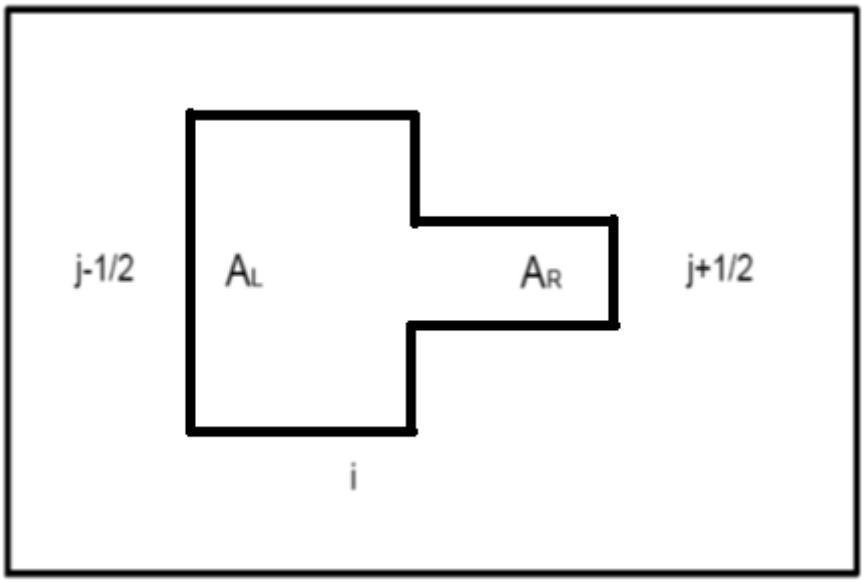


Figure 27: Flow area change [4]

The modified equation is used on all discrete box segments, and average phase volume and pressure is found after every new time step. These are found by the assistance of the mass conservative variables $w_{1,j} = A\alpha_l\rho_l$, $w_{2,j} = A\alpha_g\rho_g$ and dividing them by $\mathbf{A_{avg}}$. Lastly, the phase velocity is found by these equations: By combining the vale for $w_{3,j} = A(\alpha_l\rho_l v_l + \alpha_g\rho_g v_g)$ and the gas slip relation [1].

6.2 Boundary treatment

The model used in the simulations in this thesis, is of a hyperbolic nature. We have to be careful in defining the boundaries in the model. It is important to be precise in treating the information going in and out of the system. In this work simple extrapolation techniques have been used for the treatment of the boundaries. Without complex compatibility relations in combination with the physical boundary conditions, there is a possibility of problems when simulating as we will show in the next chapter [5].

At the inlet boundary, we know the mass flowrates of liquid and gas. So the convective flux term is given. The unknown is the inlet pressure flux, and this is found through extrapolation, by using information from neighboring segments. The boundary treatment in the node is given by [5]:

$P_{inlet} = p(1) + 0.5(p(1) - p(2))$

In an open outlet boundary, as the one we are dealing with in all the simulations, the outlet pressure is given. In these open end boundaries one has to extrapolate to specify the numerical fluxes for the convective terms [5].

The figure below show that pressure pulses typically will propagate back and forth in the well until they are dampened out by friction forces. Pulses are generated by e.g pump rate changes. These plots were generates with a numerical scheme that used compatibility relations for solving the boundary problems.

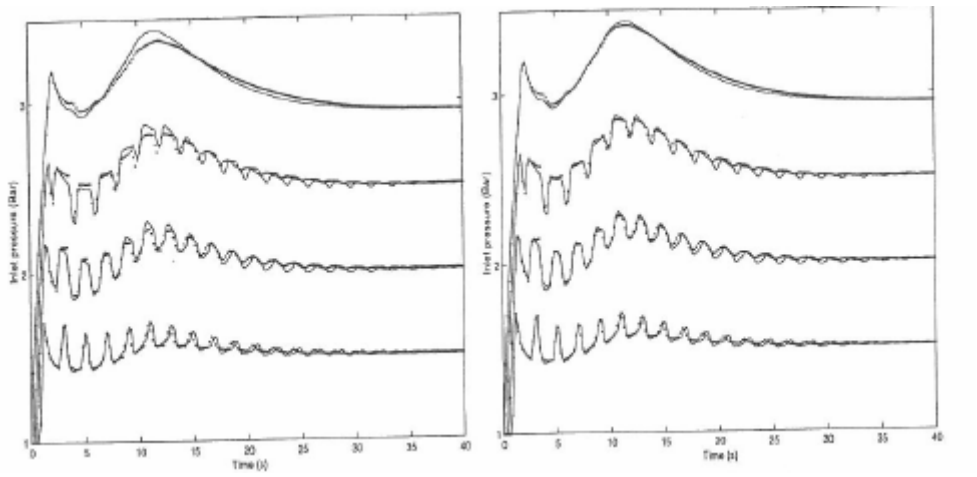


Figure 28: Inlet pressure vs time. Examples of pressure pulse propagating back and forth in the well [6].

7. Simulations

The AUSMV scheme is used in this section to simulate different scenarios concerning propagation of pressure pulses in a well. When starting up pumps in a well, pressure pulses are generated and they will propagate in the well with a speed equal to the sonic velocity. The same is true when we adjust choke valves at the outlet of the flowpath. This means that the effect of for instance a choke adjustment in an MPD operation will take time before its impact will be felt in other parts of the well. The choke pressure adjustment signal need time to propagate in the system.

In an MPD backpressure system, it is common to try to control the pressure in the well at a certain location. This could be at the place where there is small margin between the pore and the fracture pressure. By measuring the pressure in the well, proper choke adjustments are then carried out to compensate if the measured pressure deviated from the pressure that is wanted at the location. In this context, one should be aware that it will take time before the choke pressure adjustment is felt at the sensor measuring the pressure. There will be no point in making a new choke pressure adjustment before the previous adjustment is "felt" at the sensor and gives advice on what the next step should be. There is a certain lag time involved caused by the sonic wave propagation velocity.

Hence in this chapter, we want to study the behavior of pressure pulse propagation to get more insight into this and we also want to study what impacts the propagation velocity of the pulses. It is also demonstrated that the AUSMV scheme in combination with the fully transient drift flux model is able to simulate the propagation of pressure pulses.

Here, pressure pulse propagation caused by pump start up and choke valve adjustments will be studied. It will be demonstrated what effect friction has on the pulses and if there are differences between having a one-phase and two-phase flow system. In the latter case, we will also study how the gas volume content in the well affects the propagation velocities.

In order to display the pressure pulses (and not let them be subdued by the hydrostatic pressure in a vertical well), we have considered a horizontal well path configuration at surface conditions (1 bar). The well data used in the simulations are given as:

Well length: 5000m

Density of liquid: 1000kg/m^3

Initial pressure of horizontal well: 1 bar

Speed of sound: 1500m/s

Viscosity of liquid: 0,05Pa*s

Viscosity of gas: 0,000005Pa*s

Hole diameter: 0,2159m

Drillpipe diameter: 0,127m

First, pressure pulses generated by increasing the inlet rates are considered. The pump startup is performed in a short time interval i.e. 0.1 sec. In practice, this pump startup time is very extreme, a more normal pump start up time would be 20-30 seconds. However, we have induced this rapid startup time to exaggerate the pressure pulse propagation effect and visualize them better. Another reason was that it was experienced during simulations that the current version of the AUSMV scheme is not able to handle reflection of pressure pulses at the well boundaries. The numerical boundary treatment needs to be revised to include this.

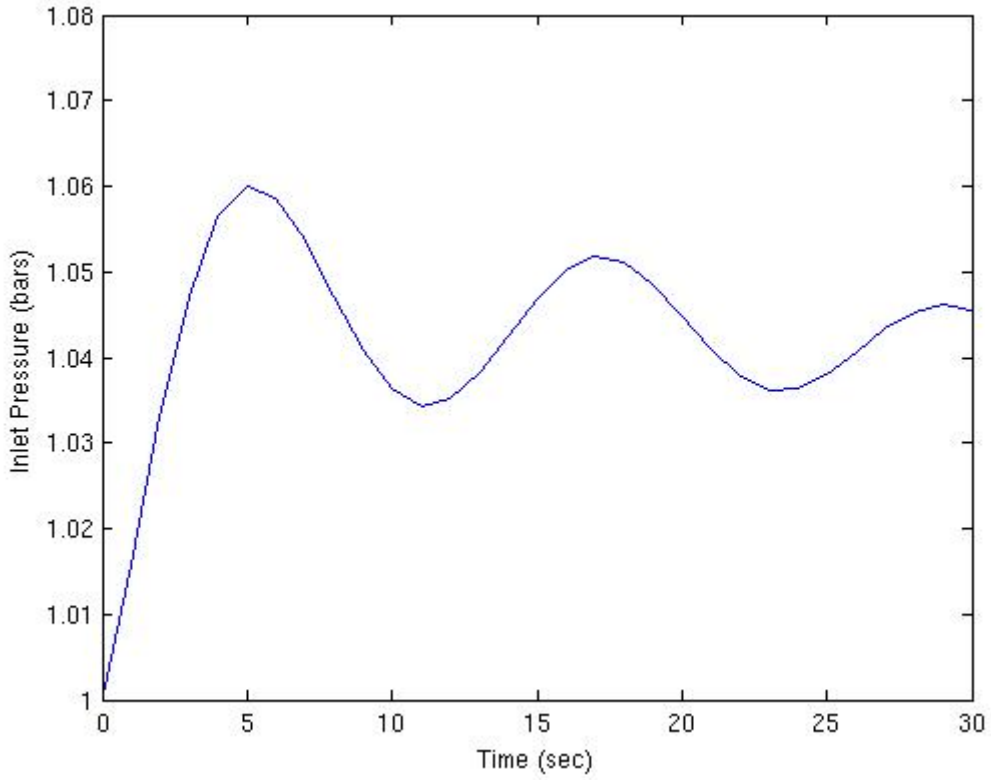
7.1 Simulation: Boundary treatment

In this section we try to simulate pressure pulses and see how the AUSMV scheme behaves with the boundary treatment proposed in the previous chapter (extrapolation techniques). All the simulations under this heading (boundary treatment) have the same startup time for the pump i.e. 0,1 sec. The pressure pulse is generated by increasing the inlet liquid rate from 0 – 25 kg/s in 0,1 sec. The plots in this chapter have captured the situation after 30s. As we will look at different scenarios in the well, we can see that the AUSMV scheme used, needs some tuning of the boundary treatment.

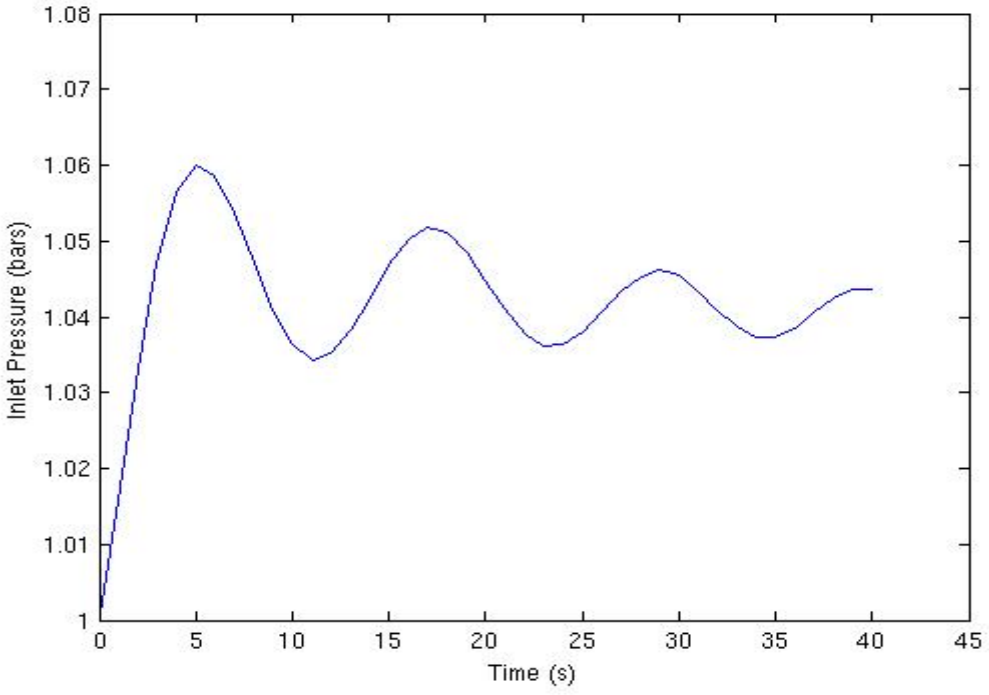
7.1.1 Boundary treatment: 50% Gas and 50% Liquid, Without friction: startup time = 0,1 sec, inlet mass rate = 25 kg/sec:

In this first scenario, we are simulating in horizontal well with 50% gas and 50 % liquid. We let the pressure pulse propagate in the well for 30 and 40 seconds respectively. The plots are simulated without friction included.

Here there can look like there is some kind of reflection of pulses taking place in the system. However the pulse should not be expected to dampened since there is no friction. Taking into account the speed of the sound in this fluid mixture (25 m/s), one should not expect that any reflection should have had time to take place. It is therefore quite difficult to know what the oscillations below expresses.



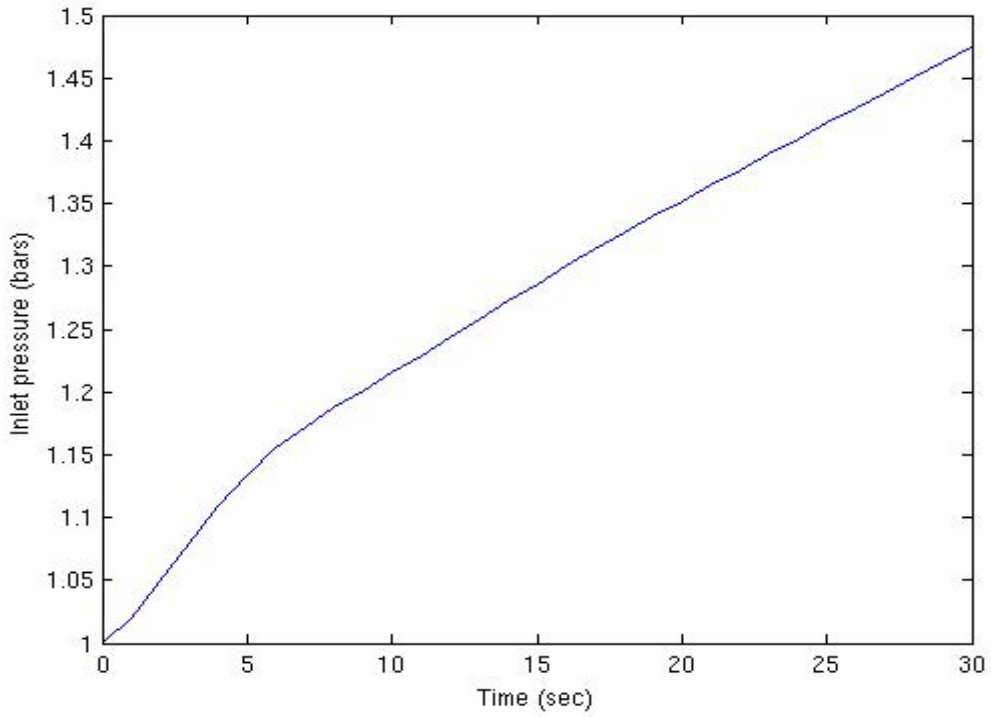
Plot 1.1: Boundary treatment: two-phase zone, without friction. Time = 30sec



Plot 1.2: Boundary treatment: two-phase zone, without friction. Time = 40sec

7.1.2 Boundary treatment: 50% Gas and 50% Liquid, With friction: startup time = 0,1 sec, inlet mass rate = 25 kg/sec:

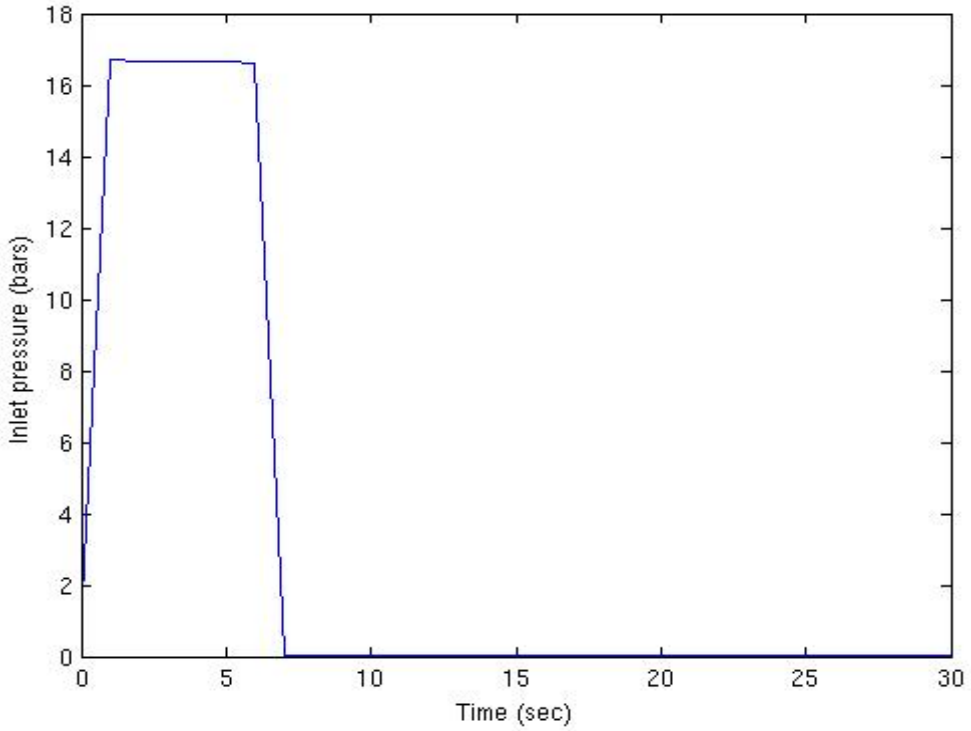
In this plot, we have added the friction into the equation. Analyzing how the AUSMV scheme handles this change. After 30 seconds, the pulse has only migrated 30x25m/s= 750 meter. The pulse has not had time to be reflected at any boundaries. Hence this plot could possibly be normal.



Plot 1.3: Boundary treatment: two-phase zone, With friction. Time = 30sec

7.1.3 Boundary treatment: 100% pure liquid, Without friction: startup time = 0,1 sec, inlet mass rate = 25 kg/sec:

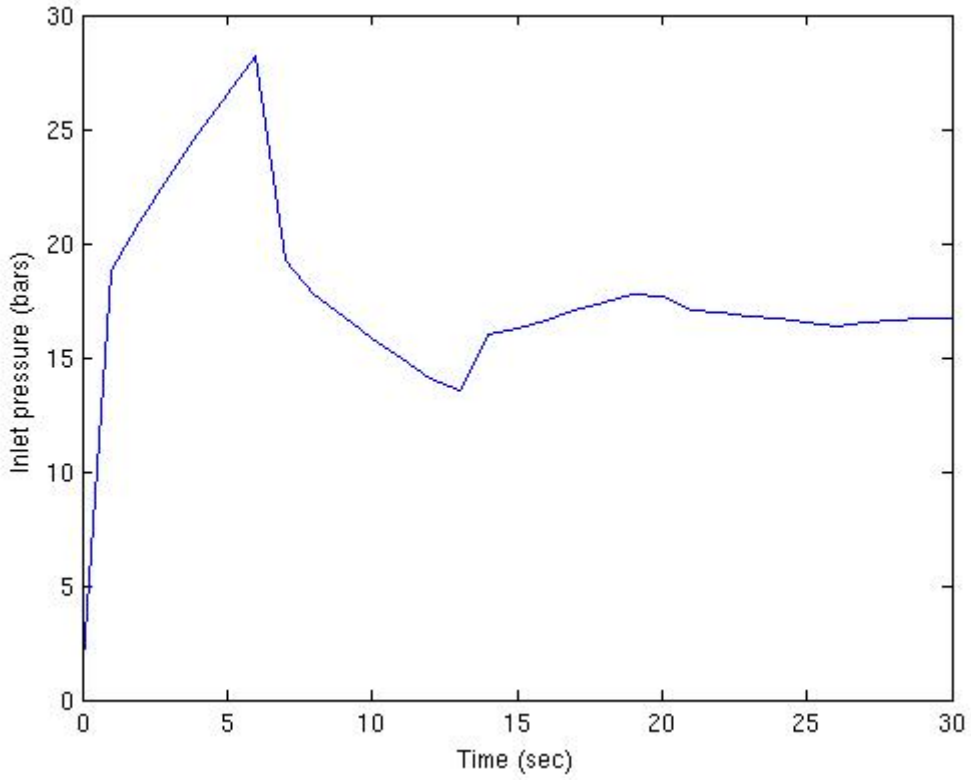
We now change the scenario, and look at a 5000m horizontal well filled with 100% pure liquid. In a system without friction, one should expect that the pressure pulses would migrate back and forth undamped vs time. From the plot below, it does not seem that this happens in this case. The pulse is killed when it has been reflected back to the inlet.



Plot 1.4: Boundary treatment: 100% pure liquid, Without friction. Time = 30sec

7.1.4 Boundary treatment: 100% pure liquid, With friction: startup time = 0,1 sec, inlet mass rate = 25 kg/sec:

We consider a 5000m horizontal well filled with 100% liquid. The friction is included in this simulation. Here we see that there is some kind of reflection of the pulses taking place. However the dampening is not quite smooth as one should expect.



Plot 1.5: Boundary treatment: 100% pure liquid, With friction. Time = 30sec

In conclusion, the boundary treatment does not seem to work for a system where friction is not considered. The pulses are not transmitted un-attenuated back and forth in the system. For the one phase flow system, the pulse is also killed after one “travel” back and forth in the well.

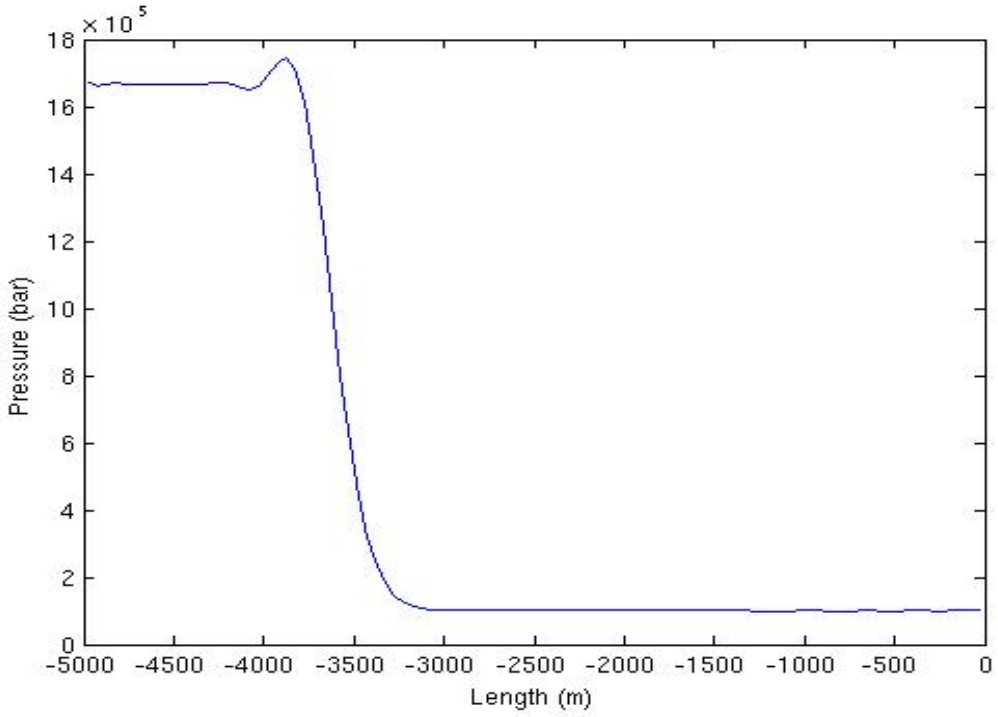
When we include the friction, the pressure developments seem to behave more in the sense we expect. However, one need to investigate further if this boundary treatment is good enough to handle pressure pulses properly and possibly one should use compatibility relations instead [29] to solve the boundary problem.

7.2 Simulation: Propagation of Pressure Pulses in 100% Pure Liquid.

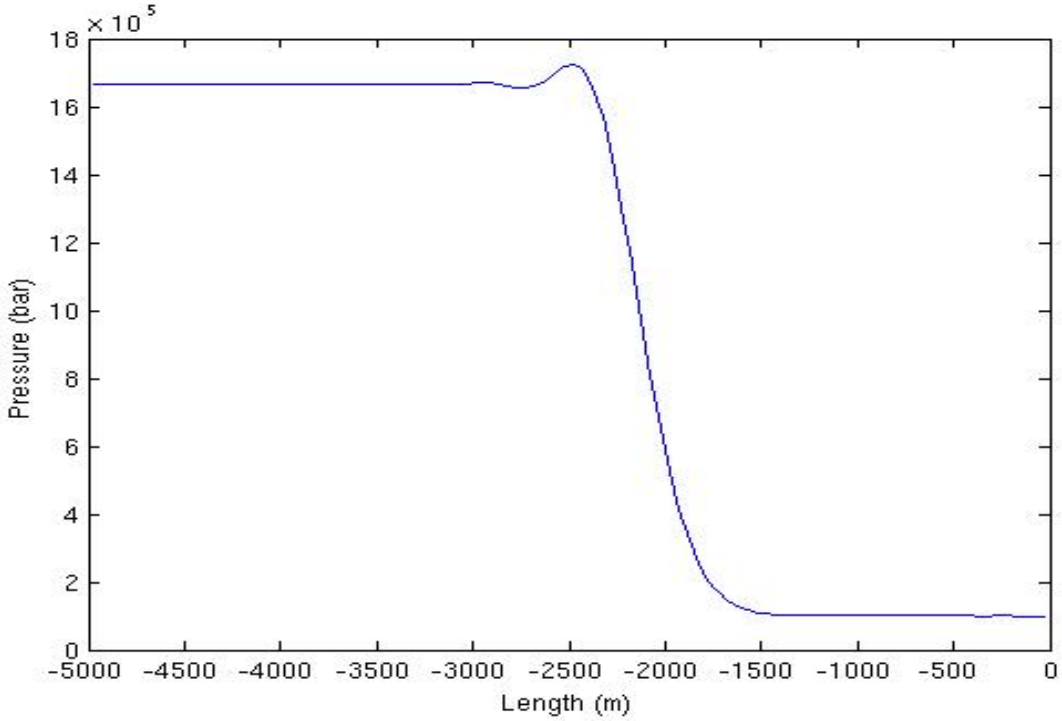
In this simulation we look at a 5000m horizontal well filled with 100% pure stagnant liquid. We then generate a pressure pulse by shock starting the pump. This is done by increasing the inlet liquid rate from 0 to 1500 lpm in a very short time span. This results in an extremely powerful pressure pulse (around 20 bars) as shown in the plots. The flowrate is ramped up to 1500 lpm which corresponds to 25 kg/s. **(1500 lpm** = 1, 5 m^3/min = 1, 5 m^3/60s = 0,025 m^3/s => 0,025 m^3/s * 1000 kg/m^3= **Q= 25 kg/s).**

7.2.1 Propagation of pressure pulses without friction in (100%) pure liquid

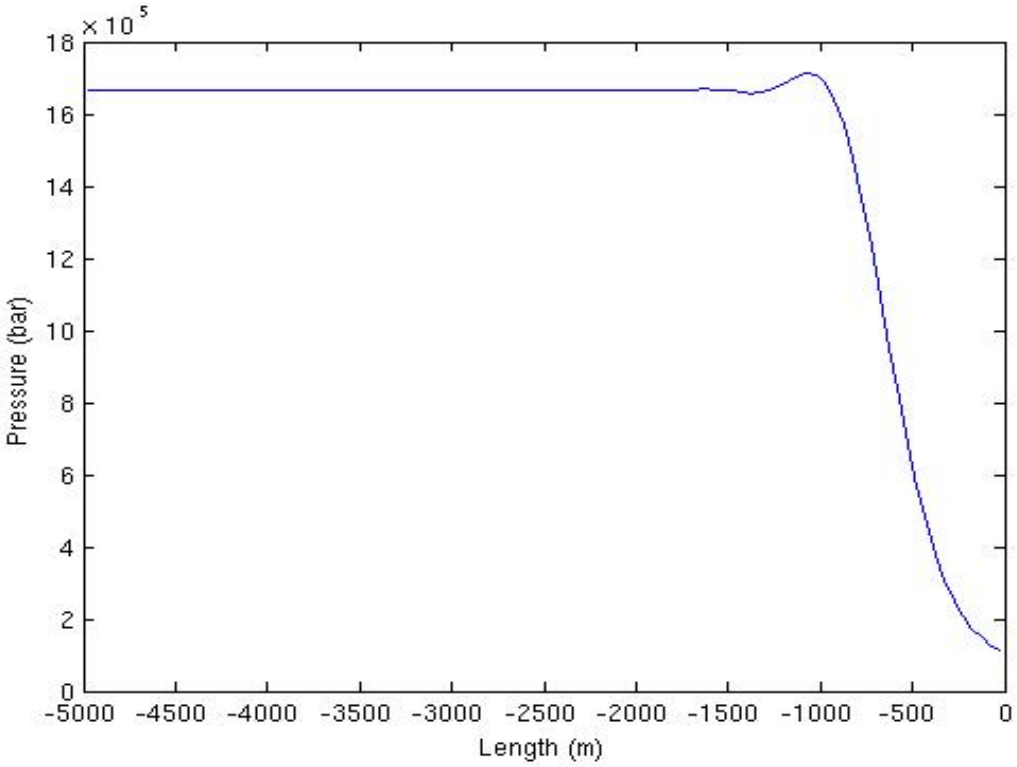
The plots 2 & 3 (1s, 2s, 3s) show two time series of a pressure pulse propagating through the well without friction. The two time series have two different startup times for the pump. In Plot 2.1 (1s, 2s, 3s), the pump has a startup time equal to 0,1 sec and in Plot 2.2 (1s, 2s, 3s), the pump has a startup time of 1 sec. The startup time is the time we increase the inlet liquid rate from 0 to 25 kg/s. The time series (1s, 2s, 3s) is a "picture" of the pulse moving through the well after 1 sec, 2 sec, 3 sec respectively. From the plots we can see that the two time series show no significant difference in amplitude of the pressure pulses. Since the friction is not considered in these simulations, we can see that the pressure pulse amplitude does not reduce or decrease as it moves through the well. After 3 seconds, we observe that the sonic wave has propagated 4500 meter which is in accordance with the sonic velocity used for defining the compressibility of the liquid in the AUMSV scheme (See Plot 2.3). When comparing Plot 2.3 and Plot 3.3, we also observe that the pulse generated by the 0.1 second startup time has propagated longer which is natural since it is induced more rapidly.



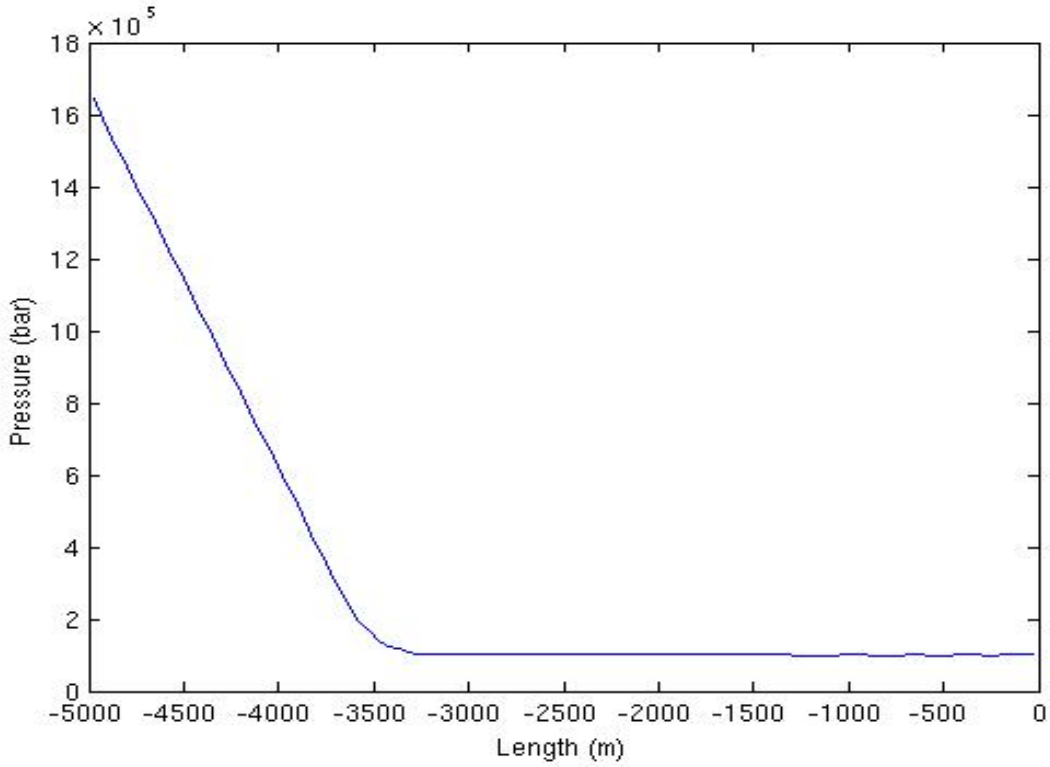
Plot 2.1: Startup time = 0,1 sec, and situation after time = 1 sec



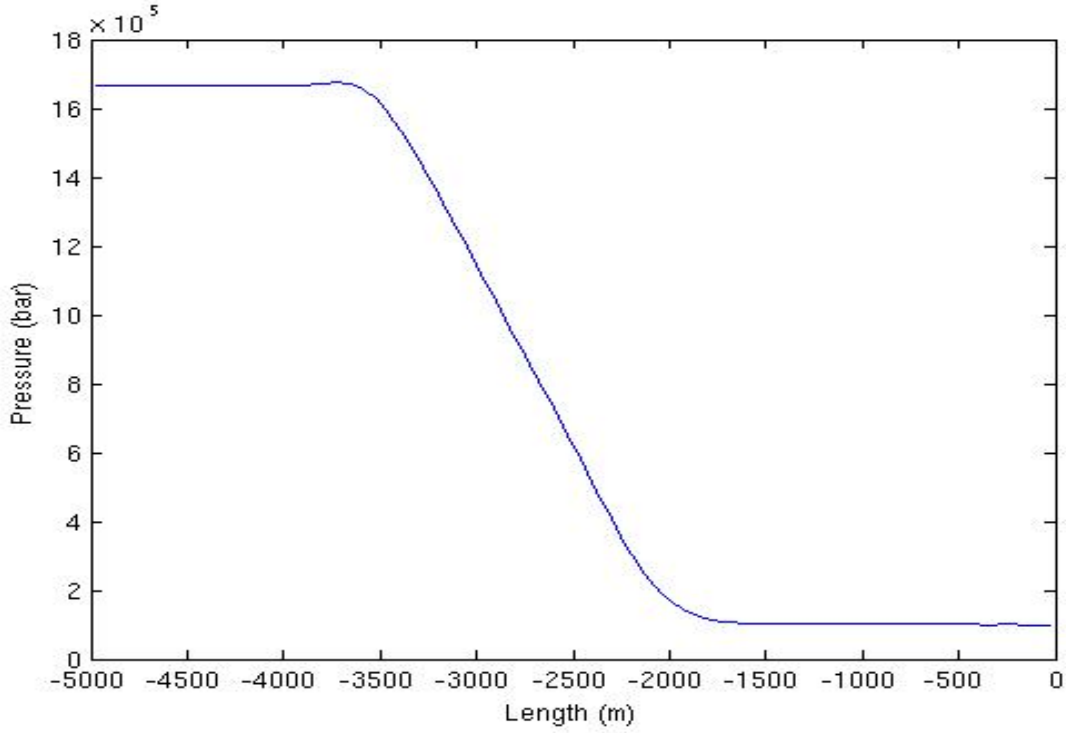
Plot 2.2: Startup time = 0,1 sec, and situation after time = 2 sec



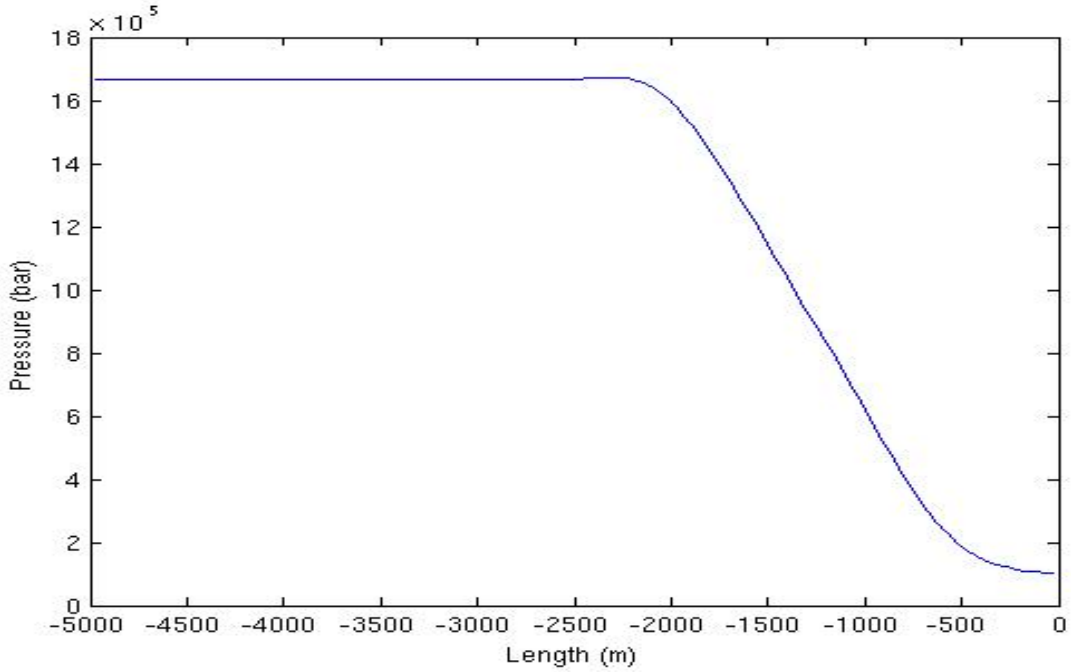
Plot 2.3: Startup time = 0,1 sec, and situation after time = 3 sec



Plot 3.1: Startup time = 1 sec, situation after time = 1 sec



Plot 3.2: Startup time = 1 sec, Situation after time = 2 sec



Plot 3.3: Startup time = 1 sec, situation after time = 3 sec

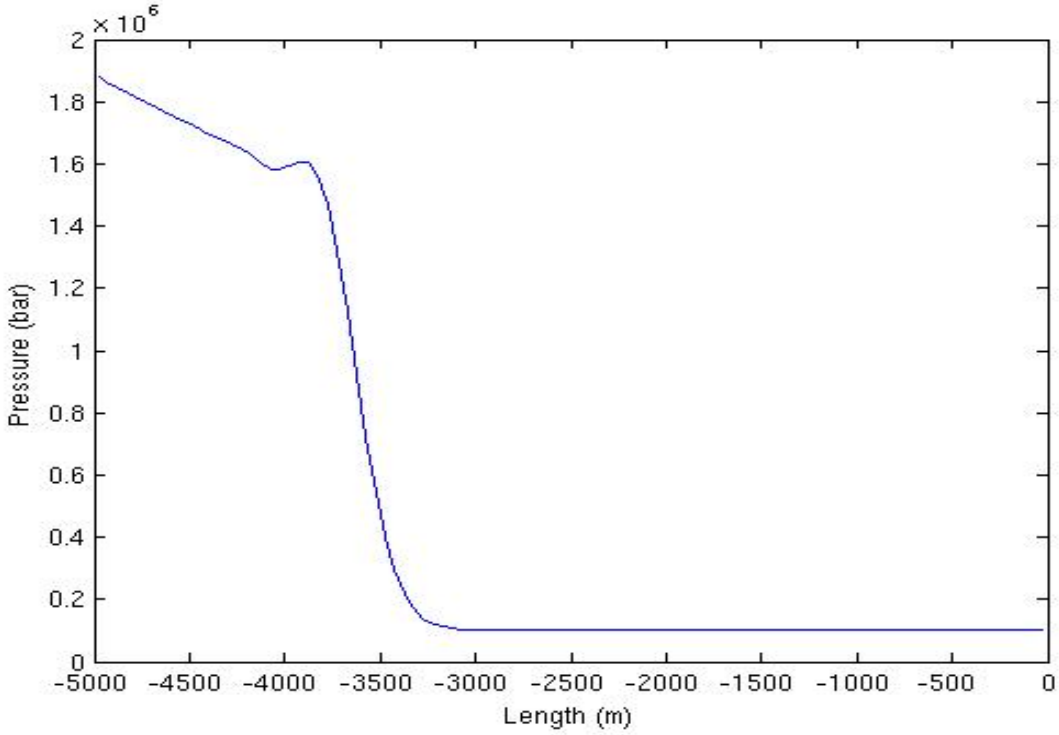
7.2.2 Propagation of pressure pulses with friction in (100%) pure liquid

In the following, the effect of including well friction in the model will be studied.

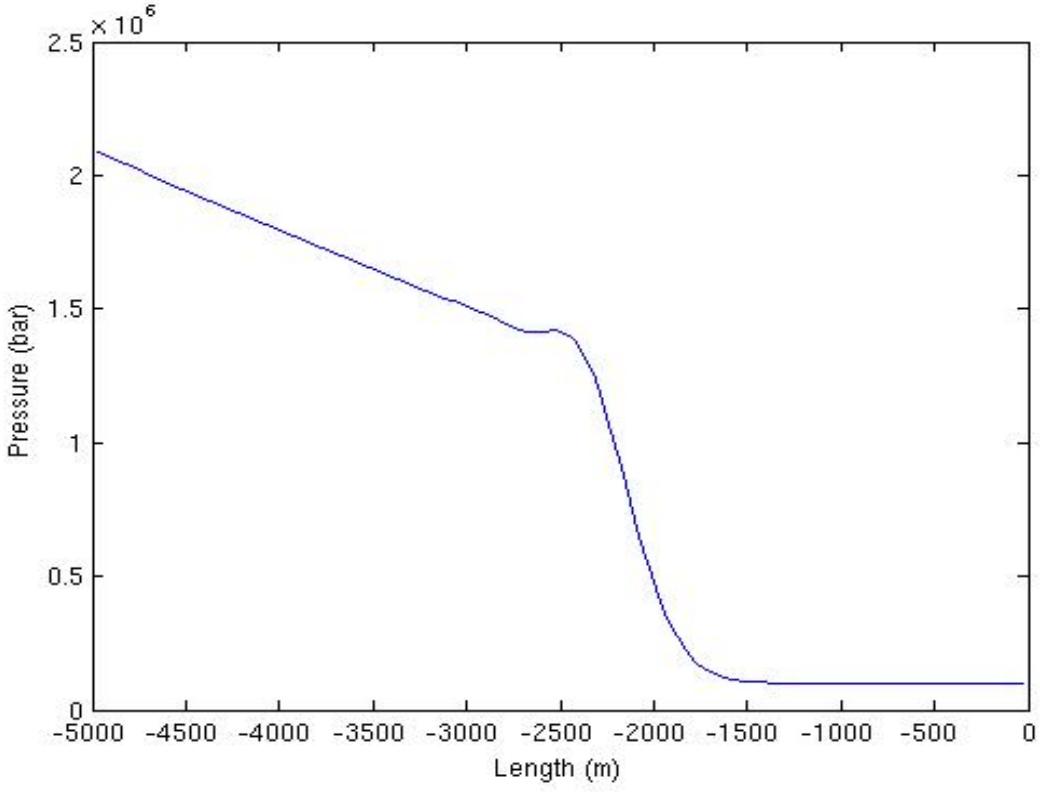
The plots 4 & 5 (1s, 2s, 3s) show two time series of a pressure pulse propagating through the well **with** friction. The two time series have two different startup times for the simulation. In Plot 4 (1s, 2s, 3s) the pump has a startup time equal to 0,1 sec and in Plot 5 (1s, 2s, 3s) the pump has a startup of 1 sec. If we consider Plot 4.1, the first steep increase in pressure at –3500 meters corresponds to the pressure pulse (the pressure increase seen at the front). Here it is around 16 bars. The linear increase in pressure towards the inlet of the well represent the additional pressure caused by the friction itself due to fluid movement. In Plot 4.3, we see that the inlet pressure is around 23 bars while in the no friction case, it was around 17 bars. This difference is caused by the friction in the system and this additional pressure is what is required to move the fluid when friction working against the flow is present. If we compare Plot 4.1, 4.2 and 4.3 we also see that the inlet pressure increases vs time. The reason for this is the friction increases as the pulse propagates further down the well. This is related to that the pressure pulse also reflects a velocity change and more and more fluid in the well is set into motion.

With respect to the pressure pulse itself, if we consider Plot 4.3, we observe that the magnitude of the pressure pulse in front has decreased to 12-13 bars from 16-17 bars in Plot 4.1. This attenuation of the amplitude of the pressure pulse is also caused by friction since it will dampen the amplitude of pressure pulses.

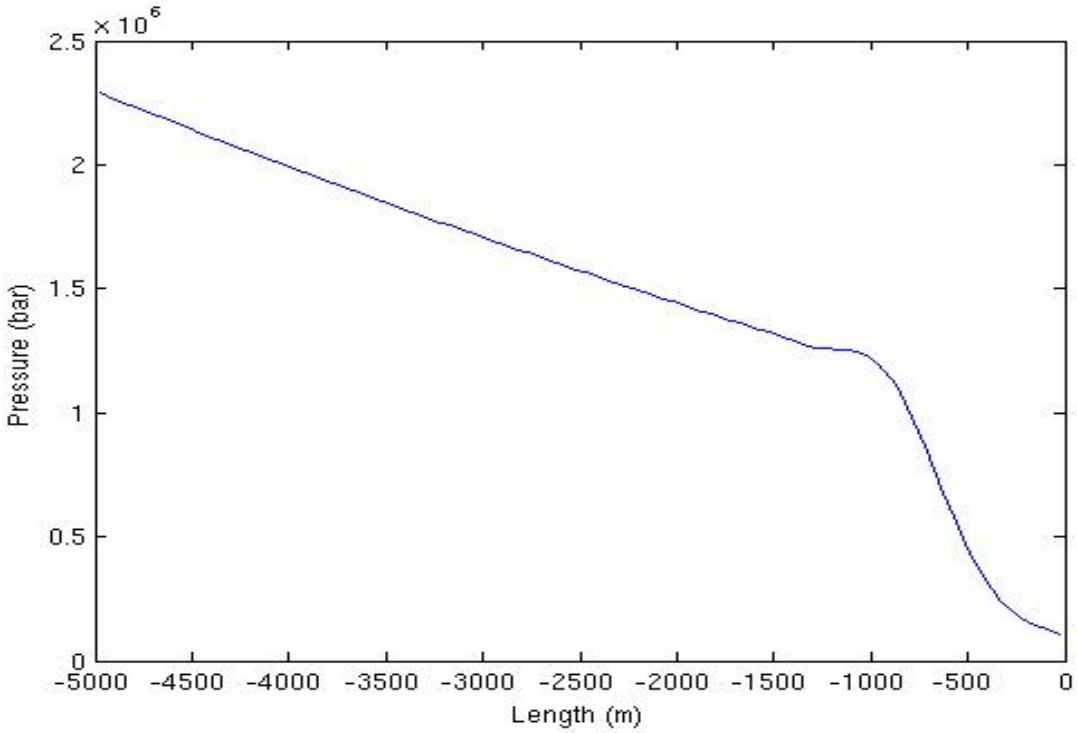
From the two simulation series using different start up times performed here, we could not observe that large difference in pressure pulse amplitudes.



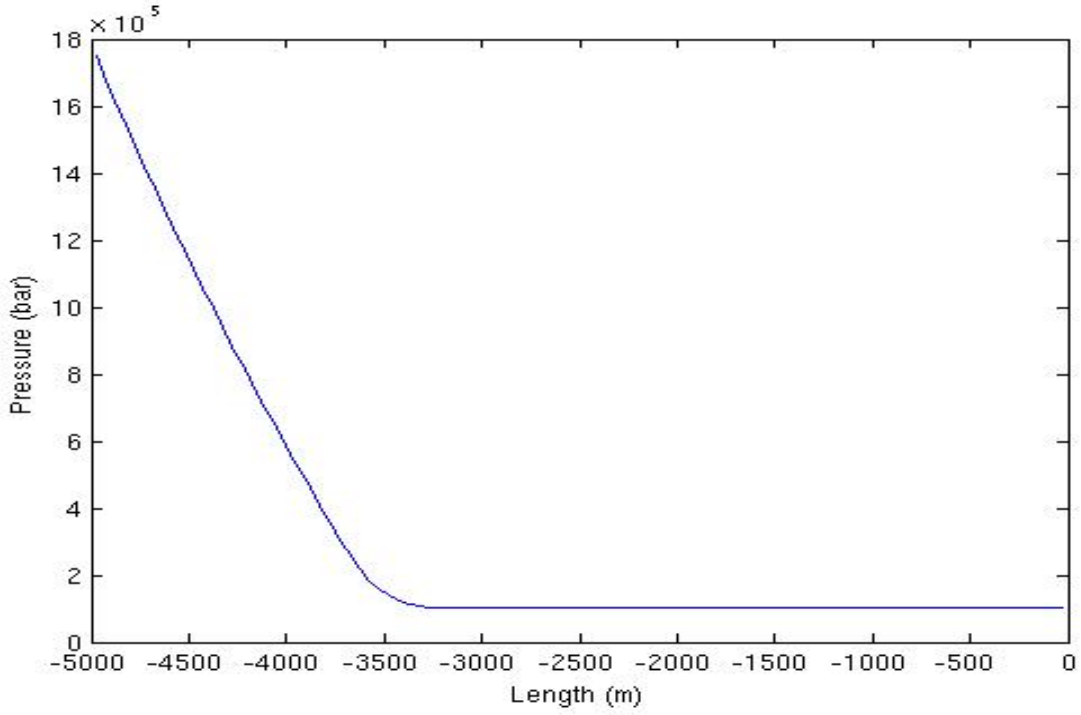
Plot 4.1: Startup time = 0,1 sec, after time = 1 sec:



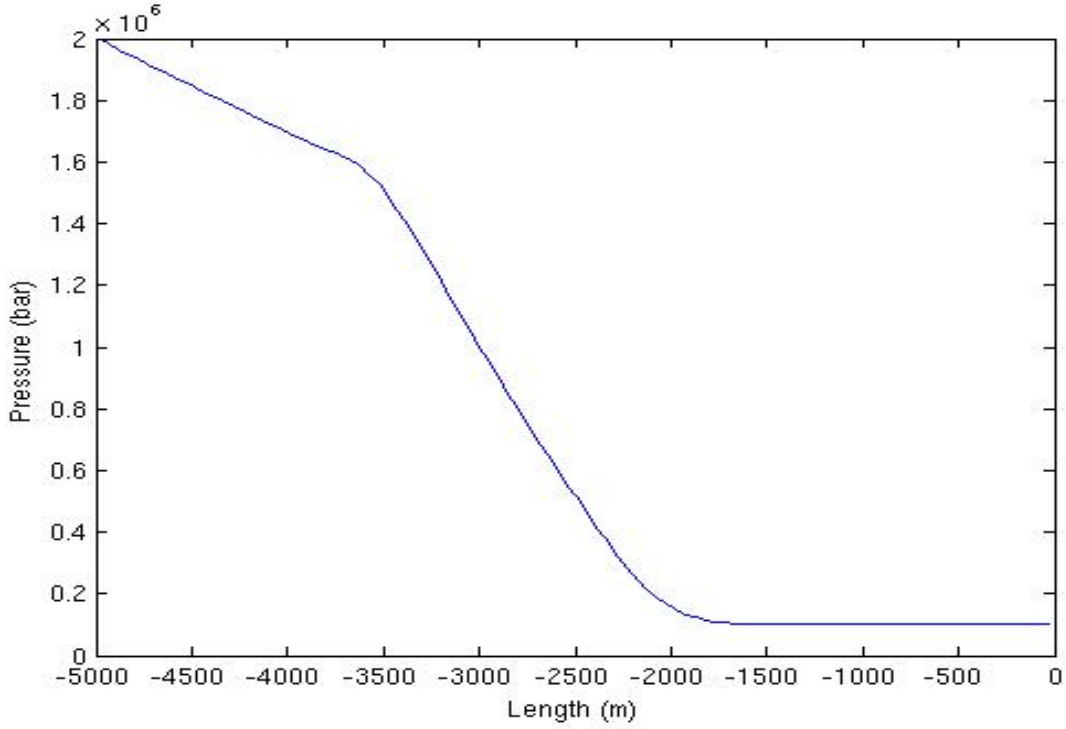
Plot 4.2: Startup time = 0,1 sec, after time = 2 sec



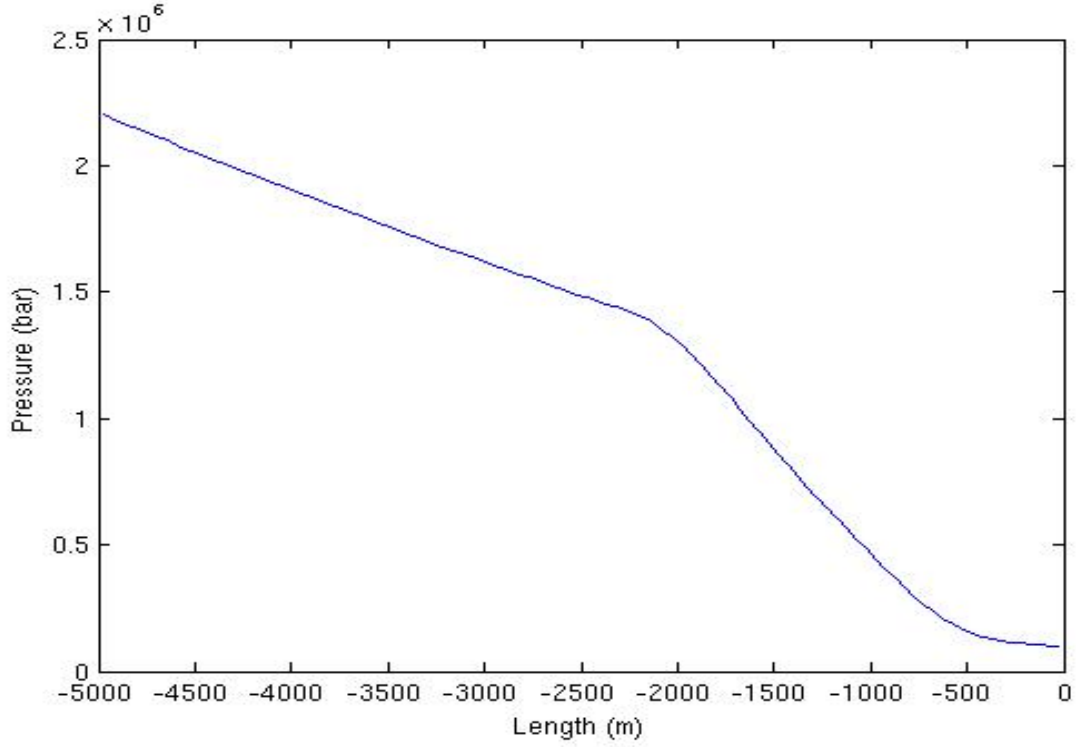
Plot 4.3: Startup time = 0,1 sec, after time = 3 sec



Plot 5.1: Startup time = 1 sec, after time = 1 sec



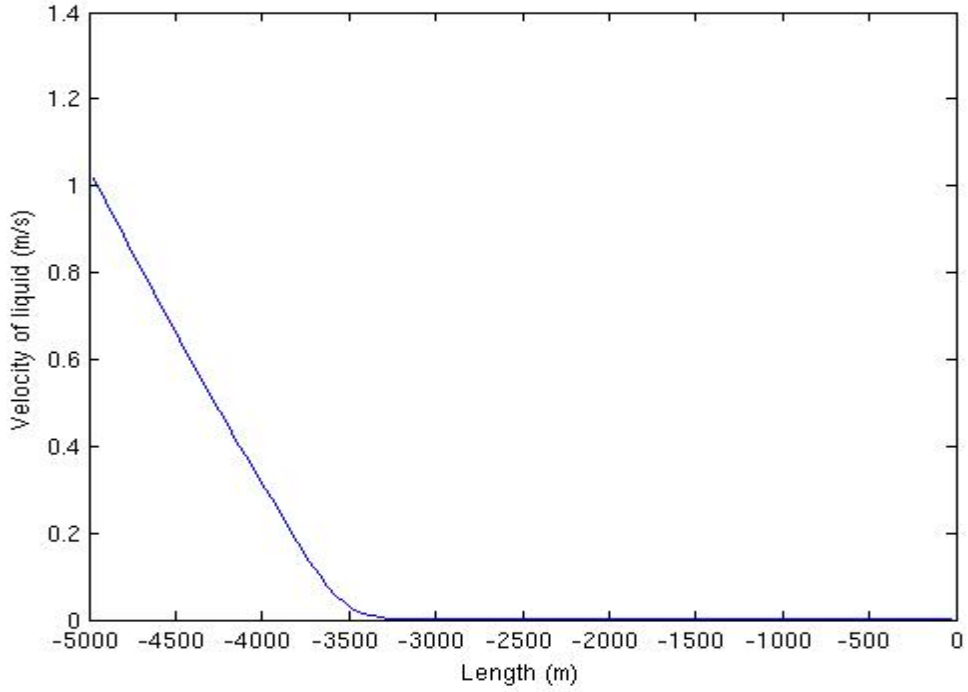
Plot 5.2: Startup time = 1 sec, after time = 2 sec



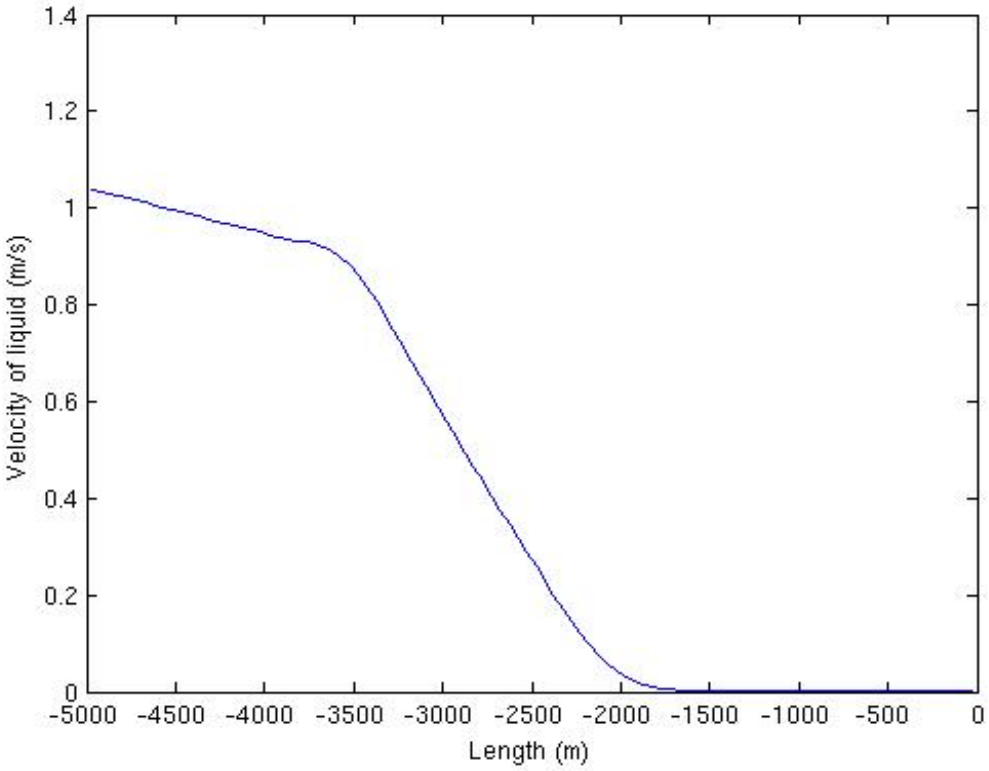
Plot 5.3: Startup time = 1 sec, after time = 3 sec

7.2.3 Velocity profile for pressure pulses in 100% liquid w/ friction:

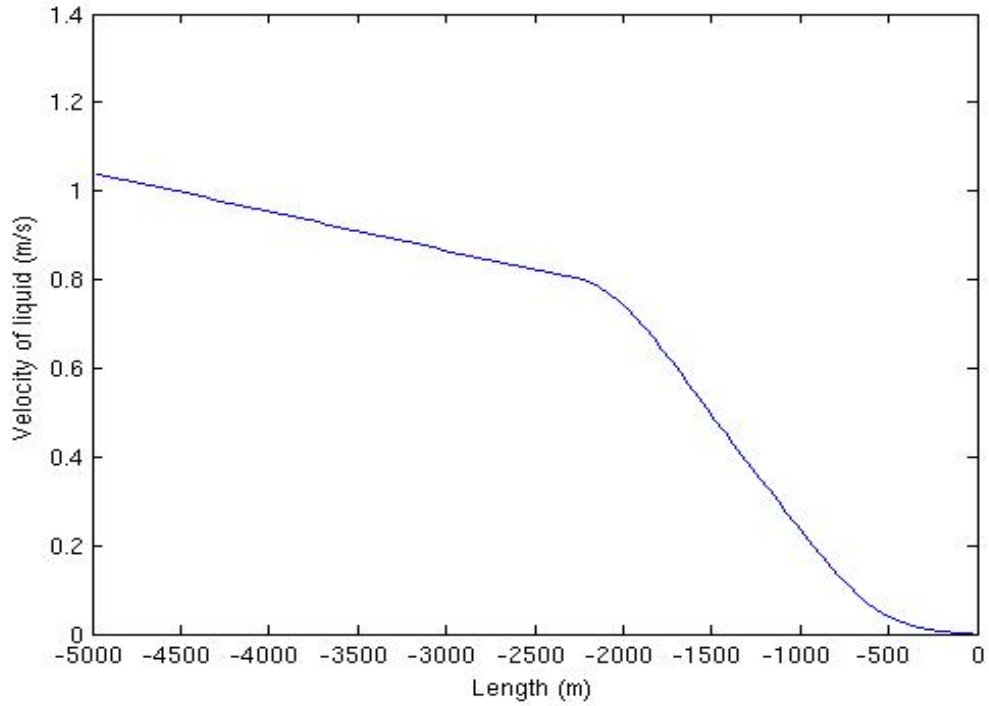
One should also be aware that the pressure pulse is associated with a corresponding change in fluid velocity as shown in the following three figures. Using the well data (inner and outer diameter of the well), we end up with a flow area of $\frac{\pi}{4}(0.2159^2 - 0.127^2) = 0.0239\,\text{m}^2$. The flowrate is 1500 lpm = 0.025 m^3/s. This gives an annular velocity of $v = \frac{Q}{A} = \frac{0.025}{0.0239} = 1.04$ m/s. This corresponds fairly well with what is seen in the plots below. This velocity is also what causes the additional pressure at the inlet of the well since it affects the frictional pressure loss term. As discussed before, the more fluid that is set in motion, the larger the inlet pressure will become.



Plot 6.1: Velocity profile, startup time = 1s, after time=1s



Plot 6.2: Velocity profile, startup time = 1s, after time = 2s



Plot 6.3: Velocity profile, startup time = 1s, after time = 3s

7.3 Simulation: Propagation of Pressure Pulses in Two-phase Region.

In this section, we will consider different pressure pulse propagation scenarios when considering presence of two phases. First we consider a 5000 m horizontal well with mixed density of gas and liquid. We generate a pressure pulse by "instantaneously" increasing the inlet liquid flow rate from 0 to 25 kg/s using a startup time = 0,1 sec. Friction is considered in all the simulations in this section. The two-phase scenarios simulated are:

1. Simulation where the well is filled with 99% liquid and 1% gas. The pressure pulse is then generated in the well. We simulate a time series for this scenario, with the time= 10s, 20s, 30s, respectively. Plot 7.1: time= 10s, plot 7.2: time=20s, plot 7.3: time= 30s.
2. Simulation where the well is filled with 90% liquid and 10% gas. The pressure pulse is then generated in the well. We simulate a time series in this scenario, with the time= 10s, 20s, 30s, respectively. Plot 8.1: time= 10s, plot 8.2: time=20s, plot 8.3: time= 30s.
3. Simulation where the well is filled with 50% liquid and 50% gas. The pressure pulse is then generated in the well. We simulate a time series in this scenario, with the time= 10s, 20s, 30s, respectively. Plot 9.1: time= 10s, plot 9.2: time=20s, plot 9.3: time= 30s.

By analyzing the plots, we observe the effect of the friction. The inlet pressure will increase as more and more fluid is set in motion. In addition, we observe that the amplitude of the pressure pulse itself is dampened vs time due to friction.

It is of especial interest to mention that for P = 1 bar, the speed of sound is 1500 m/s in pure liquid regions, but it drops dramatically in two-phase regions. In the region of 1% gas, the speed of sound is approximately 100 m/s. **Figure 29** and **Figure 30** for P =200. This can be an important fact to have in mind when considering MPD operations. If there are small amounts of gas in the MPD system then the pressure pulses will take longer to propagate and that might have to be taken into account when for instance regulating the choke. One has at least to be aware that there will be a longer time lag before we get feedback from the pressure sensors that are located in the well. In an MPD operation, we are operating quite close to the pore pressure and small kicks are easily induced. So the presence of small amounts of gas is likely in at least parts of the well.

An approximate expression for the sound velocity in the two phase flow mixture is given by [27], [28]:

$$w = \sqrt{\frac{p}{(\alpha_g \rho_l (1 - K\alpha_g)}}$$

If we assume 1 bar = 100000 Pa, the gas density will be 1 kg/m^3. The liquid density will be 1000 kg/m^3. If we assume no slip condition K = 1 and $\alpha_g = 0.01$. This gives w = 100 m/s.

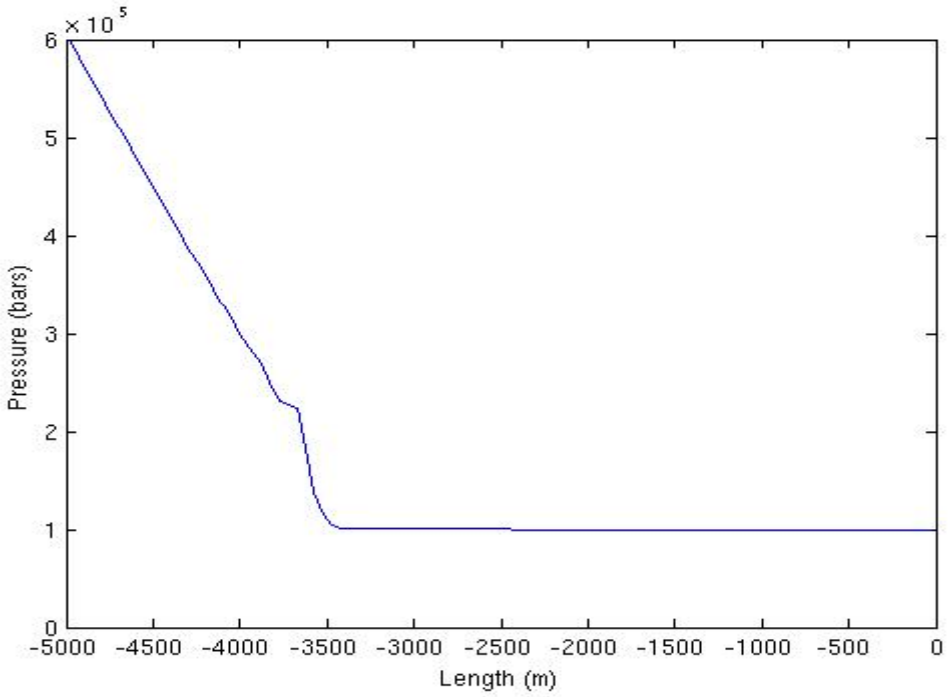
However, if we consider 200 bar = 200 00000 Pa, The liquid density will be 1008 kg/m^3 when using the formula: $\rho_l = 1000 + \frac{(p - p_0)}{1500^2}$ where p_o = 100000 Pa. This gives w = 1415 m/s. Hence for larger pressures the effect is not that profound. However if the gas content was 50 % at 200 bars, the sonic velocity would be 281 m/s. Hence, we observe that the sonic velocity is highly dependent on both the gas volume content but also the pressure conditions. The sonic velocity will e.g. be lower in the upper parts of the well. Here the pressure is lower, in addition the gas content can be larger here since small kicks will become larger in the upper part of the well due to gas expansion.

Table 1: Calculated sound velocities for two-phase flow mixtures, slip condition K=1:

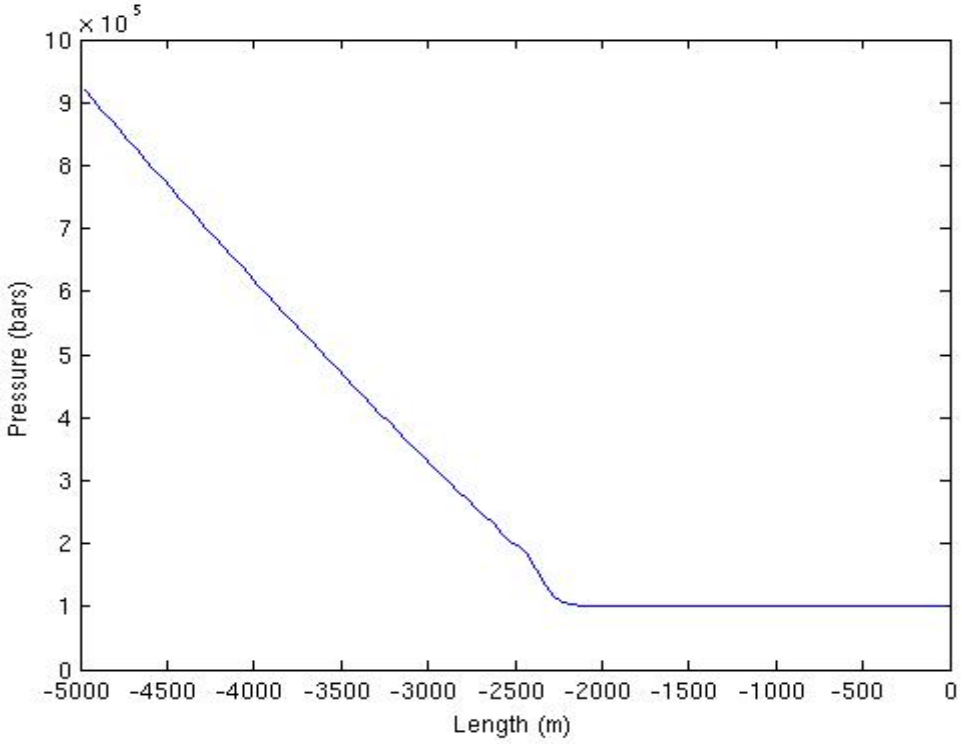
Gas Content/Pressure	1 bar	10 bar	100 bar	200 bar	500 bar
1 %	100 m/s	318 m/s	1003 m/s	1415 m/s	2223 m/s
10 %	33 m/s	105 m/s	332 m/s	469 m/s	737 m/s
50%	20 m/s	63 m/s	199 m/s	281 m/s	442 m/s

7.3.1. Propagation of pressure pulses with friction in 1% gas and 99% liquid:

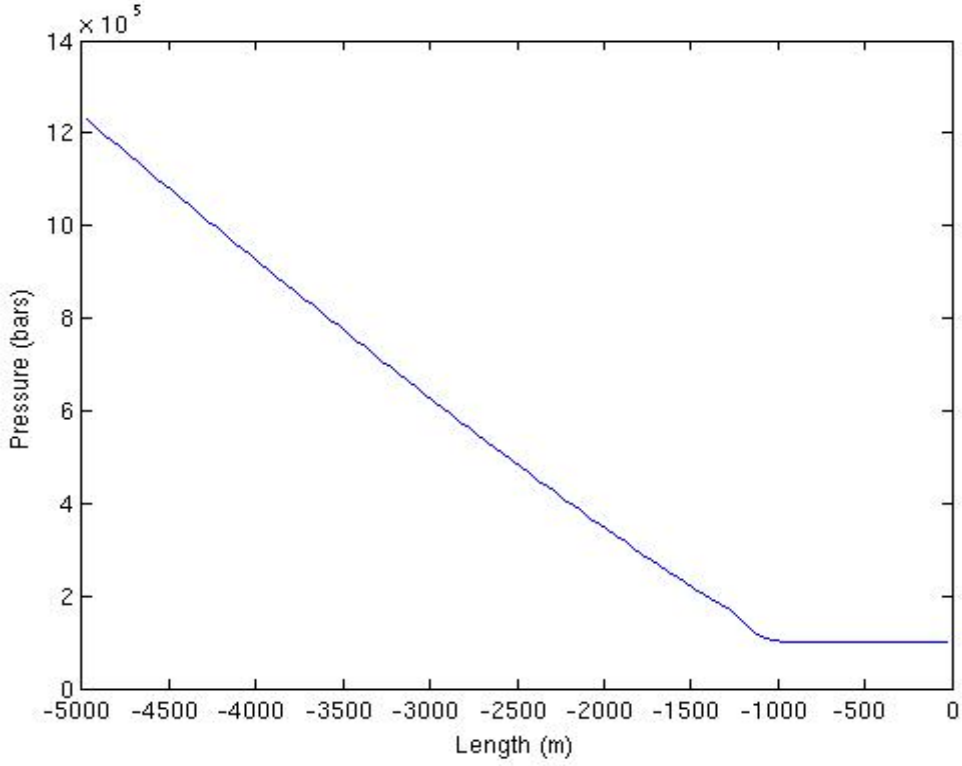
In this simulation, the well is filled with 99% liquid and 1% gas. The pressure pulse is then generated by increasing the inlet flow from 0 to 25 kg/s in 0,1s. The friction is included here. Pressure plots are then simulated in a time series, time= 10s, 20s, 30s, respectively. Plot 7.1: time= 10s, plot 7.2: time=20s, plot 7.3: time= 30s.



Plot 7.1: Pressure pulse in 1% gas and 99% liquid. Time= 10 sec



Plot 7.2: Pressure pulse in 1% gas and 99% liquid. Time= 20 sec



Plot 7.3: Pressure pulse in 1% gas and 99% liquid. Time= 30 sec

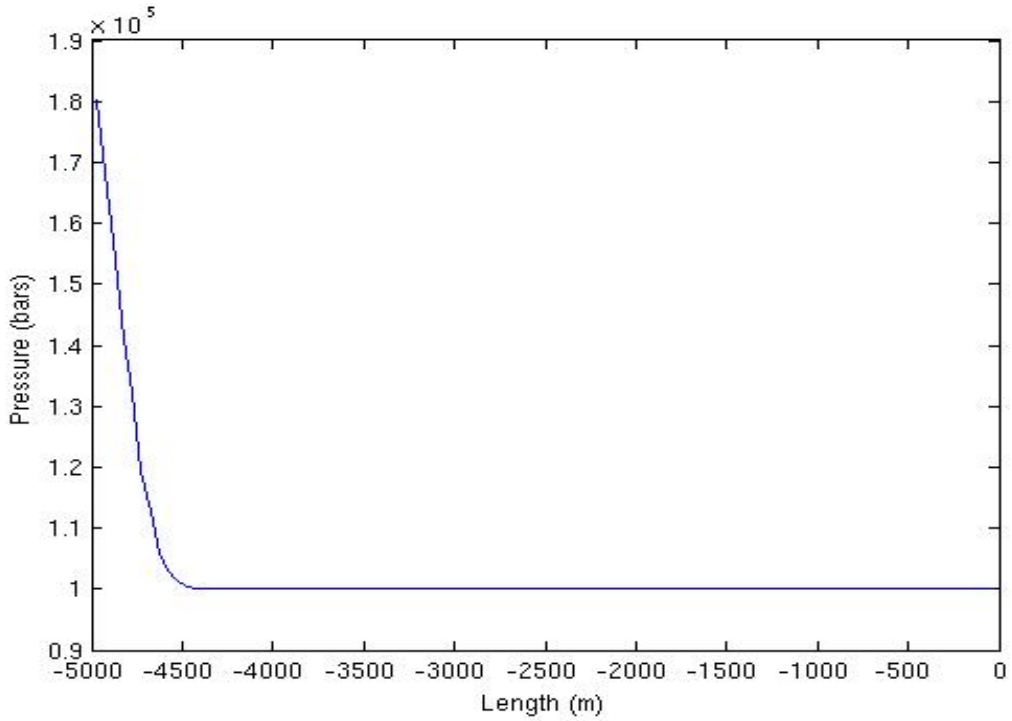
The pressure pulse needs much longer time to travel through this well than in pure liquid zones as seen in earlier sections. For the pure liquid case, we observed the pressure pulse propagating through the well using the time slices 1, 2 and 3 seconds. However, in the two phase flow case considered here (1 % gas), the time slices considered were 10, 20 and 30 seconds. This shows that the pressure pulse propagates much slower. If we consider Plot 7.3, an average propagation velocity can be obtained: 3700 m/30 s = 123 m/s.

We also observe how the friction attenuates the pressure pulse as it propagates in the well. After 10 seconds (Plot 7.1), the amplitude of the pressure pulse is around 1.3 bars (in front). However at time = 30 seconds it is reduced to slightly below 1 bar. In addition, we observe that the amplitude of the pulse itself (1-1.5 bar) is much smaller than for the one-phase flow case where it was around 16-17 bar. The presence of gas leads to a much smaller initial pressure pulse amplitude even though the startup time and the rate is the same as for the one-phase flow case. We also observe that the inlet pressure increase vs time also in this case but the magnitude is much smaller compared to the one-phase flow case (12 bar vs 23

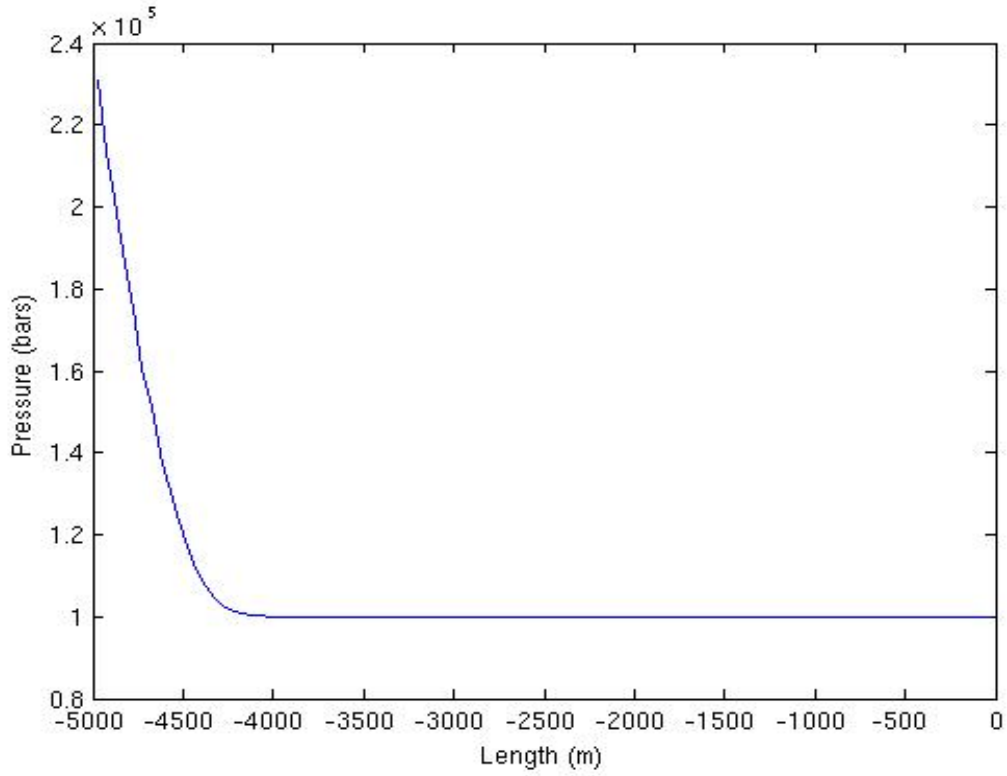
7.3.2. Propagation of pressure pulses with friction in 10% gas and 90% liquid:

The objective of this simulation is to show what will happen if the gas content is increased even further.

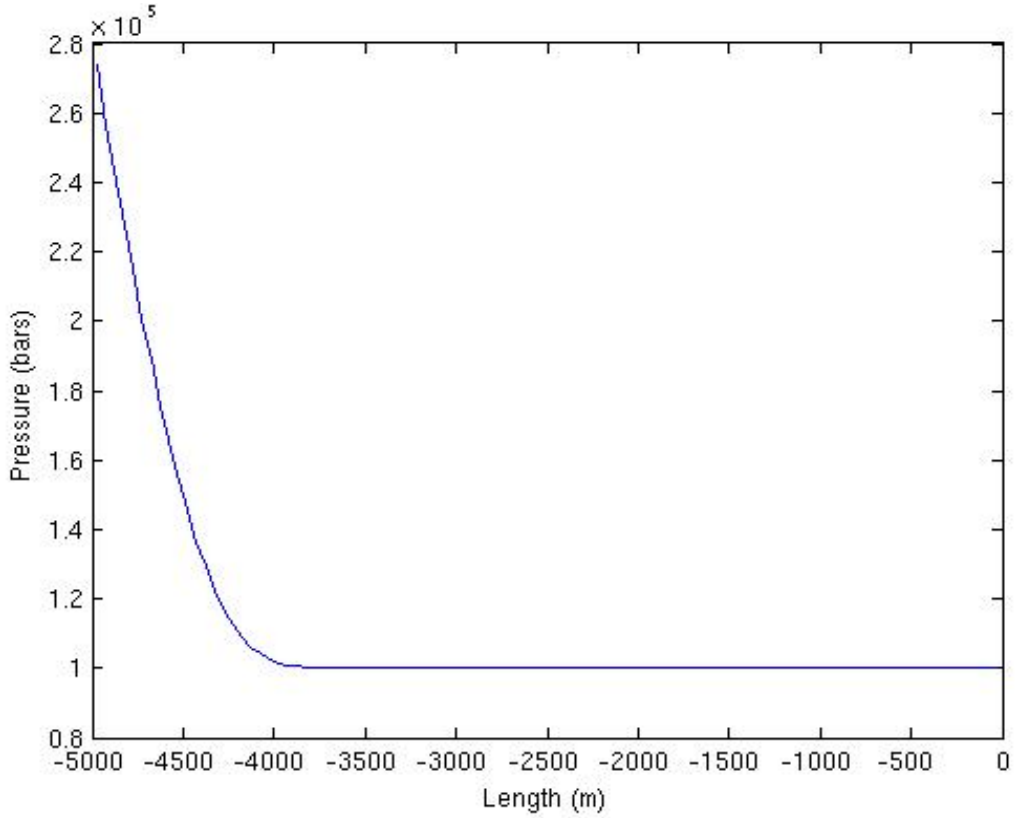
In this simulation the well is filled with 90% liquid and 10% gas. The pressure pulse is then generated by increasing the inlet flow from 0 to 25 kg/s using a startup time of 0,1s. The friction is included here. Pressure plots are then generated in a time series, time= 10s, 20s, 30s, respectively. Plot 8.1: time= 10s, plot 8.2: time=20s, plot 8.3: time= 30s.



Plot 8.1: Pressure pulse in 10% gas and 90% liquid. Time= 10 sec



Plot 8.2: 10% gas and 90% liquid. Time= 20 sec



Plot 8.3: 10% gas and 90% liquid. Time= 30 sec

It is evident from the plots that an increase in gas content will lead to even more reduction in the propagation velocity of the pressure pulses. There is a significant velocity change when we only have 1% gas compared to 10 % gas in the well. In addition, the sonic velocity of the mixture will depend heavily on the pressure conditions in the well. Below, we have plotted the expression for the approximate sound velocity for 1 bar and 200 bars. It can be noted that it is an approximate expression since the sound velocity for large gas volumes approaching one considering 200 bars may seem a bit unrealistic. However we observe that we have a minimum in the mixture sound velocity when the gas content is 50 %. We also observe that the mixture sound velocity increase from 25 m/s and approaches 200-300 m/s for larger pressures.

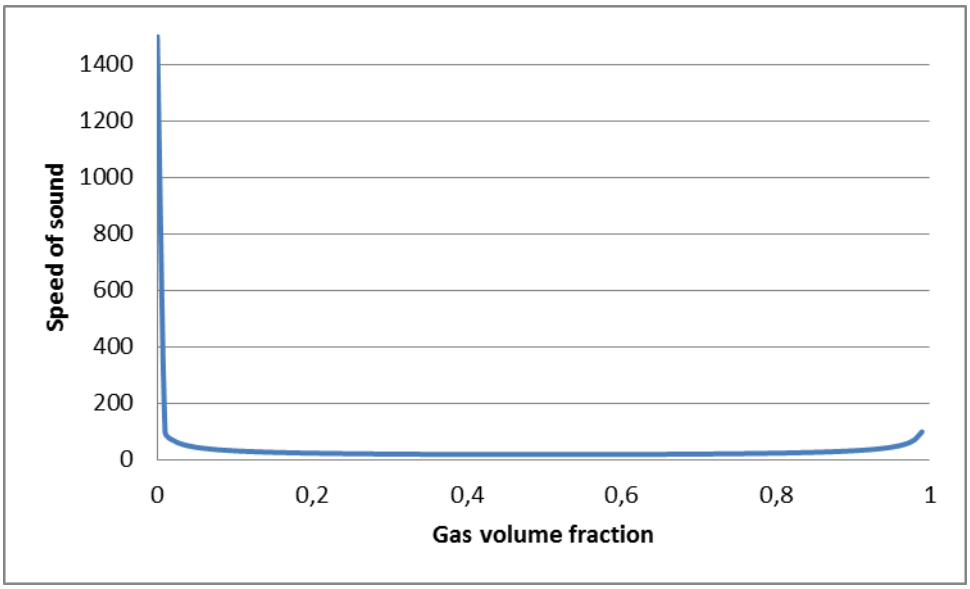


Figure 29: Sound velocities for P = 1 bar

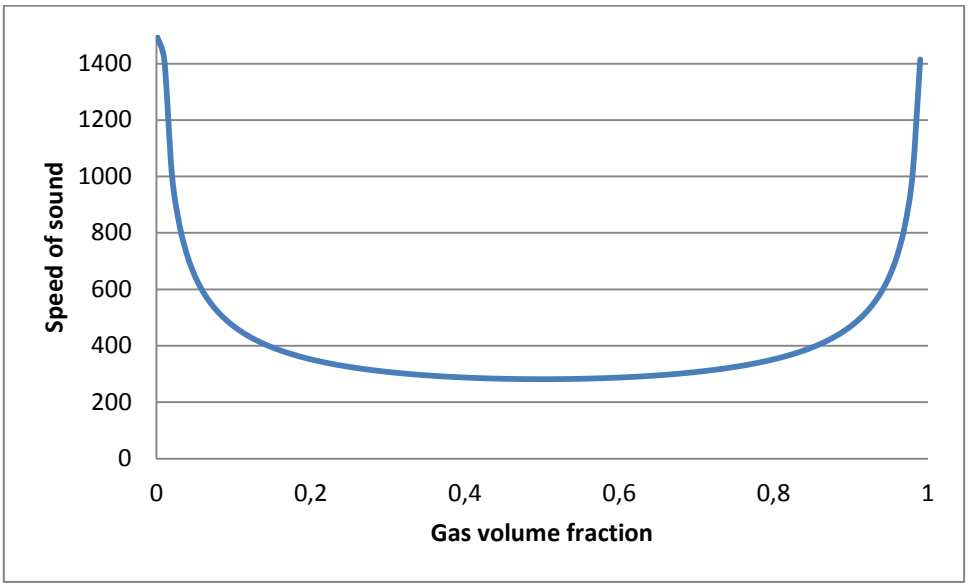
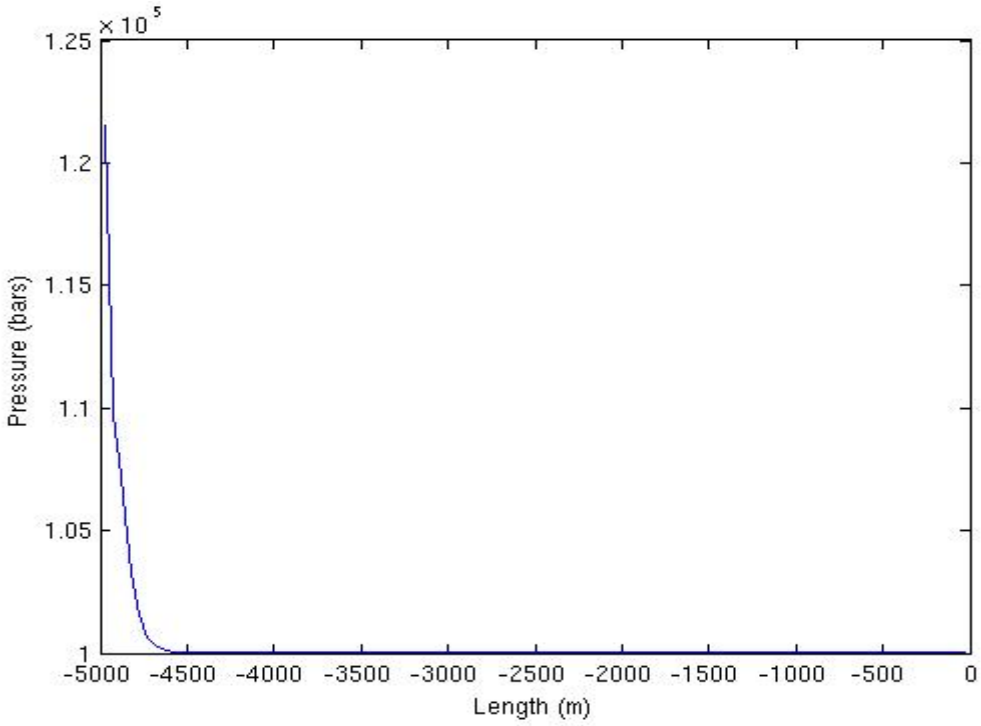


Figure 30: Sound velocities for P = 200 bar

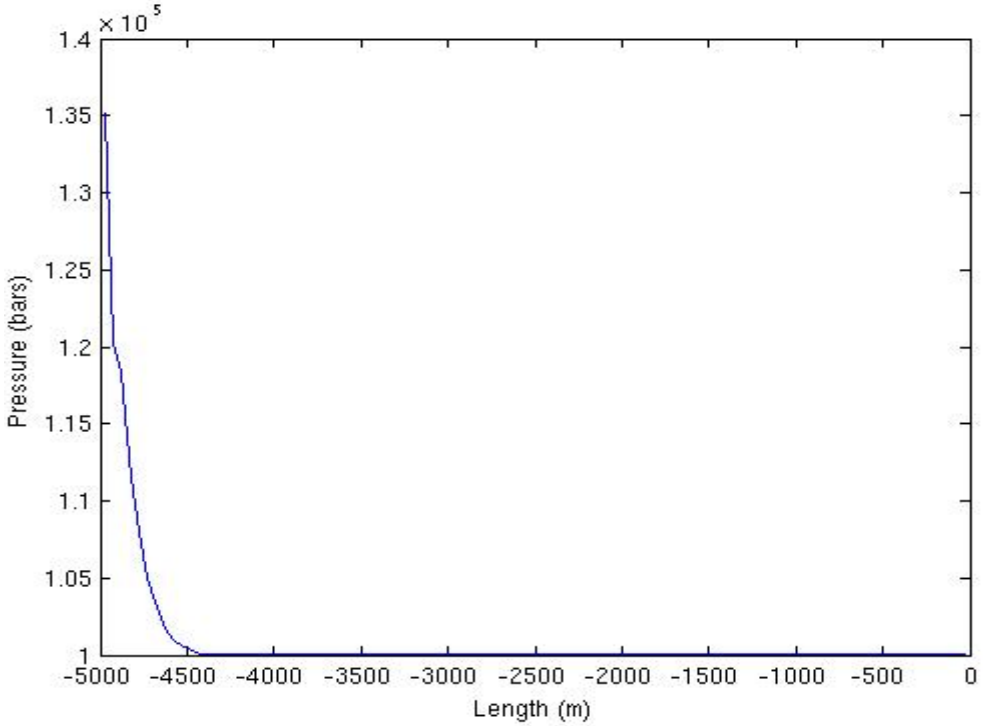
7.3.3. Propagation of pressure pulses with friction in 50% gas and 50% liquid:

In this simulation the well is filled with 50% liquid and 50% gas. The pressure pulse is then generated by increasing the inlet flow from 0 to 25 kg/s during a startup time of 0,1s. Friction is included. Pressure plots are then simulated in a time series, time= 10s, 20s, 30s, respectively. Plot 9.1: time= 10s, plot 9.2: time=20s, plot 9.3: time= 30s.

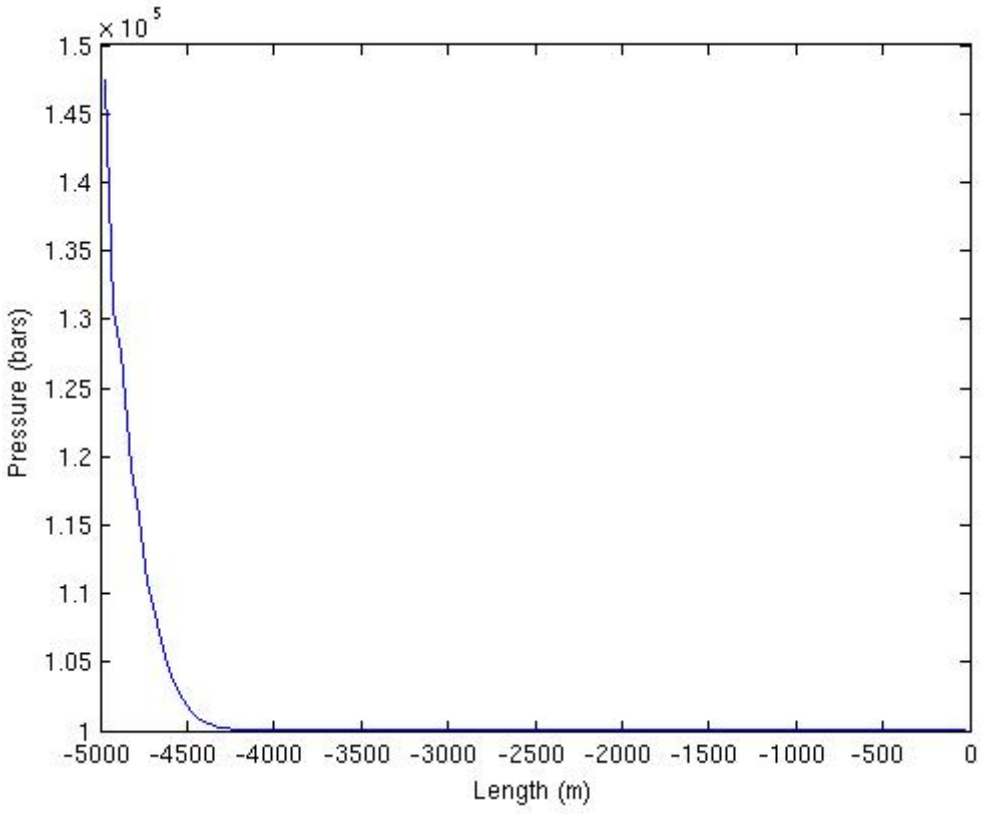
From the following three plots we can see that the pressure pulse needs a lot more time to move through this two phase flow mixture. As discussed previously, the sonic velocity is lowest when we have 50 % gas. Looking at the time series of plots 7, 8 and 9, we can conclude that the more gas in the well, the slower the pressure pulse travels. This is maybe something we need to have in mind when performing automatic choke regulation of two-phase flow mixtures. The simulations in this section show that an increased fraction of gas in the well will cause the speed of sound to decrease. In the upper part of the well with high gas contents and low pressures, the sonic velocity will be at its lowest.



Plot 9.1: 50% gas and 50% liquid. Time= 10 sec



Plot 9.2: 50% gas and 50% liquid. Time= 20 sec

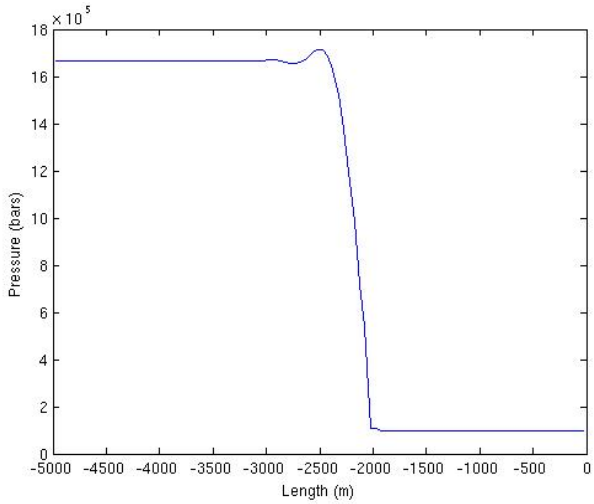


Plot 9.3: 50% gas and 50% liquid. Time= 30 sec

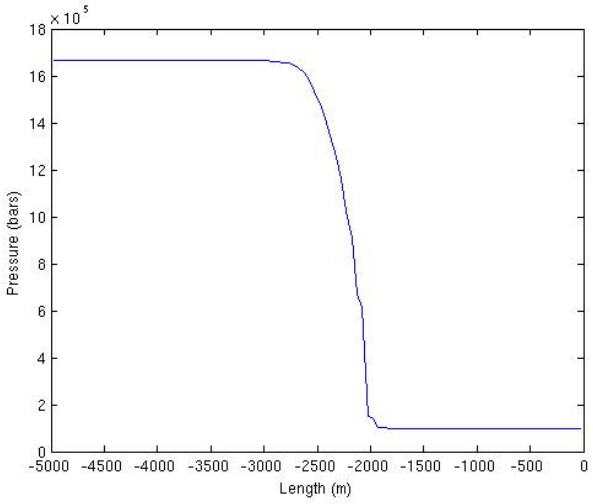
7.3.4 Propagation of pressure pulses in a well where 60% is liquid and 40% is two- phase.

In this simulation, we want to demonstrate the reflection of pressure pulses. We consider a 5000 m long horizontal well. The first 3000m are filled with 100% pure liquid while the last 2000m are filled with 1% gas and 99% liquid. We combine in this simulation both scenarios simulated in the earlier sections. The same procedure is followed in this simulation as the others in this chapter. We increase the inlet liquid flow from 0 to 25 kg/s in 0,1 sec to generate a pressure pulse. The pressure pulse is simulated in a time series: time = 2s, 2.2s, 2.4s, 2.6s, 2.8s, 3s, 3.2s, 3.4s and 3.6s. The time series show the pressure pulse which moves rightwards through the pure liquid region and into the two-phase region while another part of the initial pressure pulse is reflected backwards towards the inlet.

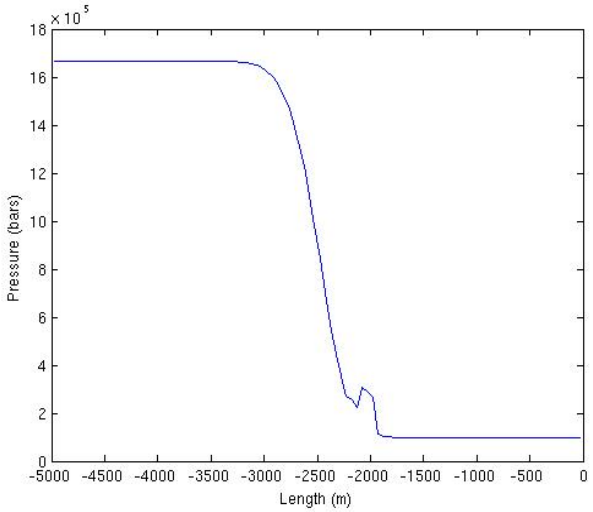
After analyzing the following plots, we can see that after this “instantaneous” increase of the inlet liquid flow rate, a right-going pressure wave is generated. In the pure liquid region, the pressure pulse has the same characteristics as the ones discussed in the first simulations of this. An extremely high pressure pulse is generated and it travels with high velocity. In the 1% gas region we recall the sonic velocity is strongly reduced. When the pressure pulse reaches the two-phase region, a large part of the pulse is reflected due to changes in the mixture density and acoustic velocity. Furthermore, we see from the plots that a strong shock is formed which continues to travel to the right in the two-phase region with a strongly reduced velocity. The amplitude of the wave which continues to travel to the right is also strongly reduced (from 17 bars initially to 2 bars)



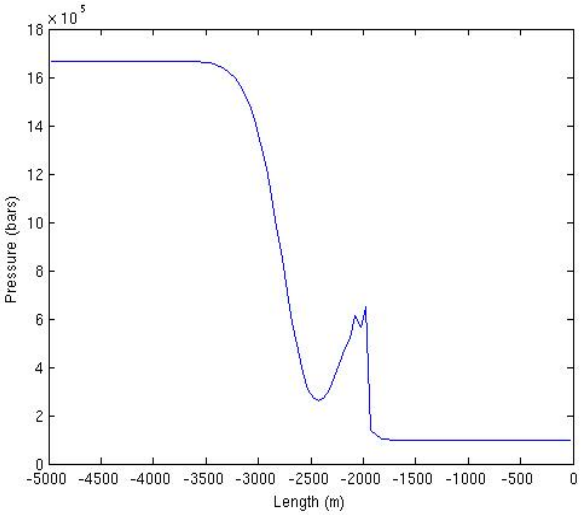
Plot 10.1: time = 2,0 s



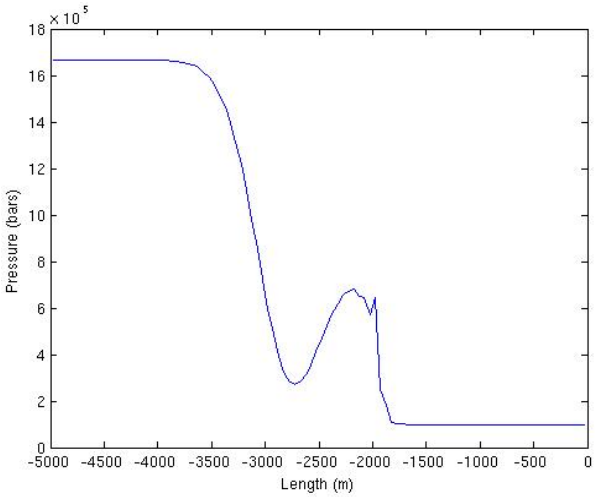
Plot 10.2: time = 2,2 s



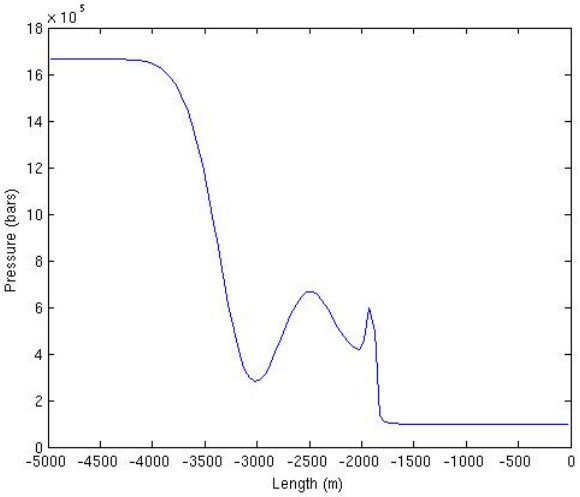
Plot 10.3: time = 2,4 s



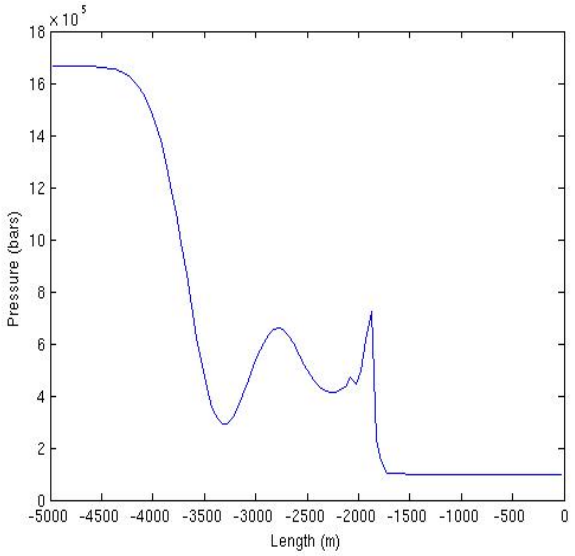
Plot 10.4: time = 2,6 s



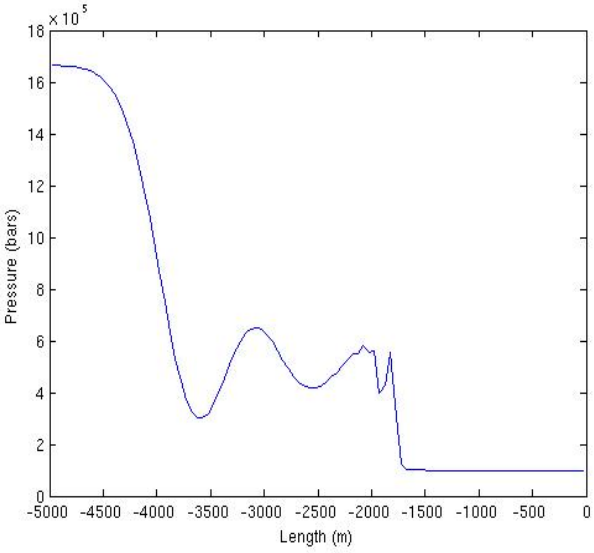
Plot 10.5: time = 2,8 s



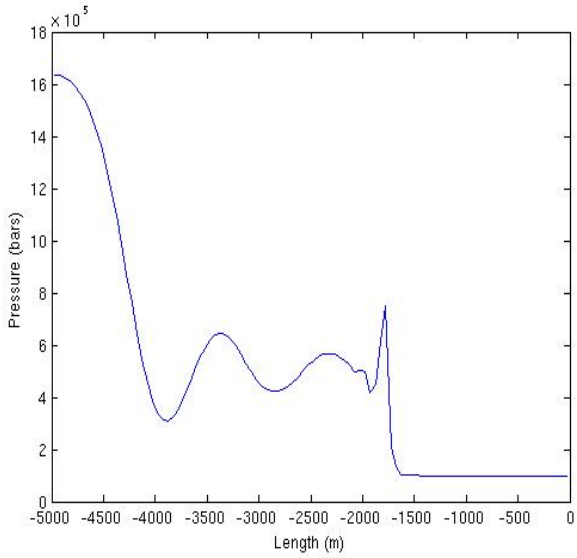
Plot 10.6: time = 3,0 s



Plot 10.7: time = 3,2 s



Plot 10.8: time = 3,4 s



Plot 10.8: time = 3,6 s

7.4 Simulation: Propagation of Pressure Pulses generated by Choke Adjustments

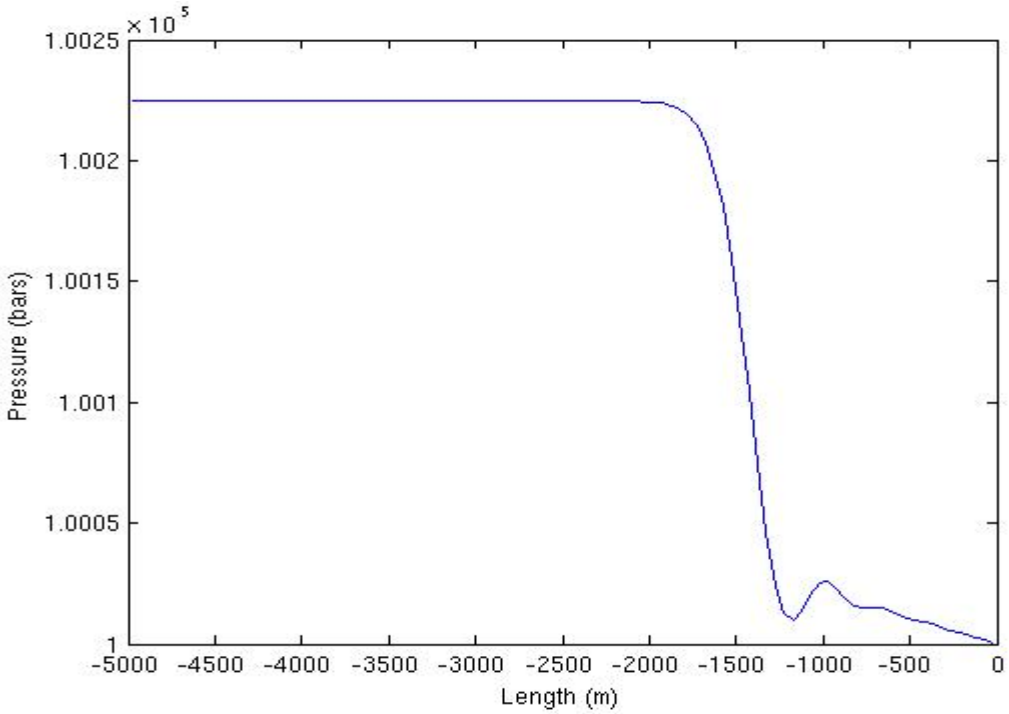
Thus far have we analyzed propagation of pressure pulses generated by increasing the inlet liquid flow. We shall in this section simulate propagation of pressure pulses generated by adjusting the choke, which is of importance in MPD. The second choke case is simulated with inlet flow and adjustment of the choke pressure at the outlet. In MPD operations during connections, situation arises where we have to increase the choke pressure to avoid kick. It is necessary when the well is filled with stagnant liquid to avoid that the bottomhole pressure drops below the porepressure. To avoid this kick scenario we have to increase the choke pressure in MPD operations. These following simulations will help us understand how long it takes to adjust the bottomhole pressure by regulating the choke pressure. After the simulations we can analyze what the effects are concerning the response time. The response time is the time it takes from the moment the choke pressure is adjusted to when it is registered in the bottom of the well. We will simulate two choke cases. In the first case we have a well with no inlet liquid flow where we generate the pressure pulse by increasing the choke pressure. We will simulate this when the well is 100% filled with liquid, and when the well contains 10% gas and 90% liquid. The second choke case includes inlet liquid flow and choke pressure adjustment.

7.4.1 Choke Case 1.1: Propagation of pressure pulses in a well with choke in pure liquid

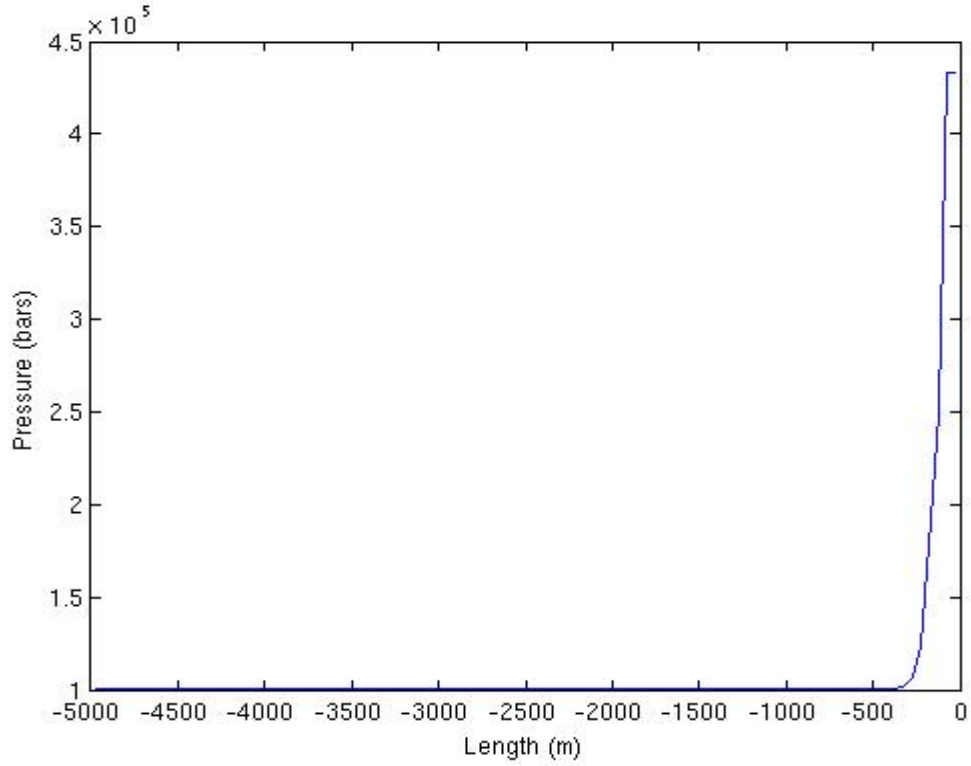
In this simulation we have a 5000m horizontal well filled with 100% liquid. The pressure pulse is generated by adjusting the choke at the outlet. Initial pressure is 1 bar. We increase the choke pressure from 1 bar to 4 bars using a startup time of 1 second. The pressure plot is plotted in a time series of 1 sec, 2 sec and 3 sec respectively. Friction is included in these simulations. Plot 11.1 shows the initial phase just before the pressure pulse development. If we consider Plot 11.3, the first steep increase in pressure at – 1500 meters corresponds to the pressure pulse (the pressure increase seen at the front). Here it is around 3,75 bars. The linear increase in pressure towards the outlet of the well represents probably an effect that is caused by the frictional terms in the model. Even if we start out with a stagnant well, fluid will be set in motion when we compress the system by increasing the outlet pressure. In Plot 11.4, we see that the outlet pressure is around 4 bars.

With respect to the pressure pulse itself, if we consider Plot 11.4, we observe that the magnitude of the pressure pulse in front has decreased to 3,4 bars from 3,8 bars in Plot 11.3. This attenuation of the amplitude of the pressure pulse is probably also caused by friction since it will dampen the amplitude of pressure pulses.

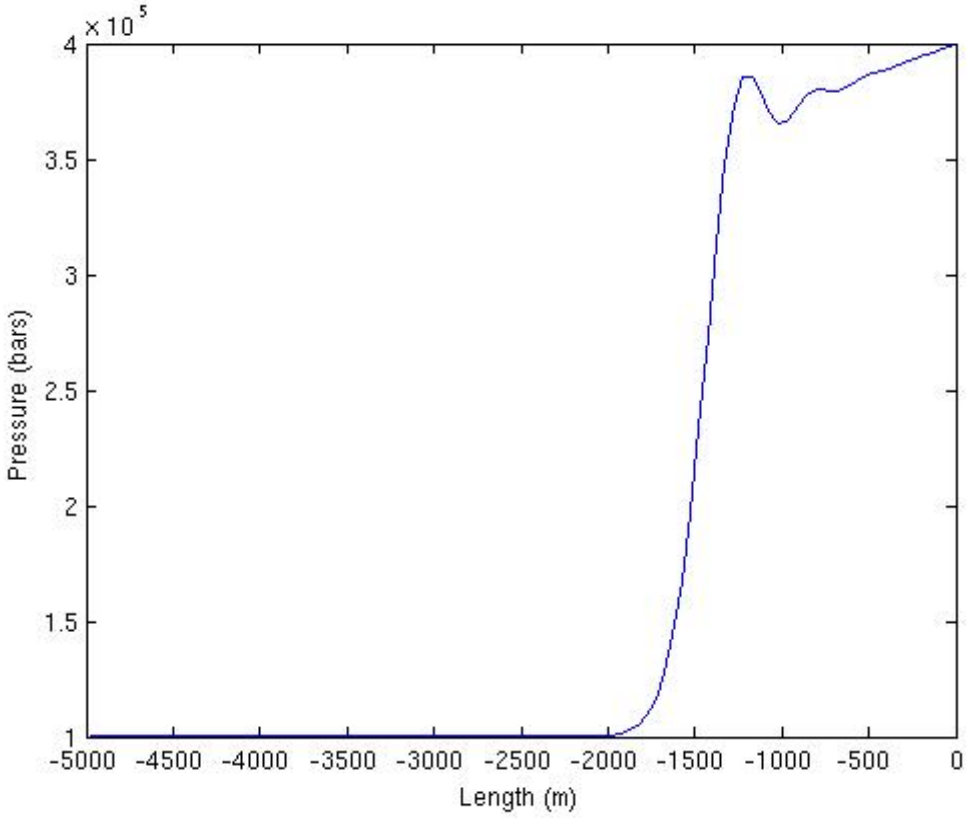
.



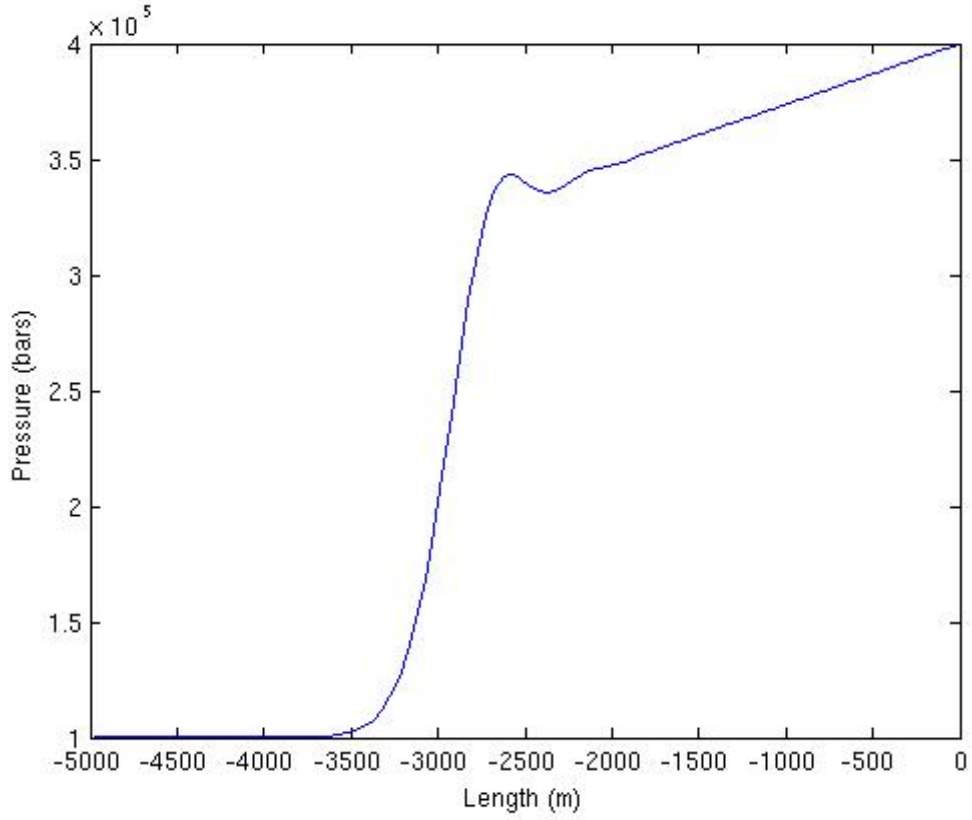
Plot 11.1. Choke case 1.1: time = 1 sec



Plot 11.2 Chokecase1,1: time = 1,1 sec



Plot 11.3. Choke case 1.1: time = 2 sec

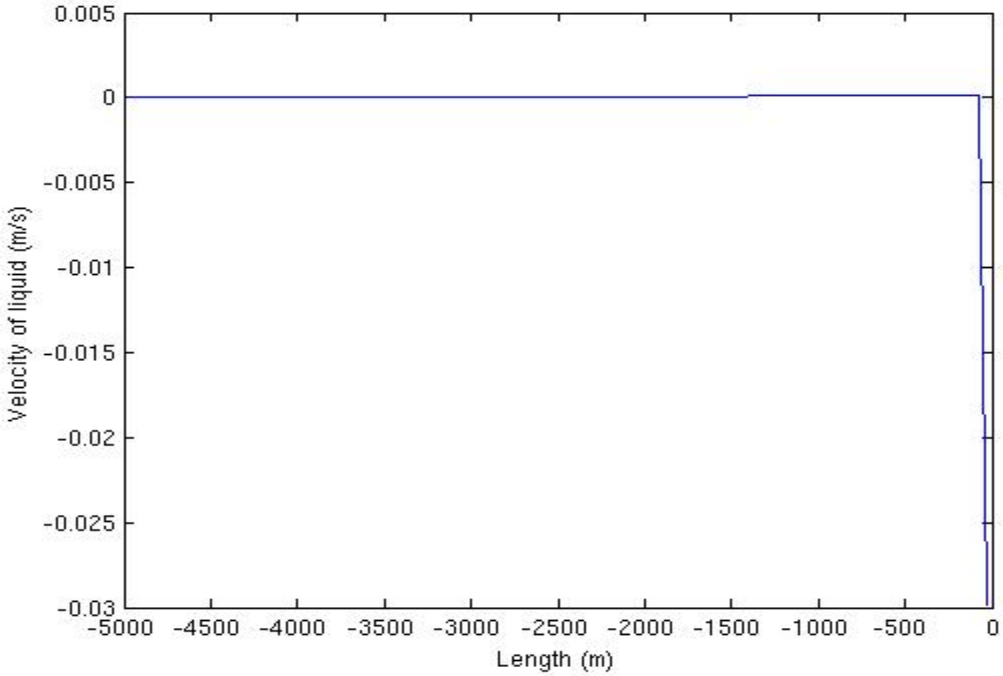


Plot 11.4. Choke case 1.1: time = 3 sec

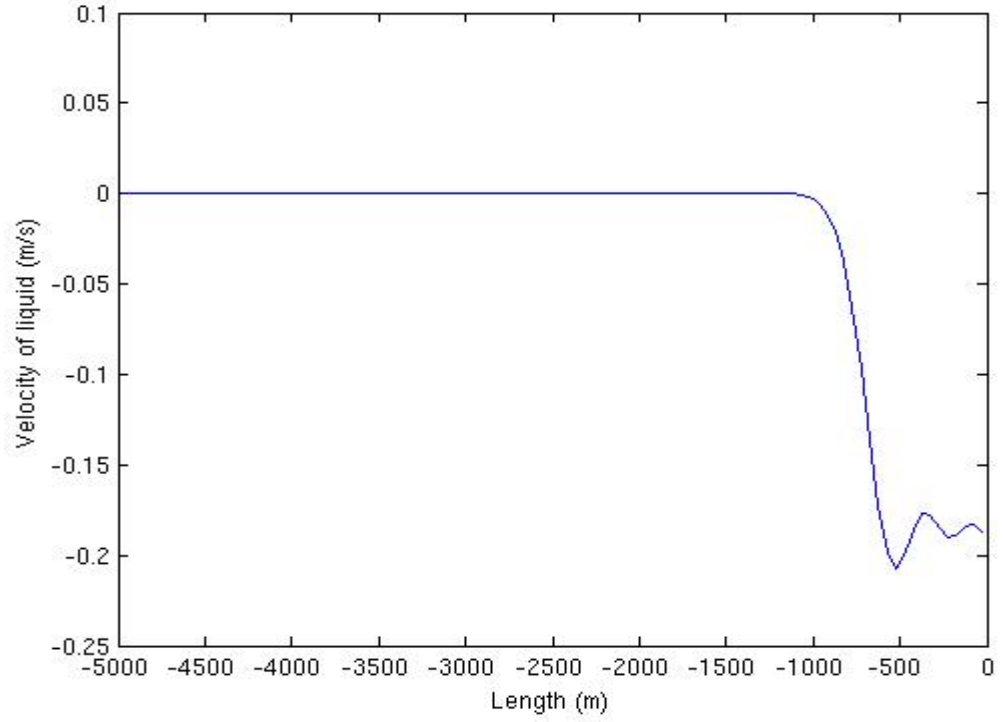
Propagation of pressure pulses in pure liquid are expected to travel with a much higher velocity than in two-phase regions. The plots above show the development of the pressure pulse in time.

7.4.2 Choke Case 1.2: Velocity profile for pressure pulses in 100% liquid w/ friction:

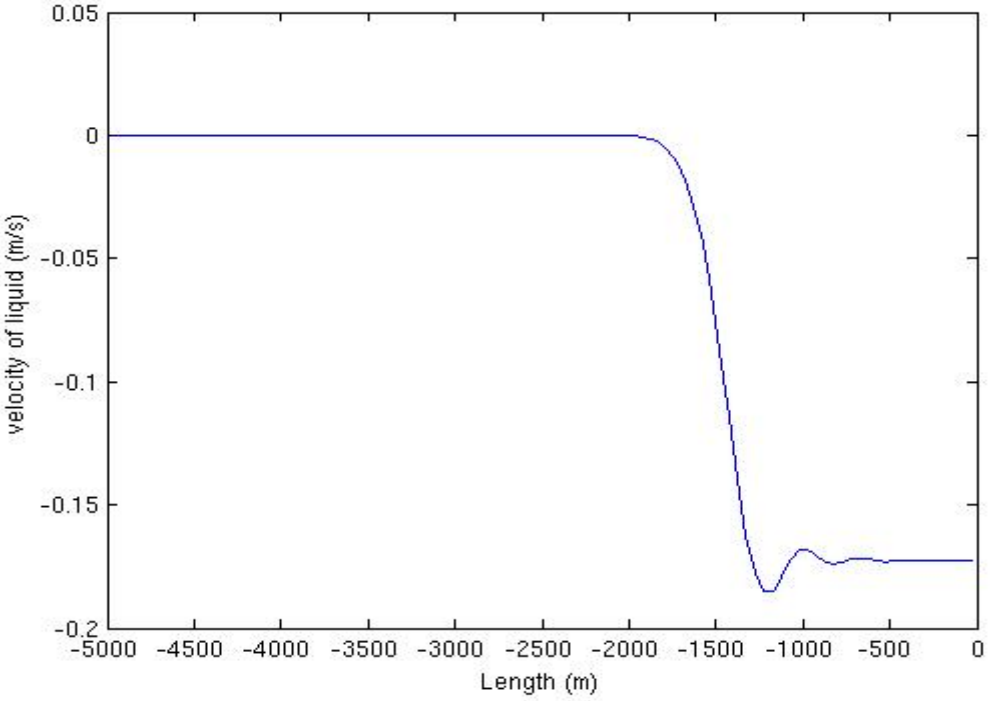
One should also be aware that the pressure pulse is associated with a corresponding change in fluid velocity as shown in the following four Plots. When the pressure pulse migrates from the outlet and towards the inlet of the pipe, this will lead to a negative velocity which expresses that the fluid is compressed due to the pressure increase which migrates. We can look upon this as a compressional wave. When pressure increases, fluid volumes will decrease and the velocity will become negative.



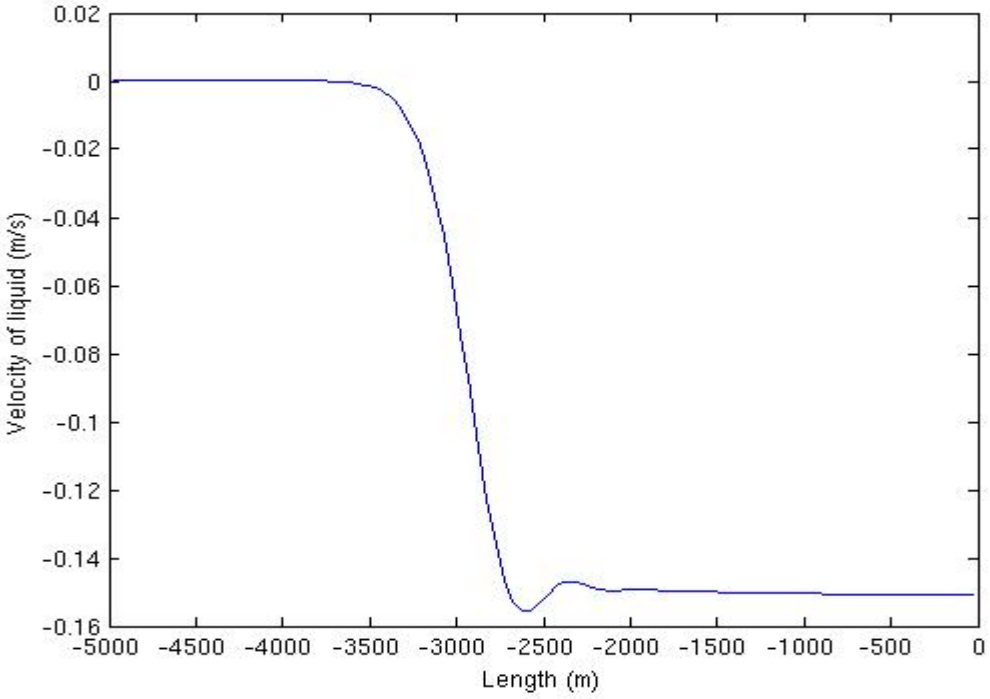
Plot 11.5: Choke case 1.2: Velocity profile, after time = 1s



Plot 11.6: Choke case 1.2: Velocity profile, after time = 1,5 s



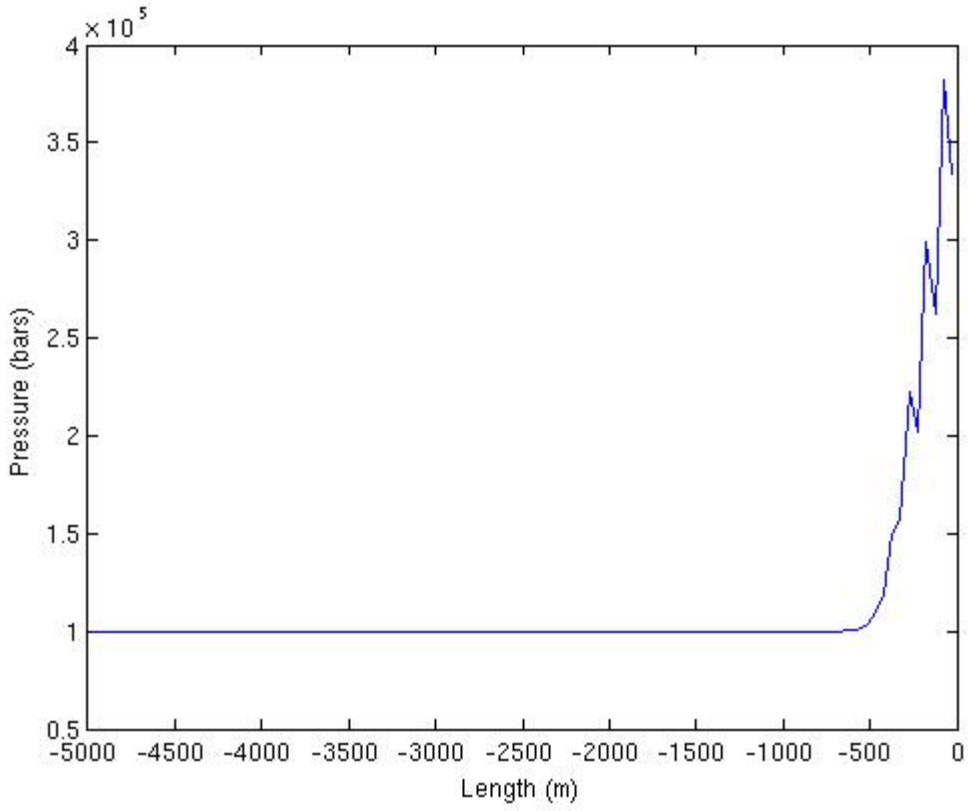
Plot 11.7: Choke case 1.2: Velocity profile, after time = 2s



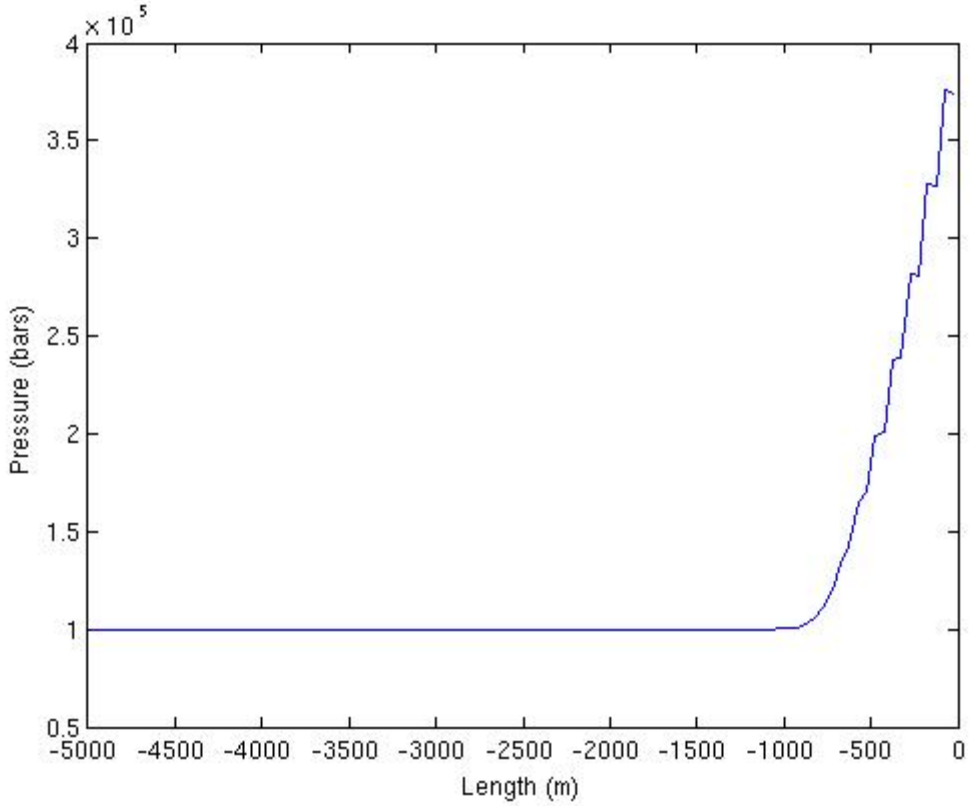
Plot 11.8: Choke case 1.2: Velocity profile, after time = 3s

7.4.3 Choke Case 1.3: Propagation of pressure pulses in a well with choke in 10% gas and 90% liquid.

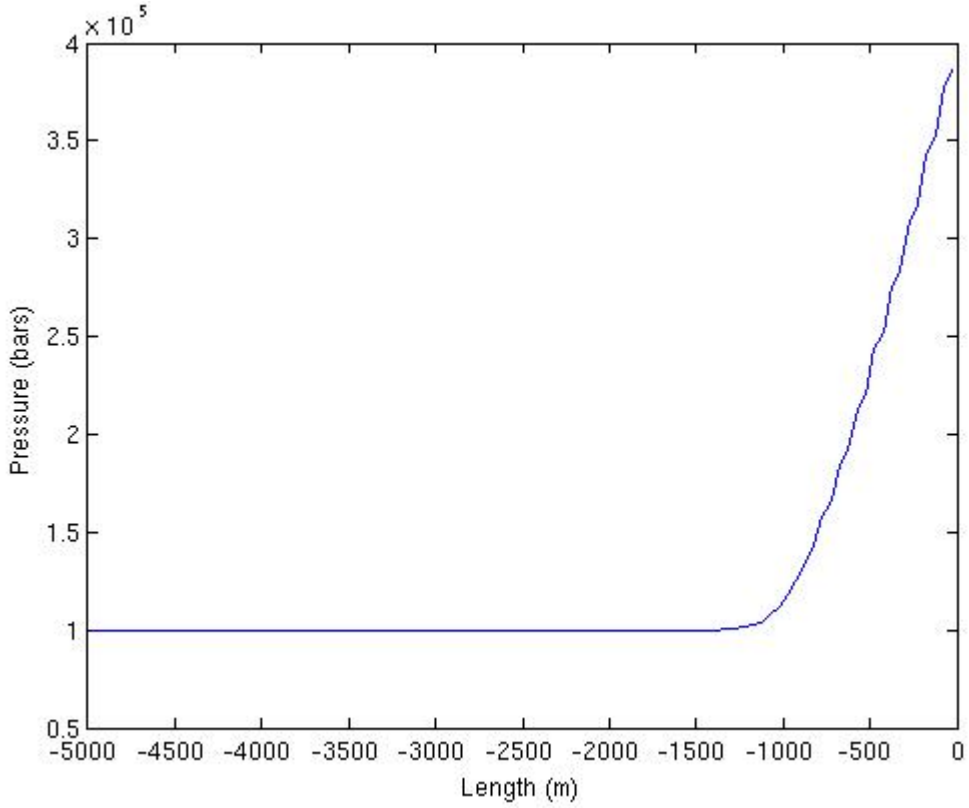
In this simulation we have a 5000m horizontal well filled with 10% gas and 90% liquid. The pressure pulse is generated by adjusting the choke at the outlet. Initial pressure is 1 bar. We increase the choke pressure from 1 bar to 4 bars in a startup time of 1 second. The pressure plots are plotted in a time series of 10 sec, 20 sec, 30 sec and 60 sec respectively. The time for this simulation differ from the previous one where only liquid was considered, because the pressure pulse has a much lower velocity in two-phase regions. Friction is included in these simulations. In an MPD system, presence of gas might easily occur since the system is designed for taking small gas kicks while drilling. Hence, in a long extended reach well, this is maybe an effect that one has to consider when working with an automated choke regulation system. After a given choke adjustment, one must give the well time to respond before an additional choke adjustment is introduced. If we consider Plot 12.4, it seems that the pressure pulse has propagated around 2000 meters in 60 second. This gives an average velocity of 33 m/s. This shows that for surface pressure conditions with presence of both liquid and gas, the pressure pulse propagation velocity becomes very low. However, if the well was vertical we would see that the pulse would move faster downwards in the well when the well pressures increase vs depth.



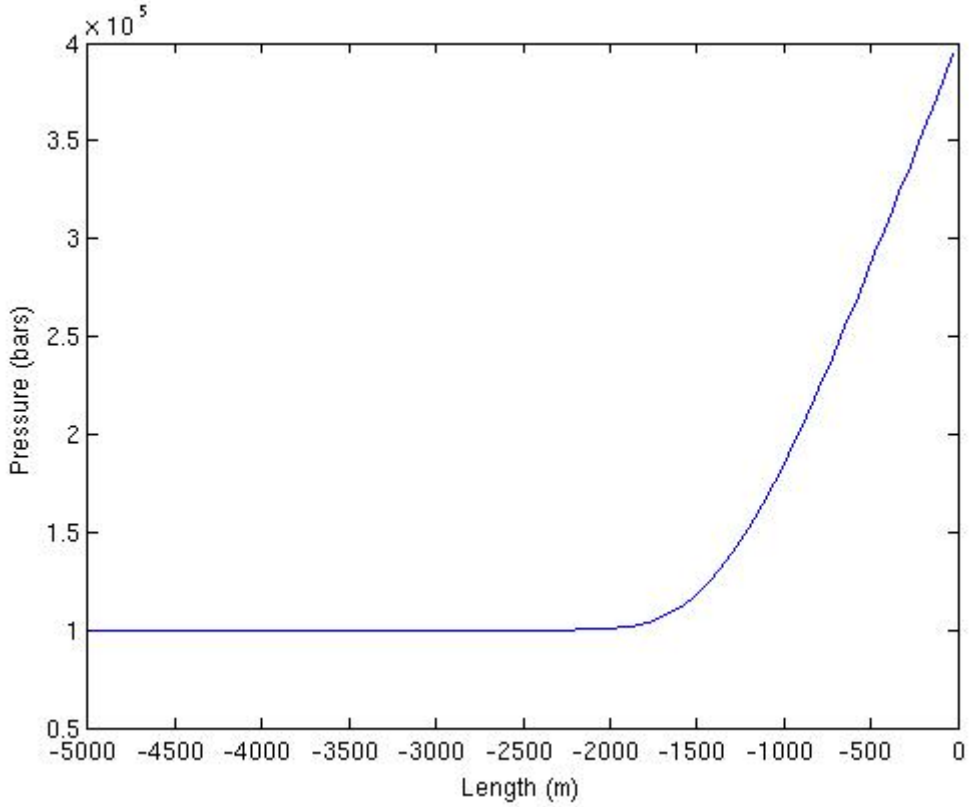
Plot 12.1. Choke case 1.2: time = 10 sec



Plot 12.2. Choke case 1.2: time = 20 sec



Plot 12.3. Choke case 1.2: time = 30 sec



Plot 12.4. Choke case 1.2: time = 60 sec

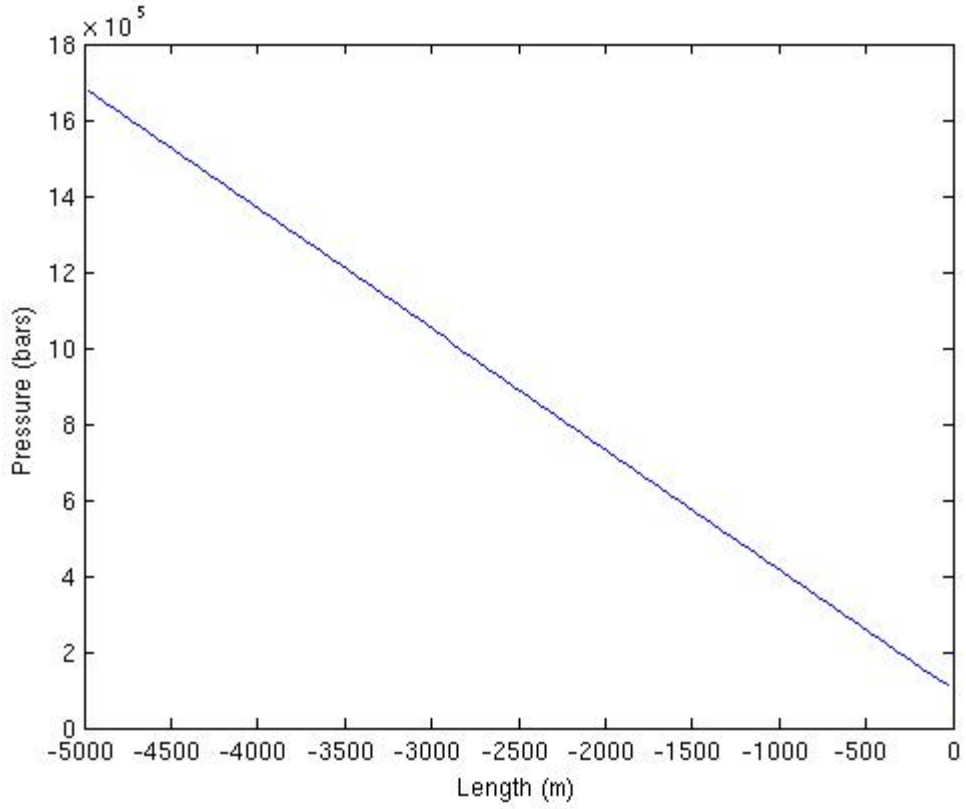
7.4.4 Choke Case 2.1: Propagation of pressure pulses in a well with choke & pump

In this simulation we consider a 5000m long horizontal well filled with 100% liquid. The inlet liquid flow rate is 25 kg/s. We generate a pressure pulse by closing the choke and increasing the choke pressure from 1 to 4 bars in one second. If we consider plot 13.1, we see a linear graph with constant decrease in pressure throughout the well. This represents the flow conditions in the well just before the pressure pulse is initiated.

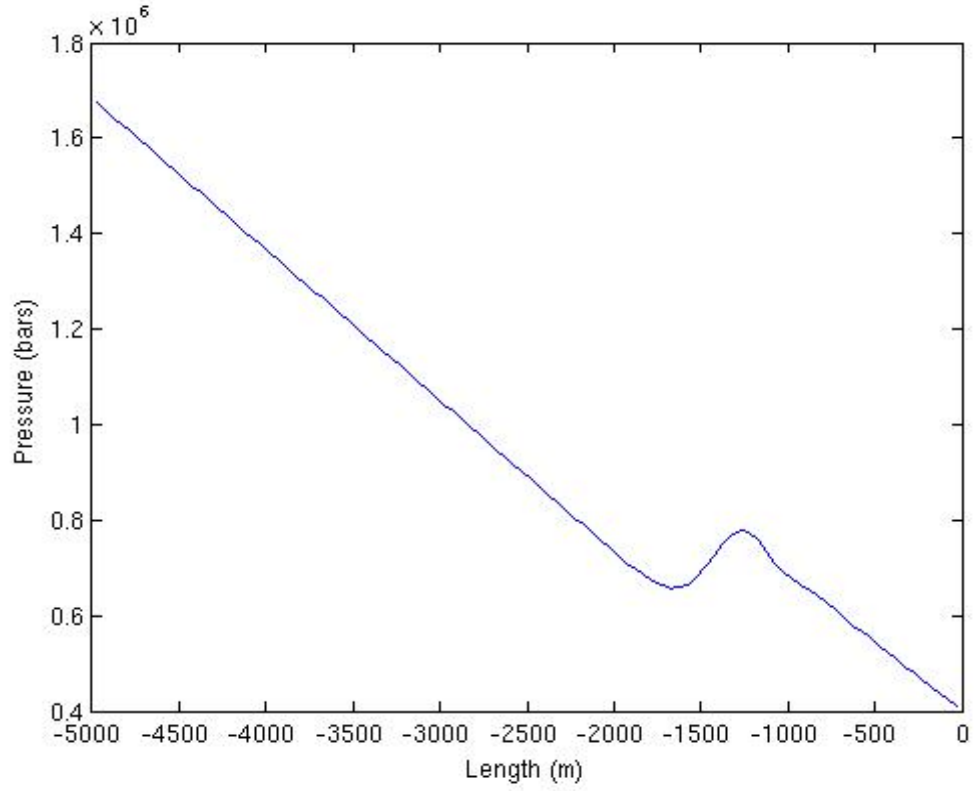
This linear decrease is due to the friction in the well. From this we can deduce that the friction in the well is 17 bars.

After 61 seconds the pressure adjustment of the choke has an effect and from plots 13.2, 13.3, 13.4 and 13.5 we see the generated pressure pulse moving from the outlet to the inlet. In Plot 13.6 we see the pressure pulse reaching the inlet at 20.05 bars. If we compare plot 13.1 and plot 13.6, we gather that the generated pressure pulse reaches the inlet with an amplitude of 20 – 17 bars = 3 bars. This corresponds with the increase in choke pressure of 3 bars (4 – 1 bars = 3 bars) at the outlet.

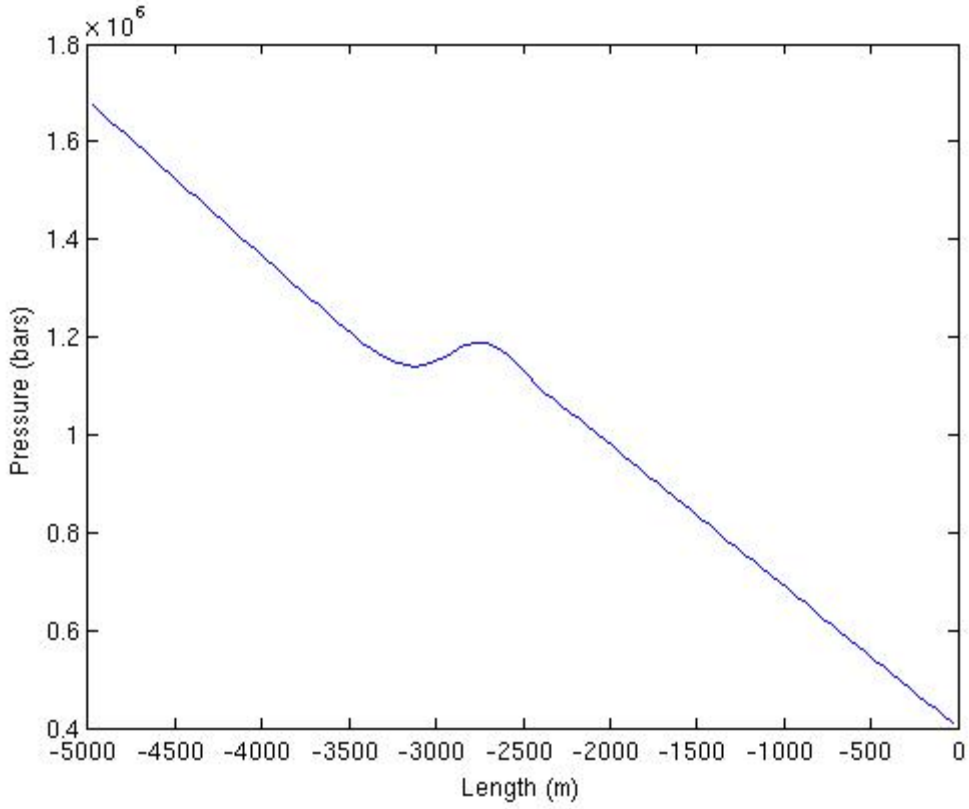
In MPD operations the wells are usually vertical wells. This simulation deals with a horizontal well. In a vertical well pressure will increase with TVD. There are higher pressures in the bottom of a vertical well than on the top. Hence, the pressure pulses generated with choke will move at a higher velocity in a vertical well as the pressure pulse reaches the inlet or more precisely bottom of the well.



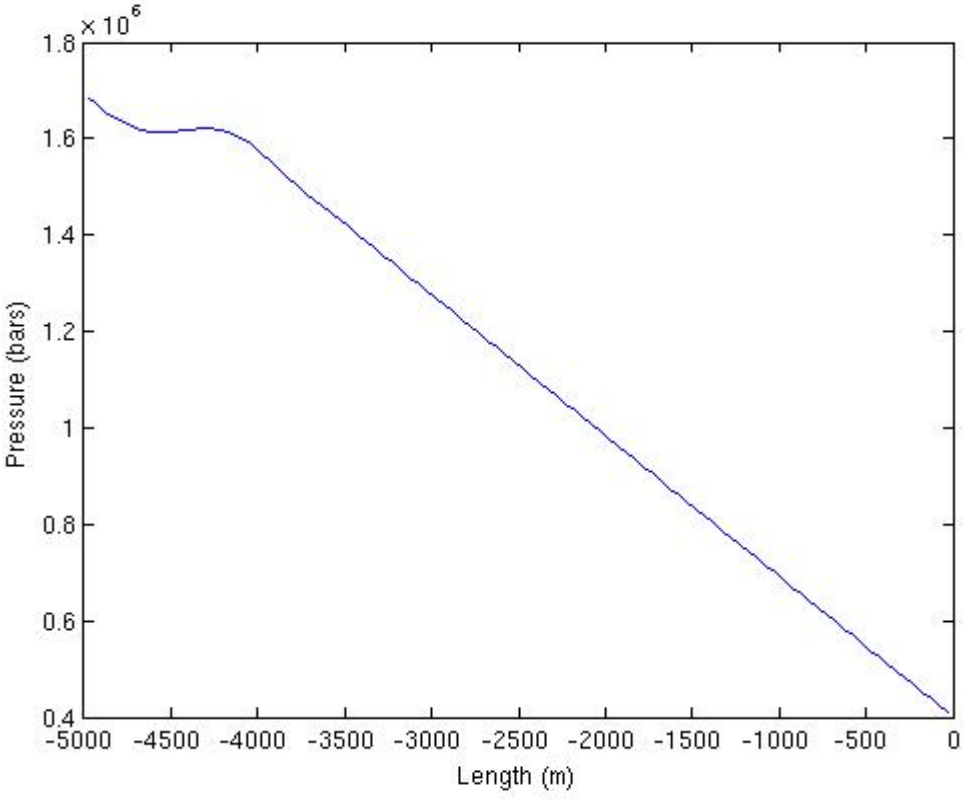
Plot 13.1. Choke case 2.1: time = 61 sec



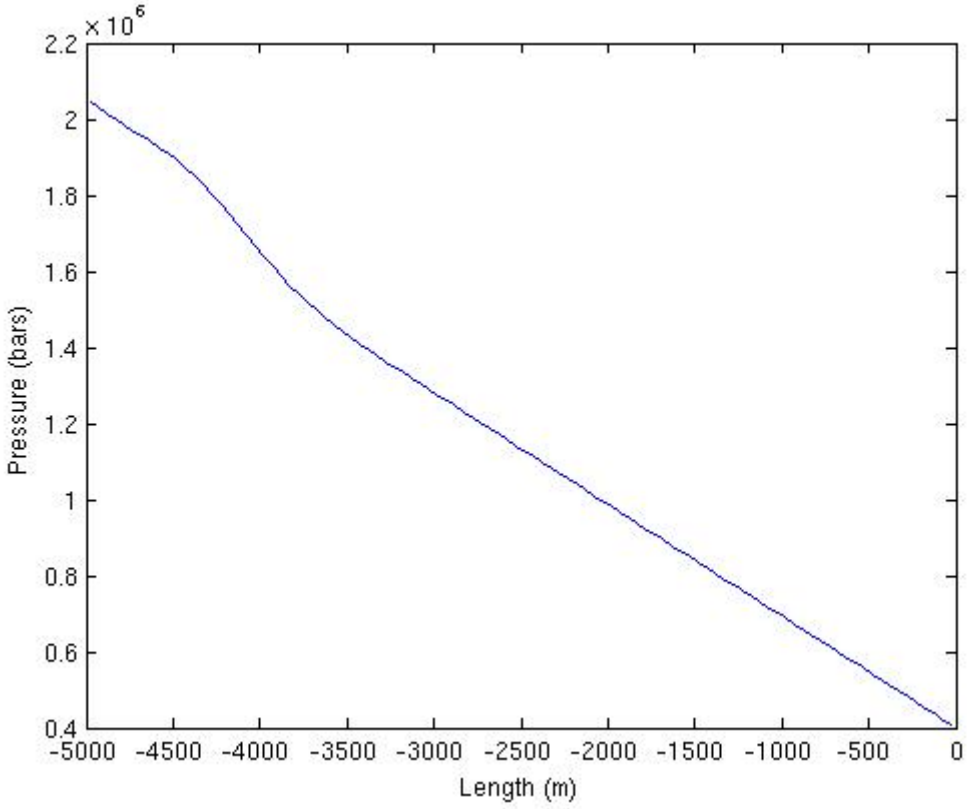
Plot 13.2. Choke case 2.1: time = 62 sec



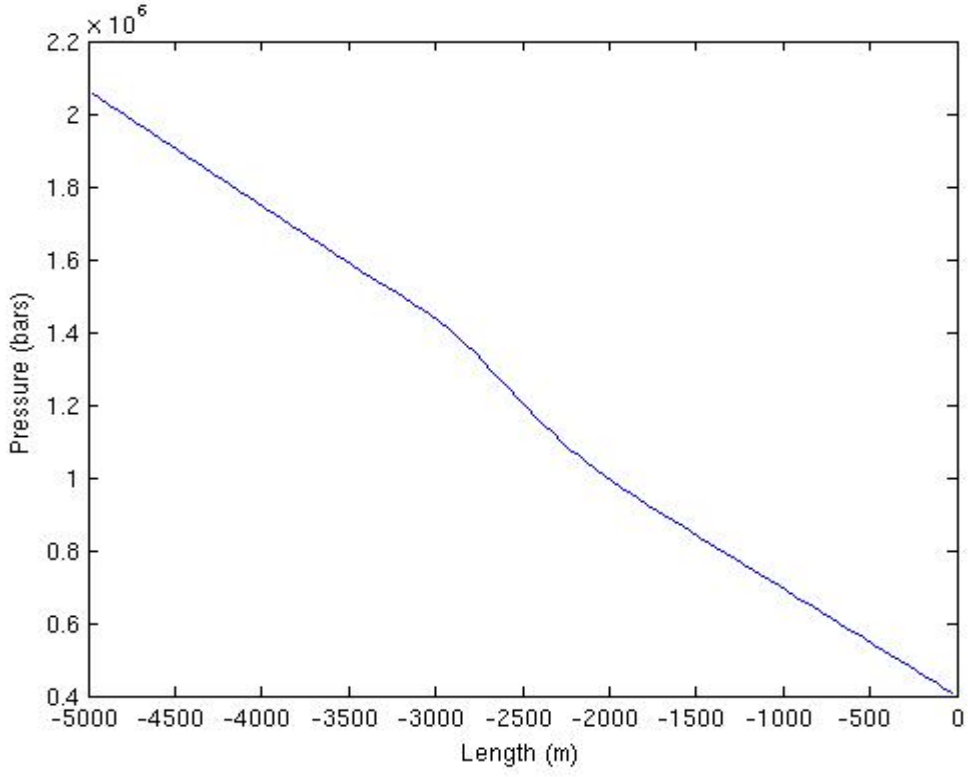
Plot 13.3. Choke case 2.1: time = 63 sec



Plot 13.4. Choke case 2.1: time = 64 sec



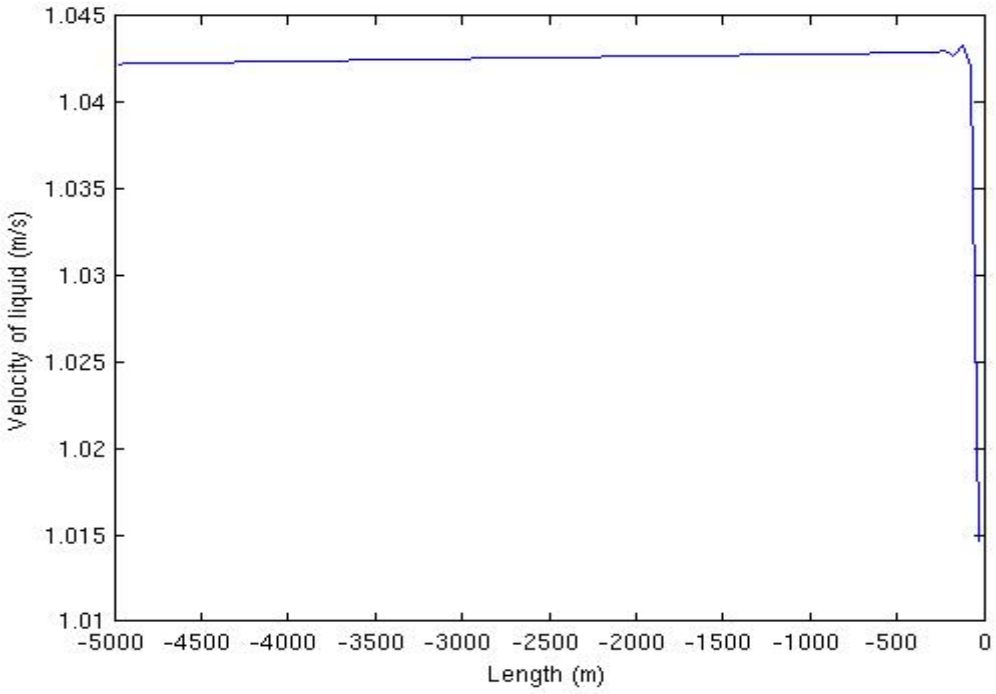
Plot 13.5. Choke case 2.1: time = 65 sec



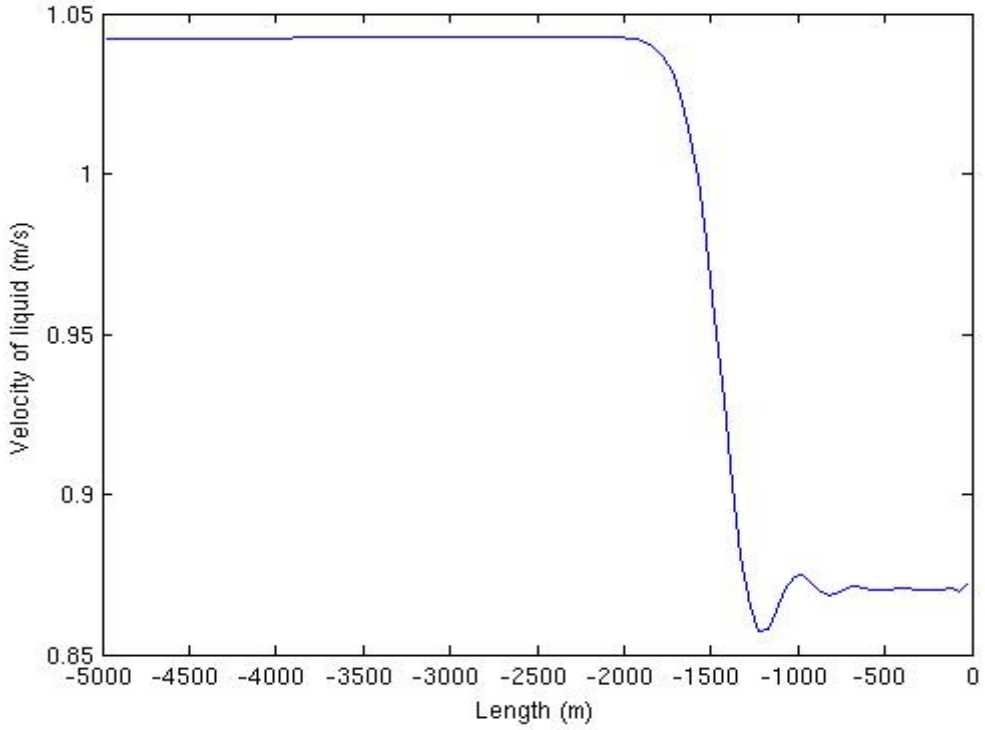
Plot 13.6. Choke case 2.1: time = 66 sec

7.4.5 Choke Case 2.2: Velocity profile for pressure pulses in liquid w/ Choke & pump:

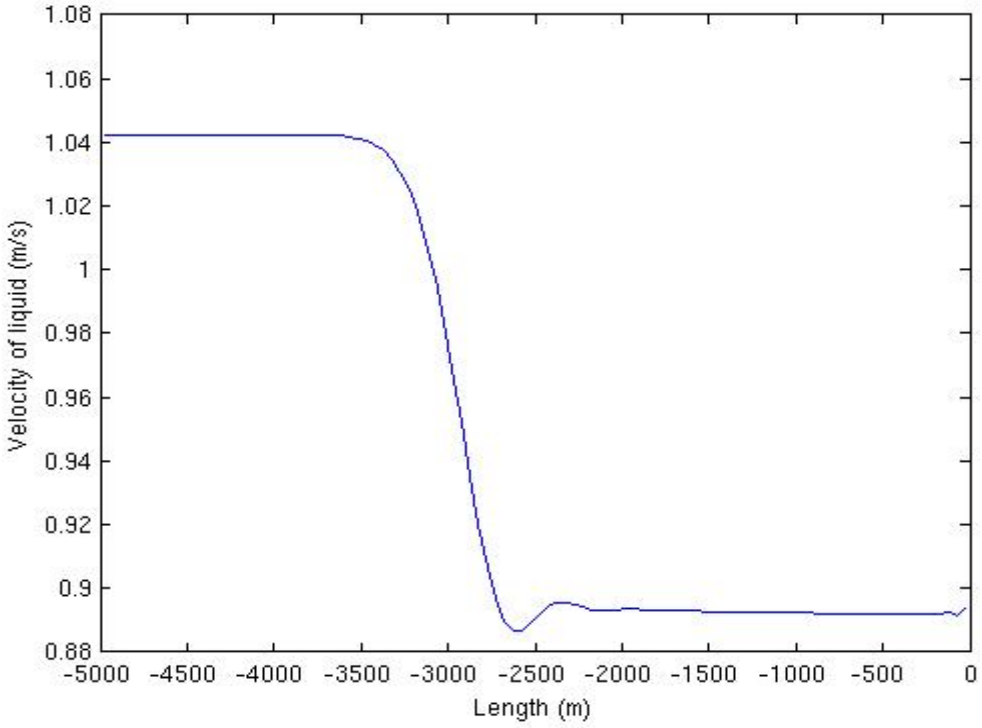
The corresponding change in fluid velocity for the pressure pulse is as shown in the following plots. Also here we see that the original fluid velocity is reduced as the pressure pulse migrates towards the inlet due to the compression effect that is exerted by the pressure pulse as it migrates towards the inlet. We observe that the fluid velocity is reduced with almost 10 percent. The massrate M in kg/s is constant but we know that the relation between massrate, density and velocity is given by $M = \rho(p) \cdot v$ so when the density ρ is increased due to increased pressure the flow velocity v will be reduced to maintain mass conservation. Hence, a pressure pulse will always be associated with a corresponding change in fluid velocity.



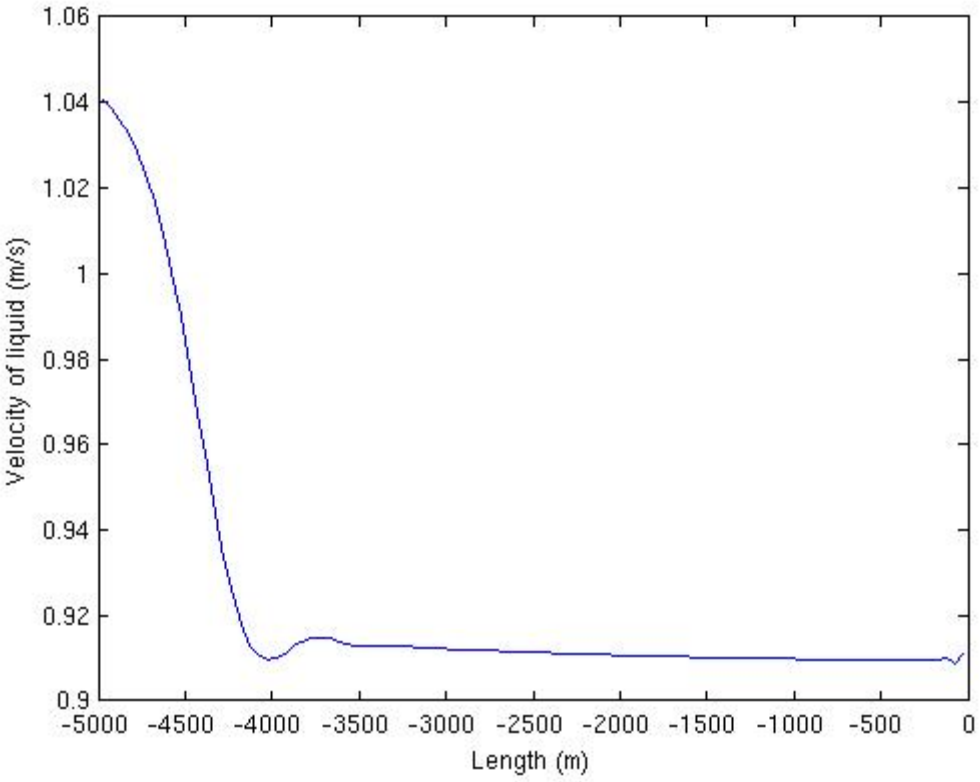
Plot 13.7: Choke case 2.2: Velocity profile, time = 61s



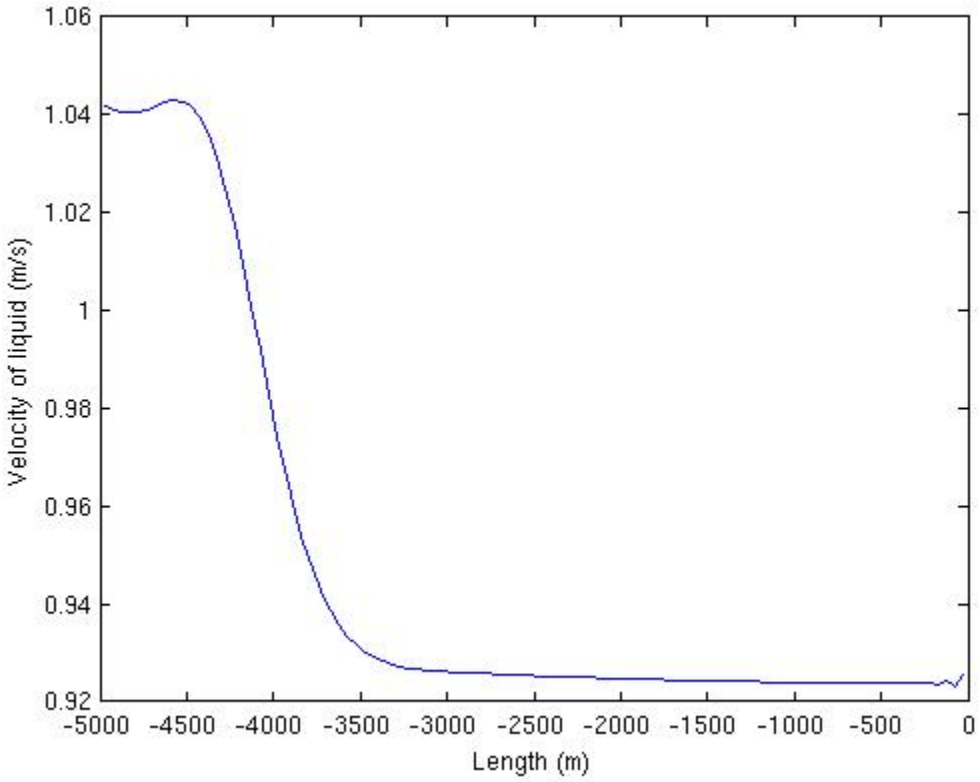
Plot 13.8: Choke case 2.2: Velocity profile, time = 62s



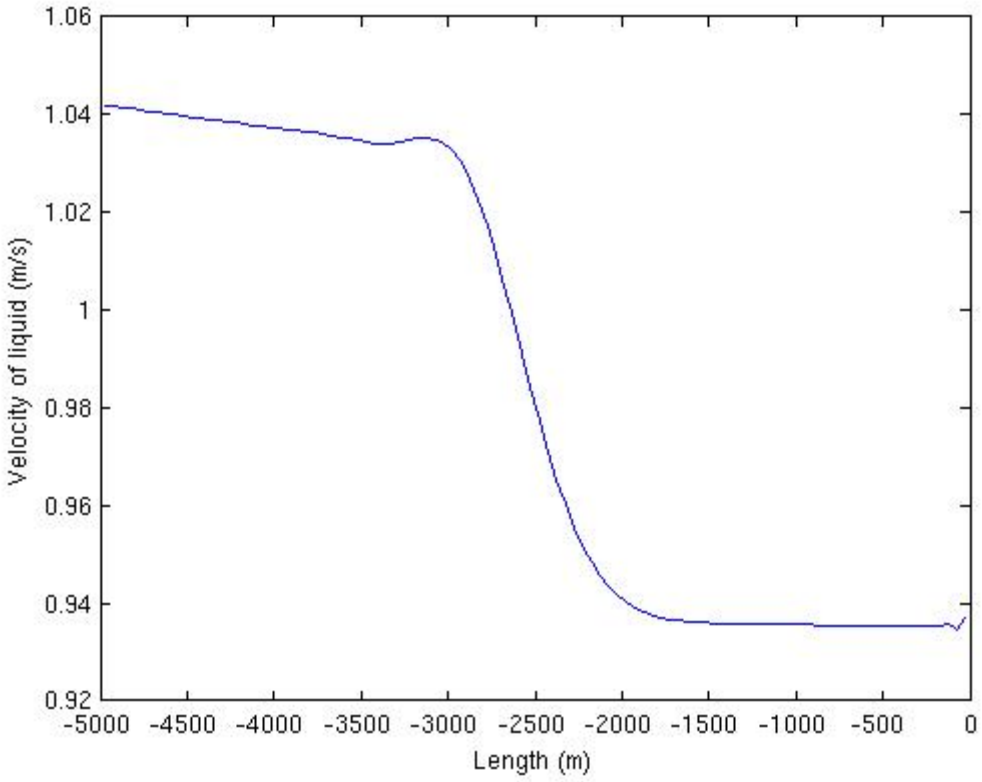
Plot 13.9: Choke case 2.2: Velocity profile, time = 63s



Plot 13.10: Choke case 2.2: Velocity profile, time = 64s



Plot 13.11: Choke case 2.2: Velocity profile, time = 65s



Plot 13.12: Choke case 2.2: Velocity profile, time = 66s

8. Discussion and Conclusion

- Pressure pulse propagation caused by pump start up and choke valve adjustments is studied in the simulations. Studying the propagation of pressure pulses generated by adjusting the choke, is of importance in MPD because situations arise where we have to increase the choke pressure to avoid kick.
- In an MPD backpressure system, it is common to try to control the pressure in the well at a certain location. This could be at the place where there is small margin between the pore and the fracture pressure. By measuring the pressure in the well, proper choke adjustments are then carried out to compensate if the measured pressure deviated from the pressure that is wanted at the location. In this context, one should be aware that it will take time before the choke pressure adjustment is felt at the sensor measuring the pressure. There will be no point in making a new choke pressure adjustment before the previous adjustment is "felt" at the sensor and gives advice on what the next step should be. There is a certain lag time involved caused by the sonic wave propagation velocity.
- It is of interest to mention that the speed of sound is 1500 m/s in pure liquid regions, but it drops dramatically in two-phase regions. In the region of 1% gas, the speed of sound is approximately 100 m/s. Assuming pressure = 1 bar, the gas density will then be 1 kg/m^3. The liquid density will be 1000 kg/m^3. If we assume no slip condition K = 1 and $\alpha_g = 0.01$. This gives w = 100 m/s. This can be an important fact to have in mind when considering MPD operations. If there are small amounts of gas in the MPD system then the pressure pulses will take longer to propagate and that might have to be taken into account when for instance regulating the choke. Since the MPD system is based on the possibility of taking small kicks this may be an issue.
- It is evident from the plots that an increase in gas content will lead to even more reduction in the propagation velocity of the pressure pulses. There is a significant velocity change when we only have 1% gas compared to 10 % gas in the well. In addition, the sonic velocity of the mixture will depend heavily on the pressure conditions in the well (see Figure 29 & 30).
- By analyzing the plots, we observe the effect of the friction. The inlet pressure will increase as more and more fluid is set in motion. In addition, we observe that the amplitude of the pressure pulse itself is dampened vs time due to friction. Plot 4.3 was simulated with friction included. We see that the inlet pressure is around 23 bars while in the no friction case, it was around 17 bars. This difference is caused by the friction in the system and this additional pressure is what is required to move the fluid when friction working against

the flow is present. If we compare Plot 4.1, 4.2 and 4.3 we also see that the inlet pressure increases vs time. The reason for this is the friction increases as the pulse propagates further down the well. This is related to that the pressure pulse also reflects a velocity change as more and more fluid in the well is set into motion. With respect to the pressure pulse itself, if we consider Plot 4.3, we observe that the magnitude of the pressure pulse in front has decreased to 12-13 bars from 16-17 bars in Plot 4.1. This attenuation of the amplitude of the pressure pulse is also caused by friction since it will dampen the amplitude of pressure pulses.

- During simulations it was experienced that the current version of the AUSMV scheme is not able to handle reflection of pressure pulses at the well boundaries. The boundary treatment does not seem to work for a system where friction is not considered. The pulses are not transmitted un-attenuated back and forth in the system. For the one phase flow system, the pulse is killed after one “travel” back and forth in the well. When we include the friction, the pressure developments seem to behave more in the sense we expect. However, one needs to investigate further if this boundary treatment is good enough to handle pressure pulses properly and possibly one should use compatibility relations instead to solve the boundary problem.
- The simulations show that the sonic velocity depends on both gas fraction and pressure. If we operate with gas in a well, typically we find that the sonic velocity is reduced most at the top of the well. As the pressure increases with well depth, the sonic wave propagation velocity increases. Considering a 5000m vertical well and an average sound velocity of 300 m/s, it will take approximately 17 sec from the moment we adjust the choke, until the sensor picks up the signal at the bottom of the well. With an average sound velocity of 200 m/s, the response time is approximately 25 sec. Therefore, if we adjust the choke by making fast updates based on frequent measurements in long wells, there is a possibility that this “timelagg” is a factor which must be taken into account. This will particularly apply to underbalanced drilling systems where we know there will be significant volumes of gas in the well.
- In an MPD system, presence of gas might easily occur since the system is designed for taking small gas kicks while drilling. Hence, in a long extended reach well, this is maybe an effect that one has to consider when working with an automated choke regulation system. After a given choke adjustment, one must give the well time to respond before an additional choke adjustment is introduced. If we consider Plot 12.4, it seems that the pressure pulse has propagated around 2000 meters in 60 second. This gives an average velocity of 33 m/s. This shows that for surface pressure conditions with presence of both liquid and gas, the pressure pulse propagation velocity becomes very low. However, if the well was vertical we would see that the

pulse would move faster downwards in the well since the well pressures increase vs depth.

- The simulations in the thesis deals with a horizontal well. In a vertical well pressure will increase with TVD. There are higher pressures in the bottom of a vertical well than on the top. Hence, the pressure pulses generated with choke will move at a higher velocity in a vertical well as the pressure pulse reaches the inlet or more precisely bottom of the well.

9. References

1. Steinar Evje and Kjell Kåre Fjelde: “Hybrid Flux-Splitting Schemes for a Two-Phase Flow Model”, Journal of Computational Physics 175, 2002, p. 674-701.
2. Kjell Kåre Fjelde, Rolv Rommetveit, Antonio Merlo and Antonio C. V. M. Lage: “Improvements in Dynamic Modeling of Underbalanced Drilling” – SPE 81636, Presented at the IADC/SPE Underbalanced Technology Conference and Exibition Houston, Texas, 25- 26 March 2003.
3. Kjell Kåre Fjelde: “Modelling of Wellflow”, PET510, Compendium.
4. Maiken Ravndal: “Models for Dynamic Kill of Blowouts”, Master’s Thesis, The University of Stavanger, June 2011.
5. Kjell Kåre Fjelde and Steinar Evje, Work note: “The AUSMV Scheme – A Simple but Robust Model for Analyzing Two-Phase Flow”, University of Stavanger, 17 August 2010.
6. Kjell Kåre Fjelde and Kenneth Hvistendahl Karlsen, work note: “A Comparative Study of some Fully Numerical Shock-capturing Schemes for Simulating Two-phase flow”, University of Bergen, 5 April 2000.
7. Managed Pressure drilling: What it is and What it is not, by Kenneth P. Malloy, C. Rick Stone, George H. Medley, Don Hannegan, Oliver Coker, Don Reitsma, Joseph Kinder, Johan Eck-Olsen, John McCaskill, James May and Paul Sonneman. SPE 122281. This paper was presented at the IADC/SPE Managed Pressure Drilling and Underbalanced Operations Conference and Exhibition held in San Antonio, Texas, 12-13 February 2009.
8. Underbalanced Drilling: Praises and Perils, by D.B. Bennion, F.B. Thomas and R.F. Bletz. SPE52889. This paper was presented at the 1996 SPE Permian Basin Oil and Gas Recovery Conference held in Midland, Texas, 27-29 March.
9. Kjell Kare Fjelde, 2011. Lecture notes (Managed Pressure Drilling – what is it.) Course: MPE 710 Advanced Drilling technology.
10. Managed Pressure Drilling in Marine Environments- Case Studies by Don M. Hannegan. SPE 92600. This paper was prepared for presentation at the SPE/IADC Drilling Conference held in Amsterdam, Netherlands, 23-25 February 2005.
11. Managed Pressure drilling: A Multi-Level Control Approach, by Øyvind Breyholtz, Gerhard Nygaard, Hardy Siahaan and Michael Nikolaou. SPE 12851. This paper was presented at the SPE Intelligent Energy Conference and Exhibition held in Utrecht, Netherland, 23-25 March 2010.
12. Managed Pressure Drilling – What is it anyway? By Kenneth P. Malloy. Originally appeared in: World Oil, March 2007, pages: 27-34.
13. NORSOK STANDARD, D-010. Rev.3, August 2004.
14. SPE 2006-2007 Distinguished Lecturer Series, by Don M. Hannegan, Weatherford International Ltd.
15. Vieira, P., Amone, M., Russel, B., Cook, I., Moyse, K., Torres, F., Qutob, H., Chen, Y., and Chen, Q.: “Constant bottomhole pressure: Managed-pressure drilling technique applied in an exploratory well in Saudi Arabia,” Paper SPE/IADC 113679 presented at

the SPE/IADC Managed Pressure Drilling and Underbalanced Operations Conference and Exhibition, Abu Dhabi, U.A.E., 28-29 January 2008.
16. Bansal, R.K., Brunnert, D., Todd, R., Bern, P.A., Baker, R.V., and Richard, C.: "Demonstrating managed-pressure drilling with the ECD reduction tool," Paper SPE/IADC 105599 presented at the SPE/IADC Drilling Conference, Amsterdam, The Netherlands, 20-22 February 2007.
17. Rehm, B., Schubert, J., Hagshenas, A., Paknejad, A. S. & Hughes, J. (2008). Managed Pressure Drilling. Houston, Texas: Gulf Publishing Company.
18. Managed Pressure Drilling from Floaters – Existing Technology Where do we go from here, by Trond Stødle. (M.Sc Thesis), UIS, 05 june 2013.
19. Drilling in depleted reservoirs, by Line Hoff Nilsen, StatoilHydro, ESRA seminar, 5 March 2009.
20. Paco Vieira, Maurizio Arnone, and Fabian Torres, Weatherford International, and Fernando Barragan, AGR Group. : "Roles of Managed Pressure Drilling Technique in Kick Detection and Well Control—The Beginning of New Conventional Drilling Way" SPE/IADC 124664, Presented at SPE/IADC Middle East Drilling Technology Conference and Exhibition held in Manama, Bahrain, 26-28 October 2009
21. Managed Pressure Drilling Techniques, Equipment and Applications. By Erdem Tercan. (M.Sc Thesis), May 2010
22. Automated Well Control Using MPD Approach, by Trygve Birkeland. (M.Sc Thesis), June 2009.
23. Riserless Drilling (Managed Pressure Drilling), by Ahsan Khan. (M.Sc Thesis), June 2012.
24. Goodwin, R., Medley, G., and Reynolds, P.B.B.: "Understanding MPD complexity levels, Hart's E&P 81 (2008), No. 10, 37-39.
25. Jenner, J.W., Elkins, H.L., Springett, F., Laurie P.G., and Wellings, J.S.: "The continuous circulation system: An advance in constant-pressure drilling," Paper SPE 90702 presented at the SPE Annual Technical Conference and Exhibition, Houston, Texas, U.S.A., 26-29 September 2004.
26. PhD dissertation of Øyvind Breyholtz:"Managed Pressure Drilling: Improved Pressure Control through Model Predictive Control". University of Stavanger, Spring 2011.
27. S. B. Gavage. Analyse numerique des modeles hydrodynamiques d'ecoulements diphasiques instationnaires dans les reseaux de production petroliere. These, ENS Lyon France, 1991.
28. B. Theron. Ecoulements diphasique instationnaires en conduit horizontale. These, INP Toulouse, France, 1989.
29. "Numerical Schemes for Complex Nonlinear Hyperbolic Systems of Equations", by Kjell Kåre Fjelde; PhD Thesis, University of Bergen, April 5, 2000.

10. Appendices

Appendix 1

The AUSMV Code:

Note: The additions made to the AUSMV code are marked with Red in Appendix 2.

```
% Transient two-phase code based on AUSMV scheme: Gas and Water
% The code can handle area changes. The area changes are defined
inside
% the cells such that the where the fluxes are calculated, the
geometry is
% uniform.

clear;

% Geometry data/ Must be specified
welldepth = 5000;
nobox = 100; %Number of boxes in the well
nofluxes = nobox+1;
dx = welldepth/nobox; % Boxlength
%dt = 0.005;

% Welldepth array
x(1)= -1.0*welldepth+0.5*dx;
for i=1:nobox-1
 x(i+1)=x(i)+ dx;
end

dt= 0.005  % Timestep
dtdx = dt/dx;
time = 0.0;
endtime = 30; % Rime for end of simulation
nosteps = endtime/dt;  %Number of total timesteps
timebetweensavingtimedata = 1.0;  % How often in s we save data vs
time for plotting.
nostepsbeforesavingtimedata = timebetweensavingtimedata/dt;

% Slip parameters used in the gas slip relation. vg =Kvmix+S
k = 1.2;
s = 0.5;

% Viscosities (Pa*s)/Used in the frictional pressure loss model.
viscl = 0.05; % Liquid phase
viscg = 0.000005; % Gas phase

% Density parameters. These parameters are used when finding the
% primitive variables pressure, densities in an analytical manner.
```

```
% Changing parameters here, you must also change parameters inside
the
% density routines roliq and rogas.

% liquid density at stc and speed of sound in liquid
  dstc = 1000.0;   %Base density of liquid, See also roliq.
  pstc = 100000.0;
  al = 1500; % Speed of sound/compressibility of liquid phase.
  t1 = dstc-pstc/(al*al);
% Ideal gas law constant
  rt = 100000;

% Gravity constant

  grav = 0;

% Well opening. opening = 1, fully open well, opening = 0 (<0.01),
the well
% is fully closed. This variable will control what boundary
conditions that
% will apply at the outlet (both physical and numerical): We must
change
% this further below in the code if we want to change status on
this.

  wellopening = 1.0


% Specify if the primitive variables shall be found either by
% a numerical or analytical approach. If analytical = 1, analytical
% solution is used. If analytical = 0. The numerical approach is
used.
% using the itsolver subroutine where the bisection numerical method
% is used.

  analytical = 1;


% Define and intilalize flow variables



% Check area: MERK HVORDAN VI KAN FORANDRE AREALET.
% Gemoetry below is 8.5" X 5" /typical 8 1/2" section well.untitled

   for i = 1:nobox
    do(i) = 0.2159;
    di(i) = 0.127;
    areal(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    arear(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    area(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    ang(i)=3.14/2;
   end

% The code below can be activated if one wants to introduce area
changes inside
```

```
% the well.
%     for i = 12:nobox-1
%      do(i)=0.1;
%      di(i)=0.0;
%      areal(i+1) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
%      arear(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
%      end
%     arear(nobox) = arear(nobox-1);



% Now comes the intialization of the physical variables in the well.
% First primitive variables, then the conservative ones.
    for i = 1:nobox
% Here the well is intialized. We start out with a horizontal well
and
% lift this gradually up later during a 100 sec period to get the
initial
% condtions in the well prior to starting the injection of fluids at
the
% bottom. The extension letter o refers to the table represententing
the
% values at the previous timestep (old values).untitled

        % Density of liquid and gas:
        dl(i) = 1000.0;
        dg(i)= 1.0;
        %"Old" density is set equal to new density to calculate new
values
        %based on the old ones:
        dlo(i)= dl(i);
        dgo(i)=dg(i);
        % Velocity of liquid and gas at new and previous timesteps:
        vl(i) = 0.0;
        vlo(i)= 0.0;
        vg(i)= 0.0;
        vgo(i)= 0.0;
        %The pressure in the horisontal pipe is the same
        %all over:
        p(i) = 100000.0;
        po(i) = p(i);
        %Phase volume fractions of gas and liquid:
        eg(i)= 0.1;    %Gas
        ego(i)=eg(i);
        ev(i)=1-eg(i); % Liquid
        evo(i)=ev(i);

        vg(i)=0.0;
        vgo(i)=0.0;
        vl(i)=0.0;
        vlo(i)=0.0;

        % Variables related to the velocity of the flux boundaries
at old
        %and new times, and on the left and right side of the boxes
```

```
        % reflecting that area changes can take part inside cells
(i.e :
        % (A x v)left = (A x v)right, continuity equation.
        vgr(i)=0.0;
        vgor(i)= 0.0;
        vgl(i)= 0.0;
        vgol(i)= 0.0;

        vlr(i)=0.0;
        vlor(i)=0.0;
        vll(i)=0.0;
        vlol(i)=0.0;

  % Conservative variables:

       qv(i,1)=dl(i)*ev(i)*(areal(i)+arear(i))*0.5;
       qvo(i,1)=qv(i,1);

       qv(i,2)=dg(i)*eg(i)*(areal(i)+arear(i))*0.5;
       qvo(i,2)=qv(i,2);


qv(i,3)=(qv(i,1)*vl(i)+qv(i,2)*vg(i))*(areal(i)+arear(i))*0.5;
       qvo(i,3)=qv(i,3);

    end

% Intialize fluxes between the cells/boxes

for i = 1:nofluxes
  for j =1:3
   flc(i,j)=0.0; % Flux of liquid over box boundary
   fgc(i,j)=0.0; % Flux of gas over box boundary
   fp(i,j)= 0.0; % Pressure flux over box boundary
  end
end



%  Main program. Here we will progress in time. First som
intializations
% and definitions to take out results. The for loop below
runsuntitled until the
% simulation is finished.

countsteps = 0;
counter=0;
printcounter = 1;
pbot(printcounter) = p(1);
pchoke(printcounter)= p(nobox);
pcasingshoe(printcounter)=p(25);
liquidmassrateout(printcounter) = 0;
gasmassrateout(printcounter)=0;
timeplot(printcounter)=time;

for i = 1:nosteps
```

```
   countsteps=countsteps+1;
   counter=counter+1;
   time = time+dt;

% First of all a dirty trick is used in order to make the well
vertical.
% The pipe was initialised for a horisontal case. However, for a
vertical
% case we would need a steady state solver. Since the programmer in
this
% case is quite lazy, he rather chose to adjust the gravity constant
g from
% zero to 9.81 m/s2 during 100 seconds (corresponds to hoisting the
well
% from a horisontal postion to vertical case.

% If we want to simulate a horizontal well. Just comment/deactivate
the
% code below.
% untitled
%        if (time <= 100)
%          g = grav*time/100;
%        else
%          g = grav;
%        end
%

  g=grav;

% Then a section where specify the boundary conditions.
% Here we specify the inlet rates of the different phases at the
% bottom of the pipe in kg/s. We interpolate to make things smooth.
% It is also possible to change the outlet boundary status of the
well
% here. First we specify rates at the bottom and the pressure
untitledat the outlet
% in case we have an open well.

% The code below is an example of an open connection

    inletligmassrate = 25;
    inletgasmassrate = 0;

   startuptime = 0.1;
 if (time < startuptime)
%
   xint = (time-0)/startuptime;
   inletligmassrate=25*xint;
   inletgasmassrate=0.0;
 else
   inletligmassrate=25;
   inletgasmassrate=0.0;
 end
% elseif ((time>=150) & (time < 160))
%   inletligmassrate = 22*(time-150)/10;
%   inletgasmassrate = 2.0*(time-150)/10;
```

```
%
% elseif ((time >=160) & (time<1700))
%   inletligmassrate = 22;
%   inletgasmassrate = 2.0;
%
% elseif ((time>=1700)& (time<1710))
%   inletligmassrate = 22-22*(time-1700)/10;
%   inletgasmassrate = 2.0-2.0*(time-1700)/10;
% elseif ((time>=1710)&(time<2000))
%   inletligmassrate =0;
%   inletgasmassrate =0;
% elseif ((time>=2000)& (time<2010))
%   inletligmassrate= 22*(time-2000)/10;
%   inletgasmassrate= 2.0*(time-2000)/10;
% elseif (time>2010)
%   inletligmassrate= 22;
%   inletgasmassrate= 2.0;
% end


% Below, test code for onephase connection. Commented out. Friction
% model seems reasonable.
% if (time < 150)
%
%   inletligmassrate=0.0;
%   inletgasmassrate=0.0;
%
% elseif ((time>=150) & (time < 160))
%   inletligmassrate = 44*(time-150)/10;
%   inletgasmassrate = 0;
%
% elseif ((time >=160) & (time<300))
%   inletligmassrate = 44;
%   inletgasmassrate = 0;
%
% elseif ((time>=300)& (time<310))
%   inletligmassrate = 44-44*(time-300)/10;
%   inletgasmassrate = 0;
% else
%   inletligmassrate=0;
%   inletgasmassrate = 0;
% end




% specify the outlet pressure /Physical. Here we have given the
pressure as
% constant. It would be possible to adjust it during openwell
conditions
% either by giving the wanted pressure directly (in the command
lines
% above) or by finding it indirectly through a chokemodel where the
wellopening
% would be an input parameter. The wellopening variable would
equally had
```

```
% to be adjusted inside the command line structure given right
above.


% This value can be changed if we want to simulate a choke at the
outlet.
 pressureoutlet = 100000.0;

% Based on these boundary values combined with use of extrapolations
techniques
% for the remaining unknowns at the boundaries, we will define the
mass and
% momentum fluxes at the boundaries (inlet and outlet of pipe).

% inlet fluxes first.

     flc(1,1)= inletligmassrate/areal(1);
     flc(1,2)= 0.0;
     flc(1,3)= flc(1,1)*vlo(1);

     fgc(1,1)= 0.0;
     fgc(1,2)= inletgasmassrate/areal(1);
     fgc(1,3)= fgc(1,2)*vgo(1);

     fp(1,1)= 0.0;
     fp(1,2)= 0.0;
     fp(1,3)= po(1)+0.5*(po(1)-po(2)); %Interpolation used to find
the
% pressure at the inlet/bottom of the well.



% Outlet fluxes (open & closed conditions)

    if (wellopening>0.01)
        flc(nofluxes,1)= dlo(nobox)*evo(nobox)*vlo(nobox);
        flc(nofluxes,2)= 0.0;
        flc(nofluxes,3)= flc(nofluxes,1)*vlo(nobox);

        fgc(nofluxes,1)= 0.0;
        fgc(nofluxes,2)= dgo(nobox)*ego(nobox)*vgo(nobox);
        fgc(nofluxes,3)= fgc(nofluxes,2)*vgo(nobox);

        fp(nofluxes,1)= 0.0;
        fp(nofluxes,2)= 0.0;
        fp(nofluxes,3)= pressureoutlet;
    else


         flc(nofluxes,1)= 0.0;
         flc(nofluxes,2)= 0.0;
         flc(nofluxes,3)= 0.0;

         fgc(nofluxes,1)= 0.0;
         fgc(nofluxes,2)= 0.0;
         fgc(nofluxes,3)= 0.0;
```

```
        fp(nofluxes,1)=0.0;
        fp(nofluxes,2)=0.0;
        fp(nofluxes,3)= po(nobox)-0.5*(po(nobox-1)-po(nobox));

    end

% Now we will find the fluxes between the different cells.
% NB - IMPORTANE -  Note that if we change the
compressibilities/sound velocities of
% the fluids involved, we need to do changes inside the csound
function.

     for j = 2:nofluxes-1
      cl = csound(ego(j-1),po(j-1),dlo(j-1),k);
      cr = csound(ego(j),po(j),dlo(j),k);
      c = max(cl,cr);
      pll = psip(vlor(j-1),c,evo(j));
      plr = psim(vlol(j),c,evo(j-1));
      pgl = psip(vgor(j-1),c,ego(j));
      pgr = psim(vgol(j),c,ego(j-1));
      vmixr = vlol(j)*evo(j)+vgol(j)*ego(j);
      vmixl = vlor(j-1)*evo(j-1)+vgor(j-1)*ego(j-1);

      pl = pp(vmixl,c);
      pr = pm(vmixr,c);
      mll= evo(j-1)*dlo(j-1);
      mlr= evo(j)*dlo(j);
      mgl= ego(j-1)*dgo(j-1);
      mgr= ego(j)*dgo(j);

      flc(j,1)= mll*pll+mlr*plr;
      flc(j,2)= 0.0;
      flc(j,3)= mll*pll*vlor(j-1)+mlr*plr*vlol(j);

      fgc(j,1)=0.0;
      fgc(j,2)= mgl*pgl+mgr*pgr;
      fgc(j,3)= mgl*pgl*vgor(j-1)+mgr*pgr*vgol(j);

      fp(j,1)= 0.0;
      fp(j,2)= 0.0;
      fp(j,3)= pl*po(j-1)+pr*po(j);
     end

% Fluxes have now been calculated. We will now update the
conservative
% variables in each of the numerical cells.



     for j=1:nobox

%       vmixfric = vlo(j)*evo(j)+vgo(j)*ego(j); % Mixture viscosity
%       viscmix =  viscl*evo(j)+viscg*ego(j);
      densmix = dlo(j)*evo(j)+dgo(j)*ego(j);
```

```
      a2 = arear(j);
      a1 = areal(j);
      avg = (a2+a1)*0.5;

      pressure=p(j);

%     We calculate the frictional gradient by calling upon the
dpfric function.
 % friclossgrad =
dpfric(vlo(j),vgo(j),evo(j),ego(j),dlo(j),dgo(j),pressure,do(j),di(j
),viscl,viscg);
     friclossgrad = 0;


%     Here is the updating of the conservative variables/where we
solve the
%     two mass conservation equations and the third momentum
equation.
      qv(j,1)=qvo(j,1)-dtdx*((a2*flc(j+1,1)-a1*flc(j,1))...
                           +(a2*fgc(j+1,1)-a1*fgc(j,1))...
                           +(avg*fp(j+1,1)-avg*fp(j,1)));

      qv(j,2)=qvo(j,2)-dtdx*((a2*flc(j+1,2)-a1*flc(j,2))...
                           +(a2*fgc(j+1,2)-a1*fgc(j,2))...
                           +(avg*fp(j+1,2)-avg*fp(j,2)));

      qv(j,3)=qvo(j,3)-dtdx*((a2*flc(j+1,3)-a1*flc(j,3))...
                           +(a2*fgc(j+1,3)-a1*fgc(j,3))...
                           +(avg*fp(j+1,3)-avg*fp(j,3)))...
                  -dt*avg*((friclossgrad)+g*densmix);
     end

     %Simple friction model for only pipe/laminar flow:
     %(32*vmixfric*viscmix/(do(j)*do(j))+g*densmix);



% Section where we find the physical variables (pressures, densities
etc)
% from the conservative variables. Some trickes to ensure stability


     for j=1:nobox


% Remove the area from the conservative variables to find the
% the primitive variables from the conservative ones.

      qv(j,1)= qv(j,1)/(areal(j)+arear(j))*2.0;
      qv(j,2)= qv(j,2)/(areal(j)+arear(j))*2.0;

      if (qv(j,1)<0.00000001)
        qv(j,1)=0.0;
      end

      if (qv(j,2)< 0.00000001)
```

```
        qv(j,2)=0.0000001;
      end


% Below, we find the primitive variables pressure and densities
based on
% the conservative variables q1,q2. One can choose between getting
them by
% analytical or numerical solution approach specified in the
beginning of
% the program.

    if (analytical == 1)
      % Coefficients:
      a = 1/(al*al);
      b = t1-qv(j,1)-rt*qv(j,2)/(al*al);
      c = -1.0*t1*rt*qv(j,2);

      % Analytical solution:
       p(j)=(-b+sqrt(b*b-4*a*c))/(2*a);  % Pressure
       dl(j)= dstc + (p(j)-pstc)/(al*al); % Density of liquid
       dg(j) = p(j)/rt;                   % Density of gas
    else
      %Numerical Solution:
      [p(j),error]=itsolver(po(j),qv(j,1),qv(j,2)); % Pressure
      dl(j)=rholiq(p(j)); % Density of liquid
      dg(j)=rogas(p(j)); % Density of gas

      % Incase a numerical solution is not found, the program will
write out "error":
      if error > 0
         error
      end
    end


 % Find the phase volume fractions based on new conservative
variables and
 % updated densities.

      eg(j)= qv(j,2)/dg(j);
      ev(j)=1-eg(j);

%     Reset average conservative varibles in cells with area changes
inside.

      qv(j,1)=qv(j,1)*(areal(j)+arear(j))/2.0;
      qv(j,2)=qv(j,2)*(areal(j)+arear(j))/2.0;


%     The section below is used to find the primitive variables
vg,vl
%    (phase velocities) based on the updated conservative variable
q3 and
%     the slip relation.
```

```
%      Deactivated code below, old code for no slip cond & no area
change.
%      vg(j)=qv(j,3)/(dl(j)*ev(j)+dg(j)*eg(j));
%      vl(j)=vg(j);

% Part where we interpolate in the slip parameters to avoid a
% singularities when approaching one phase gas flow.
% In the transition to one-phase gas flow, we need to
% have a smooth transition to no-slip conditions.

      xint = (eg(j)-0.75)/0.25;
      k0 = k;
      s0 = s;
      if ((eg(j)>=0.75) & (eg(j)<=1.0))
        k0 =1.0*xint+k*(1-xint);
        s0 = 0.0*xint+s*(1-xint);
      end

      if (eg(j)>=0.999999)
        k1 = 1.0;
        s1 = 0.0;
      else
        k1 = (1-k0*eg(j))/(1-eg(j));
        s1 = -1.0*s0*eg(j)/(1-eg(j));
      end
 %     help1 = dl(j)*ev(j)*k1+dg(j)*eg(j)*k0;
 %     help2 = dl(j)*ev(j)*s1+dg(j)*eg(j)*s0;

 %     vmixhelp = (qv(j,3)-help2)/help1;
 %     vg(j)=k0*vmixhelp+s0;
 %     vl(j)=k1*vmixhelp+s1;
%       help1 = qv(j,3)/(dl(j)*ev(j)+dg(j)*eg(j));
%
%       vll(j)= help1/areal(j);
%       vlr(j)= help1/arear(j);
%       vgl(j)= vll(j);
%       vgr(j)= vlr(j);


 %    Below we operate with gas vg and liquid vl velcoities
specified
 %    both in the right part and left part inside a box. (since we
have
 %    area changes inside a box these can be different. vgl is gas
velocity
 %    to the left of the disconinuity. vgr is gas velocity to the
right of
 %    the discontinuity.
 %

      help1 = dl(j)*ev(j)*k1+dg(j)*eg(j)*k0;
      help2 = dl(j)*ev(j)*s1+dg(j)*eg(j)*s0;

      vmixhelpl = (qv(j,3)/areal(j)-help2)/help1;
      vgl(j)=k0*vmixhelpl+s0;
      vll(j)=k1*vmixhelpl+s1;
```

```
      vmixhelpr = (qv(j,3)/arear(j)-help2)/help1;
      vgr(j)=k0*vmixhelpr+s0;
      vlr(j)=k1*vmixhelpr+s1;

 %  Averaging velocities.

      vl(j)= 0.5*(vll(j)+vlr(j));
      vg(j)= 0.5*(vgl(j)+vgr(j));

    end



% Old values are now set equal to new values in order to prepare
% computation of next time level.
    for j = 1:nobox
     po(j)=p(j);
     dlo(j)=dl(j);  %Liquid density
     dgo(j)=dg(j);  %Gas density
     vlo(j)=vl(j);  %Liquid velocity
     vgo(j)=vg(j);  %Gas velocity
     ego(j)=eg(j);  %Gas fration
     evo(j)=ev(j);  %Liquid fraction.

     vlor(j)=vlr(j);
     vlol(j)=vll(j);
     vgor(j)=vgr(j);
     vgol(j)=vgl(j);

      for m =1:3
       qvo(j,m)=qv(j,m);
      end
    end


% Section where we save some timedependent variables in arrays.
% e.g. the bottomhole pressure. They will be saved for certain
% timeintervalls defined in the start of the program in order to
ensure
% that the arrays do not get to long!

  if (counter>=nostepsbeforesavingtimedata)
    printcounter=printcounter+1;
    time
    pbot(printcounter)= p(1);
    pchoke(printcounter)=p(nobox);
    pcasingshoe(printcounter)=p(25); %NB THIS MUST BE DEFINED IN
CORRECT BOX
%
liquidmassrateout(printcounter)=dl(nobox)*ev(nobox)*vl(nobox)*area(n
obox);
%
gasmassrateout(printcounter)=dg(nobox)*eg(nobox)*vg(nobox)*area(nobo
x);
```

```
liquidmassrateout(printcounter)=dl(nobox)*ev(nobox)*vl(nobox)*arear(
nobox);

gasmassrateout(printcounter)=dg(nobox)*eg(nobox)*vg(nobox)*arear(nob
ox);
    timeplot(printcounter)=time;
    counter = 0;


  end
end

% end of stepping forward in time.




% Printing of resultssection


countsteps % Marks number of simulation steps.


% Plot commands for variables vs time.

% Inlet trykk
%plot(timeplot,pbot/100000)
plot(x,p)
%plot(timeplot,pchoke/100000)
%plot(timeplot,pcasingshoe/100000)
%plot(timeplot,liquidmassrateout)
%plot(timeplot,gasmassrateout)
%plot(vg)

%Plot commands for variables vs depth/Only the last simulated
%values/endtime is visualised

%plot(vl,x);
%plot(vg,x);
%plot(eg,x);
%plot(p,x);
%plot(dl,x);
%plot(dg,x);
```

Appendix 2

Transient AUSMV scheme

```
% Transient two-phase code based on AUSMV scheme: Gas and Water
% The code can handle area changes. The area changes are defined
inside
% the cells such that the where the fluxes are calculated, the
geometry is
% uniform.

clear;

% Geometry data/ Must be specified
welldepth = 5000;
nobox = 100; %Number of boxes in the well
nofluxes = nobox+1;
dx = welldepth/nobox; % Boxlength
%dt = 0.005;

% Welldepth array
x(1)= -1.0*welldepth+0.5*dx;
for i=1:nobox-1
 x(i+1)=x(i)+ dx;
end

dt= 0.005  % Timestep
dtdx = dt/dx;
time = 0.0;
endtime = 66; % Time for end of simulation
nosteps = endtime/dt;  %Number of total timesteps
timebetweensavingtimedata = 1.0;  % How often in s we save data vs
time for plotting.
nostepsbeforesavingtimedata = timebetweensavingtimedata/dt;

% Slip parameters used in the gas slip relation. vg =Kvmix+S
k = 1.2;
s = 0.5;

% Viscosities (Pa*s)/Used in the frictional pressure loss model.
viscl = 0.05; % Liquid phase
viscg = 0.000005; % Gas phase

% Density parameters. These parameters are used when finding the
% primitive variables pressure, densities in an analytical manner.
% Changing parameters here, you must also change parameters inside
the
% density routines roliq and rogas.

% liquid density at stc and speed of sound in liquid
  dstc = 1000.0;   %Base density of liquid, See also roliq.
  pstc = 100000.0;
  al = 1500; % Speed of sound/compressibility of liquid phase.
  t1 = dstc-pstc/(al*al);
% Ideal gas law constant
```

```
  rt = 100000;

% Gravity constant

  grav = 0;

% Well opening. opening = 1, fully open well, opening = 0 (<0.01),
the well
% is fully closed. This variable will control what boundary
conditions that
% will apply at the outlet (both physical and numerical): We must
change
% this further below in the code if we want to change status on
this.

  wellopening = 1.0


% Specify if the primitive variables shall be found either by
% a numerical or analytical approach. If analytical = 1, analytical
% solution is used. If analytical = 0. The numerical approach is
used.
% using the itsolver subroutine where the bisection numerical method
% is used.

  analytical = 1;


% Define and intilalize flow variables



% Check area: MERK HVORDAN VI KAN FORANDRE AREALET.
% Gemoetry below is 8.5" X 5" /typical 8 1/2" section well.untitled

   for i = 1:nobox
    do(i) = 0.2159;
    di(i) = 0.127;
    areal(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    arear(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    area(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
    ang(i)=3.14/2;
   end

% The code below can be activated if one wants to introduce area
changes inside
% the well.
%     for i = 12:nobox-1
%      do(i)=0.1;
%      di(i)=0.0;
%      areal(i+1) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
%      arear(i) = 3.14/4*(do(i)*do(i)- di(i)*di(i));
%      end
%     arear(nobox) = arear(nobox-1);
```

```
% Now comes the intialization of the physical variables in the well.
% First primitive variables, then the conservative ones.
    for i = 1:nobox
% Here the well is intialized. We start out with a horizontal well
and
% lift this gradually up later during a 100 sec period to get the
initial
% condtions in the well prior to starting the injection of fluids at
the
% bottom. The extension letter o refers to the table represententing
the
% values at the previous timestep (old values).

        % Density of liquid and gas:
        dl(i) = 1000.0;
        dg(i)= 1.0;
        %"Old" density is set equal to new density to calculate new
values
        %based on the old ones:
        dlo(i)= dl(i);
        dgo(i)=dg(i);
        % Velocity of liquid and gas at new and previous timesteps:
        vl(i) = 0.0;
        vlo(i)= 0.0;
        vg(i)= 0.0;
        vgo(i)= 0.0;
        %The pressure in the horisontal pipe is the same
        %all over:
        p(i) = 100000.0;
        po(i) = p(i);
        %Phase volume fractions of gas and liquid:

        if (i<60)
         eg(i)= 0.0;
        else
         eg(i)= 0.0;
        end  %Gas
        ego(i)=eg(i);
        ev(i)=1-eg(i); % Liquid
        evo(i)=ev(i);

        vg(i)=0.0;
        vgo(i)=0.0;
        vl(i)=0.0;
        vlo(i)=0.0;

        % Variables related to the velocity of the flux boundaries
at old
        %and new times, and on the left and right side of the boxes
        % reflecting that area changes can take part inside cells
(i.e :
        % (A x v)left = (A x v)right, continuity equation.
        vgr(i)=0.0;
        vgor(i)= 0.0;
        vgl(i)= 0.0;
```

```
        vgol(i)= 0.0;

        vlr(i)=0.0;
        vlor(i)=0.0;
        vll(i)=0.0;
        vlol(i)=0.0;

  % Conservative variables:

       qv(i,1)=dl(i)*ev(i)*(areal(i)+arear(i))*0.5;
       qvo(i,1)=qv(i,1);

       qv(i,2)=dg(i)*eg(i)*(areal(i)+arear(i))*0.5;
       qvo(i,2)=qv(i,2);


qv(i,3)=(qv(i,1)*vl(i)+qv(i,2)*vg(i))*(areal(i)+arear(i))*0.5;
       qvo(i,3)=qv(i,3);

    end

% Intialize fluxes between the cells/boxes

for i = 1:nofluxes
  for j =1:3
   flc(i,j)=0.0; % Flux of liquid over box boundary
   fgc(i,j)=0.0; % Flux of gas over box boundary
   fp(i,j)= 0.0; % Pressure flux over box boundary
  end
end



%  Main program. Here we will progress in time. First som
intializations
% and definitions to take out results. The for loop below
runsuntitled until the
% simulation is finished.

countsteps = 0;
counter=0;
printcounter = 1;
pbot(printcounter) = p(1);
pchoke(printcounter)= p(nobox);
pcasingshoe(printcounter)=p(25);
liquidmassrateout(printcounter) = 0;
gasmassrateout(printcounter)=0;
timeplot(printcounter)=time;

for i = 1:nosteps
   countsteps=countsteps+1;
   counter=counter+1;
   time = time+dt;

% First of all a dirty trick is used in order to make the well
vertical.
```

```
% The pipe was initialised for a horisontal case. However, for a
vertical
% case we would need a steady state solver. Since the programmer in
this
% case is quite lazy, he rather chose to adjust the gravity constant
g from
% zero to 9.81 m/s2 during 100 seconds (corresponds to hoisting the
well
% from a horisontal postion to vertical case.

% If we want to simulate a horizontal well. Just comment/deactivate
the
% code below.
% untitled
%        if (time <= 100)
%          g = grav*time/100;
%        else
%          g = grav;
%        end
%

  g=grav;

% Then a section where specify the boundary conditions.
% Here we specify the inlet rates of the different phases at the
% bottom of the pipe in kg/s. We interpolate to make things smooth.
% It is also possible to change the outlet boundary status of the
well
% here. First we specify rates at the bottom and the pressure
untitledat the outlet
% in case we have an open well.

% The code below is an example of an open connection

    inletligmassrate = 25;
    inletgasmassrate = 0;

   startuptime = 0.1;
 if (time < startuptime)
%
   xint = (time-0)/startuptime;
   inletligmassrate=25*xint;
   inletgasmassrate=0.0;
 else
   inletligmassrate=25;
   inletgasmassrate=0.0;
 end
% elseif ((time>=150) & (time < 160))
%   inletligmassrate = 22*(time-150)/10;
%   inletgasmassrate = 2.0*(time-150)/10;
%
% elseif ((time >=160) & (time<1700))
%   inletligmassrate = 22;
%   inletgasmassrate = 2.0;
%
% elseif ((time>=1700)& (time<1710))
```

```
%   inletligmassrate = 22-22*(time-1700)/10;
%   inletgasmassrate = 2.0-2.0*(time-1700)/10;
% elseif ((time>=1710)&(time<2000))
%   inletligmassrate =0;
%   inletgasmassrate =0;
% elseif ((time>=2000)& (time<2010))
%   inletligmassrate= 22*(time-2000)/10;
%   inletgasmassrate= 2.0*(time-2000)/10;
% elseif (time>2010)
%   inletligmassrate= 22;
%   inletgasmassrate= 2.0;
% end


% Below, test code for onephase connection. Commented out. Friction
% model seems reasonable.
% if (time < 150)
%
%   inletligmassrate=0.0;
%   inletgasmassrate=0.0;
%
% elseif ((time>=150) & (time < 160))
%   inletligmassrate = 44*(time-150)/10;
%   inletgasmassrate = 0;
%
% elseif ((time >=160) & (time<300))
%   inletligmassrate = 44;
%   inletgasmassrate = 0;
%
% elseif ((time>=300)& (time<310))
%   inletligmassrate = 44-44*(time-300)/10;
%   inletgasmassrate = 0;
% else
%   inletligmassrate=0;
%   inletgasmassrate = 0;
% end




% specify the outlet pressure /Physical. Here we have given the
pressure as
% constant. It would be possible to adjust it during openwell
conditions
% either by giving the wanted pressure directly (in the command
lines
% above) or by finding it indirectly through a chokemodel where the
wellopening
% would be an input parameter. The wellopening variable would
equally had
% to be adjusted inside the command line structure given right
above.


% This value can be changed if we want to simulate a choke at the
outlet.
```

```
 pressureoutlet = 100000.0;

 if (time > 60 & time < 61)
  pressureoutlett = 100000+time/1*300000;
 elseif (time > 61)
  pressureoutlet = 400000;
 end

% Based on these boundary values combined with use of extrapolations
techniques
% for the remaining unknowns at the boundaries, we will define the
mass and
% momentum fluxes at the boundaries (inlet and outlet of pipe).

% inlet fluxes first.

     flc(1,1)= inletligmassrate/areal(1);
     flc(1,2)= 0.0;
     flc(1,3)= flc(1,1)*vlo(1);

     fgc(1,1)= 0.0;
     fgc(1,2)= inletgasmassrate/areal(1);
     fgc(1,3)= fgc(1,2)*vgo(1);

     fp(1,1)= 0.0;
     fp(1,2)= 0.0;
     fp(1,3)= po(1)+0.5*(po(1)-po(2));
%Interpolation used to find the
% pressure at the inlet/bottom of the well.



% Outlet fluxes (open & closed conditions)

    if (wellopening>0.01)
        flc(nofluxes,1)= dlo(nobox)*evo(nobox)*vlo(nobox);
        flc(nofluxes,2)= 0.0;
        flc(nofluxes,3)= flc(nofluxes,1)*vlo(nobox);

        fgc(nofluxes,1)= 0.0;
        fgc(nofluxes,2)= dgo(nobox)*ego(nobox)*vgo(nobox);
        fgc(nofluxes,3)= fgc(nofluxes,2)*vgo(nobox);

        fp(nofluxes,1)= 0.0;
        fp(nofluxes,2)= 0.0;
        fp(nofluxes,3)= pressureoutlet;
    else


         flc(nofluxes,1)= 0.0;
         flc(nofluxes,2)= 0.0;
         flc(nofluxes,3)= 0.0;

         fgc(nofluxes,1)= 0.0;
         fgc(nofluxes,2)= 0.0;
         fgc(nofluxes,3)= 0.0;
```

```
        fp(nofluxes,1)=0.0;
        fp(nofluxes,2)=0.0;
        fp(nofluxes,3)= po(nobox)-0.5*(po(nobox-1)-po(nobox));

    end

% Now we will find the fluxes between the different cells.
% NB - IMPORTANE -  Note that if we change the
compressibilities/sound velocities of
% the fluids involved, we need to do changes inside the csound
function.

     for j = 2:nofluxes-1
      cl = csound(ego(j-1),po(j-1),dlo(j-1),k);
      cr = csound(ego(j),po(j),dlo(j),k);
      c = max(cl,cr);
      pll = psip(vlor(j-1),c,evo(j));
      plr = psim(vlol(j),c,evo(j-1));
      pgl = psip(vgor(j-1),c,ego(j));
      pgr = psim(vgol(j),c,ego(j-1));
      vmixr = vlol(j)*evo(j)+vgol(j)*ego(j);
      vmixl = vlor(j-1)*evo(j-1)+vgor(j-1)*ego(j-1);

      pl = pp(vmixl,c);
      pr = pm(vmixr,c);
      mll= evo(j-1)*dlo(j-1);
      mlr= evo(j)*dlo(j);
      mgl= ego(j-1)*dgo(j-1);
      mgr= ego(j)*dgo(j);

      flc(j,1)= mll*pll+mlr*plr;
      flc(j,2)= 0.0;
      flc(j,3)= mll*pll*vlor(j-1)+mlr*plr*vlol(j);

      fgc(j,1)=0.0;
      fgc(j,2)= mgl*pgl+mgr*pgr;
      fgc(j,3)= mgl*pgl*vgor(j-1)+mgr*pgr*vgol(j);

      fp(j,1)= 0.0;
      fp(j,2)= 0.0;
      fp(j,3)= pl*po(j-1)+pr*po(j);
     end

% Fluxes have now been calculated. We will now update the
conservative
% variables in each of the numerical cells.



     for j=1:nobox

%       vmixfric = vlo(j)*evo(j)+vgo(j)*ego(j); % Mixture viscosity
%       viscmix =  viscl*evo(j)+viscg*ego(j);
      densmix = dlo(j)*evo(j)+dgo(j)*ego(j);
```

```
      a2 = arear(j);
      a1 = areal(j);
      avg = (a2+a1)*0.5;

      pressure=p(j);

%     We calculate the frictional gradient by calling upon the
dpfric function.
  friclossgrad =
dpfric(vlo(j),vgo(j),evo(j),ego(j),dlo(j),dgo(j),pressure,do(j),di(j
),viscl,viscg);
 %    friclossgrad = 0;


%     Here is the updating of the conservative variables/where we
solve the
%     two mass conservation equations and the third momentum
equation.
      qv(j,1)=qvo(j,1)-dtdx*((a2*flc(j+1,1)-a1*flc(j,1))...
                           +(a2*fgc(j+1,1)-a1*fgc(j,1))...
                           +(avg*fp(j+1,1)-avg*fp(j,1)));

      qv(j,2)=qvo(j,2)-dtdx*((a2*flc(j+1,2)-a1*flc(j,2))...
                           +(a2*fgc(j+1,2)-a1*fgc(j,2))...
                           +(avg*fp(j+1,2)-avg*fp(j,2)));

      qv(j,3)=qvo(j,3)-dtdx*((a2*flc(j+1,3)-a1*flc(j,3))...
                           +(a2*fgc(j+1,3)-a1*fgc(j,3))...
                           +(avg*fp(j+1,3)-avg*fp(j,3)))...
                  -dt*avg*((friclossgrad)+g*densmix);
     end

     %Simple friction model for only pipe/laminar flow:
     %(32*vmixfric*viscmix/(do(j)*do(j))+g*densmix);



% Section where we find the physical variables (pressures, densities
etc)
% from the conservative variables. Some trickes to ensure stability


     for j=1:nobox


% Remove the area from the conservative variables to find the
% the primitive variables from the conservative ones.

      qv(j,1)= qv(j,1)/(areal(j)+arear(j))*2.0;
      qv(j,2)= qv(j,2)/(areal(j)+arear(j))*2.0;

      if (qv(j,1)<0.00000001)
        qv(j,1)=0.0;
      end

      if (qv(j,2)< 0.00000001)
```

```
        qv(j,2)=0.0000001;
      end


% Below, we find the primitive variables pressure and densities
based on
% the conservative variables q1,q2. One can choose between getting
them by
% analytical or numerical solution approach specified in the
beginning of
% the program.

    if (analytical == 1)
      % Coefficients:
      a = 1/(al*al);
      b = t1-qv(j,1)-rt*qv(j,2)/(al*al);
      c = -1.0*t1*rt*qv(j,2);

      % Analytical solution:
       p(j)=(-b+sqrt(b*b-4*a*c))/(2*a);  % Pressure
       dl(j)= dstc + (p(j)-pstc)/(al*al); % Density of liquid
       dg(j) = p(j)/rt;                   % Density of gas
    else
      %Numerical Solution:
      [p(j),error]=itsolver(po(j),qv(j,1),qv(j,2)); % Pressure
      dl(j)=rholiq(p(j)); % Density of liquid
      dg(j)=rogas(p(j)); % Density of gas

      % Incase a numerical solution is not found, the program will
write out "error":
      if error > 0
         error
      end
    end


 % Find the phase volume fractions based on new conservative
variables and
 % updated densities.

      eg(j)= qv(j,2)/dg(j);
      ev(j)=1-eg(j);

%     Reset average conservative varibles in cells with area changes
inside.

      qv(j,1)=qv(j,1)*(areal(j)+arear(j))/2.0;
      qv(j,2)=qv(j,2)*(areal(j)+arear(j))/2.0;


%     The section below is used to find the primitive variables
vg,vl
%    (phase velocities) based on the updated conservative variable
q3 and
%     the slip relation.
```

```
%      Deactivated code below, old code for no slip cond & no area
change.
%      vg(j)=qv(j,3)/(dl(j)*ev(j)+dg(j)*eg(j));
%      vl(j)=vg(j);

% Part where we interpolate in the slip parameters to avoid a
% singularities when approaching one phase gas flow.
% In the transition to one-phase gas flow, we need to
% have a smooth transition to no-slip conditions.

      xint = (eg(j)-0.75)/0.25;
      k0 = k;
      s0 = s;
      if ((eg(j)>=0.75) & (eg(j)<=1.0))
        k0 =1.0*xint+k*(1-xint);
        s0 = 0.0*xint+s*(1-xint);
      end

      if (eg(j)>=0.999999)
        k1 = 1.0;
        s1 = 0.0;
      else
        k1 = (1-k0*eg(j))/(1-eg(j));
        s1 = -1.0*s0*eg(j)/(1-eg(j));
      end
 %     help1 = dl(j)*ev(j)*k1+dg(j)*eg(j)*k0;
 %     help2 = dl(j)*ev(j)*s1+dg(j)*eg(j)*s0;

 %     vmixhelp = (qv(j,3)-help2)/help1;
 %     vg(j)=k0*vmixhelp+s0;
 %     vl(j)=k1*vmixhelp+s1;
%       help1 = qv(j,3)/(dl(j)*ev(j)+dg(j)*eg(j));
%
%       vll(j)= help1/areal(j);
%       vlr(j)= help1/arear(j);
%       vgl(j)= vll(j);
%       vgr(j)= vlr(j);


 %    Below we operate with gas vg and liquid vl velcoities
specified
 %    both in the right part and left part inside a box. (since we
have
 %    area changes inside a box these can be different. vgl is gas
velocity
 %    to the left of the disconinuity. vgr is gas velocity to the
right of
 %    the discontinuity.
 %

      help1 = dl(j)*ev(j)*k1+dg(j)*eg(j)*k0;
      help2 = dl(j)*ev(j)*s1+dg(j)*eg(j)*s0;

      vmixhelpl = (qv(j,3)/areal(j)-help2)/help1;
      vgl(j)=k0*vmixhelpl+s0;
      vll(j)=k1*vmixhelpl+s1;
```

```
      vmixhelpr = (qv(j,3)/arear(j)-help2)/help1;
      vgr(j)=k0*vmixhelpr+s0;
      vlr(j)=k1*vmixhelpr+s1;

 %  Averaging velocities.

      vl(j)= 0.5*(vll(j)+vlr(j));
      vg(j)= 0.5*(vgl(j)+vgr(j));

    end



% Old values are now set equal to new values in order to prepare
% computation of next time level.
    for j = 1:nobox
     po(j)=p(j);
     dlo(j)=dl(j);  %Liquid density
     dgo(j)=dg(j);  %Gas density
     vlo(j)=vl(j);  %Liquid velocity
     vgo(j)=vg(j);  %Gas velocity
     ego(j)=eg(j);  %Gas fration
     evo(j)=ev(j);  %Liquid fraction.

     vlor(j)=vlr(j);
     vlol(j)=vll(j);
     vgor(j)=vgr(j);
     vgol(j)=vgl(j);

      for m =1:3
       qvo(j,m)=qv(j,m);
      end
    end


% Section where we save some timedependent variables in arrays.
% e.g. the bottomhole pressure. They will be saved for certain
% timeintervalls defined in the start of the program in order to
ensure
% that the arrays do not get to long!

  if (counter>=nostepsbeforesavingtimedata)
    printcounter=printcounter+1;
    time
    pbot(printcounter)= p(1);
    pchoke(printcounter)=p(nobox);
    pcasingshoe(printcounter)=p(25); %NB THIS MUST BE DEFINED IN
CORRECT BOX
%
liquidmassrateout(printcounter)=dl(nobox)*ev(nobox)*vl(nobox)*area(n
obox);
%
gasmassrateout(printcounter)=dg(nobox)*eg(nobox)*vg(nobox)*area(nobo
x);
```

```
liquidmassrateout(printcounter)=dl(nobox)*ev(nobox)*vl(nobox)*arear(
nobox);

gasmassrateout(printcounter)=dg(nobox)*eg(nobox)*vg(nobox)*arear(nob
ox);
    timeplot(printcounter)=time;
    counter = 0;


  end
end

% end of stepping forward in time.



% Printing of resultssection


countsteps % Marks number of simulation steps.


% Plot commands for variables vs time.

% Inlet trykk
%plot(timeplot,pbot/100000)
plot(x,p)
%plot(timeplot,pchoke/100000)
%plot(timeplot,pcasingshoe/100000)
%plot(timeplot,liquidmassrateout)
%plot(timeplot,gasmassrateout)
%plot(vg)

%Plot commands for variables vs depth/Only the last simulated
%values/endtime is visualised

%plot(vl,x);
%plot(vg,x);
%plot(eg,x);
%plot(p,x);
%plot(dl,x);
%plot(dg,x);
```