NTNU

Norwegian University of
Science and Technology

Modelling of Controlled Wax Deposition in Oil Dominated Subsea Production Systems

Akul Viswanathan

Master of Science in Mechanical Engineering
Submission date: January 2016
Supervisor: Even Solbraa, EPT

Norwegian University of Science and Technology
Department of Energy and Process Engineering

NTNU

Norwegian University
of Science and Technology

Department of Energy
and Process Engineering

EPT-M-2015

MASTER THESIS

for

Student Akul Viswanathan

Spring 2015

Modelling of Controlled Wax Deposition in Oil Dominated Subsea Production Systems

Modellering av kontrollert voksavsetning oljedominerte subsea produksjonssystemer

Background and objective:
A large part of the worlds remaining oil and gas resources are found in harsh environments such as deep water and arctic conditions. The development of such oil and gas fields require advanced process solutions for hydrate control, separation, dew point control and transport solutions.

Most reservoirs fluids contain heavy paraffinic components that may precipitate as a solid-like material called wax if the fluid is cooled. Wax precipitation may cause operational problems when unprocessed well streams are transported in undersea pipelines.

During the last decade a number of companies have initiated work on methods for controlled wax deposition and loosening. Such methods could potentially make possible or improve production of reservoirs with high wax content, and more particularly enable long distance transport of oils with wax.

The focus of this work will be on modelling of wax deposition from oil dominated well streams. Mathematical wax deposition models will be utilised in a MATLAB program, which is to be developed. Additionally, the computer program NeqSim will be integrated with the MATLAB program to calculate physical properties.

The following tasks are to be considered:
1. Introduction to wax deposition and precipitation
2. Proposal for unit enabling subsea cold flow transport of single phase oil dominated liquid
3. Review of deposition models
4. Development of wax deposition point-model and comparison with experimental data
5. Development of wax deposition program for unit enabling subsea cold flow transport of single phase oil dominated liquid

-- ” --

Within 14 days of receiving the written text on the master thesis, the candidate shall submit a research plan for his project to the department.

When the thesis is evaluated, emphasis is put on processing of the results, and that they are presented in tabular and/or graphic form in a clear manner, and that they are analyzed carefully.

The thesis should be formulated as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents etc. During the preparation of the text, the candidate should make an effort to produce a well-structured and easily readable report. In order to ease the evaluation of the thesis, it is important that the cross-references are correct. In the making of the report, strong emphasis should be placed on both a thorough discussion of the results and an orderly presentation.

The candidate is requested to initiate and keep close contact with his/her academic supervisor(s) throughout the working period. The candidate must follow the rules and regulations of NTNU as well as passive directions given by the Department of Energy and Process Engineering.

Risk assessment of the candidate's work shall be carried out according to the department's procedures. The risk assessment must be documented and included as part of the final report. Events related to the candidate's work adversely affecting the health, safety or security, must be documented and included as part of the final report. If the documentation on risk assessment represents a large number of pages, the full version is to be submitted electronically to the supervisor and an excerpt is included in the report.

Pursuant to "Regulations concerning the supplementary provisions to the technology study program/Master of Science" at NTNU §20, the Department reserves the permission to utilize all the results and data for teaching and research purposes as well as in future publications.

The final report is to be submitted digitally in DAIM. An executive summary of the thesis including title, student's name, supervisor's name, year, department name, and NTNU's logo and name, shall be submitted to the department as a separate pdf file. Based on an agreement with the supervisor, the final report and other material and documents may be given to the supervisor in digital format.

☐ Work to be done in lab (Water power lab, Fluids engineering lab, Thermal engineering lab)
☐ Field work

Department of Energy and Process Engineering, 11. November 2015

Olav Bolland
Department Head

Even Solbraa
Academic Supervisor

Research Advisors:
Sigbjørn Daasevatn, Subsea7
Shuxin Hou, Statoil

Abstract

The Arctic region is one of the remaining unexplored areas where large discoveries of petroleum still can be made. Large portions of the world's remaining worldwide undiscovered conventional hydrocarbon resources are foreseen to be located in this region. Deep water or ultra-deep water developments are also relevant in today's market. For this reason, the oil and gas industry is currently moving production into deeper waters, more remote areas and to colder environments without much infrastructure in place. In order to provide solutions to the market, Subsea 7 is currently developing proposals for solutions to transport crude oil at ambient seawater temperature, also known as cold flow technology. This can be introduced to improve the economy of field developments.

Wax deposition in subsea pipelines is a major flow assurance challenge, especially in the previously mentioned areas and scenarios. This thesis reviews the problem related to wax precipitation and deposition in subsea oil pipelines, and how this problem can be solved. Proposals are made for a unit chosen to be called the controlled wax deposition unit, which is a subsea concentric heat exchanger where wax is precipitated. The product fluid flows in the inner pipe, whereas the seawater is pumped through the annulus. This unit yields a cold flow solution for the crude oil, i.e. the product fluid is transported from the wellhead to the unit and then transported at ambient seawater temperature to topside. Most of the wax will precipitate and some deposit in the unit due to the temperature change. The deposited wax can be removed with a pig that goes in a loop in the unit, a pigging loop. Heat and mass transfer equations are discussed and used to create a program, which is written in MATLAB, calculating various values, e.g. wax thickness and pressure drop. After tuning, the Matzain and the Heat Analogy models are the wax models that coincide best with the experimental data. The MATLAB program yields a maximum wax thickness of 1.3 mm in the deposition unit, after seven days of operation. According to the simulation result: A good pigging frequency will be about 7 days for a 6 km long pipeline, and about 2-3 days for a 4 km long pipeline.

Sammendrag (Summary in Norwegian)

Den arktiske regionen er et av de store områdene i verden hvor det er et høyt potensial for olje og gass forekomster. Store deler av den gjenværende olje og gassen er tenkt å være i denne regionen. Brønner lokalisert i dype farvann er også relevant for dagens marked. Av disse grunnene så beveger mange selskaper dypere, til mere isolerte steder og steder med kaldere klima uten mye infrastruktur tilstede. Subsea 7 holder for tiden på å utvikle forslag til løsninger for å transportere råolje ved omgivelses temperatur, dette er kjent som cold flow teknologi. Denne teknologien kan bli implementert for å minke kostnader for et felt.

Voksavsetting i undervanns oljerør er et stort problem, spesielt i de tidligere nevnte områdene. Denne masteroppgaven gjennomgår problemene relatert til voksfelling og avsettelse i undervanns oljerør og hvordan dette problemet kan løses. En varmevekslerenhet som består av to rør, ett med større diameter enn det andre, foreslås i denne oppgaven. Oljestrømmen strømmer i det innerste røret og sjøvannet pumpes gjennom ytterrøret. Denne enheten muliggjør en oljestrømmen ved omgivelses temperatur for resten av transport strekningen. Grunnet at mesteparten av voksen vil ha felt ut i enheten. Voksen som har blitt avsatt i enheten kan fjernes med en pig som går i en loop. Varme og massetransport formler er diskutert og brukt til å lage et program i MATLAB som kalkulerer ting som: vokstykkelse og trykktap. Etter tilpasning er det Matzain og Heat Analogy modellen som passer best med eksperintell data. MATLAB koden gir en maksimal vokstykkelse på 1.3 mm i enheten, etter syv dager med gjennomstrømning. En god piggefrekvens vil være rundt syv dager for en rørlengde på 6 km, og omtrent 2-3 dager for en rørlengde på 4 km.

Preface

Subsea processing can reduce costs and enable production in places which where previously not possible. Subsea 7 is exploring possible solutions to tackle various flow assurance challenges related to offshore oil production. This master thesis focuses on the wax deposition problem in oil dominated subsea production. The work was carried out at The Department of Energy and Process Engineering at NTNU in Trondheim, with the cooperation of Subsea 7. The master thesis is weighted 30 ECTS credits and is to be written within a time period of 20 weeks.

Akul Viswanathan
Trondheim, January 18th 2015

Acknowledgement

I would like to thank my supervisors Even Solbraa and Sigbjørn Daasvatn for their guidance and help.

This thesis was written in cooperation with Subsea 7. I would, therefore, like to thank everyone at Subsea 7 who has helped me, one of which is Yuri V. Novoseltsev. I hope he will find my work interesting and relevant for his PhD.

Additionally I would like to thank Shuxin Hou, from Statoil, for his great help.

Finally I would like to thank my fellow classmates and my family for their love and support.

A.V.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

CRA Corrosion Resistant Alloy

CWDU Controlled Wax Deposition Unit

NTNU Norwegian University of Science and Technology

PIP Pipe-in-Pipe

WAT Wax Appearance Temperature

WDT Wax Dissolution Temperature

Greek Symbols

ΔT_r	Temperature difference between the hot and cold fluid, radial	K
ΔT_{ax}	Temperature difference for same fluid, step in axial direction	K
δ	Wax thickness deposited	m
$\dot{\gamma}$	Shear rate	1/s
η_{pig}	Pig wax removal efficiency	[–]
μ	Dynamic viscosity	$\mathrm{Pa} \cdot \mathrm{s}$
ν	Kinematic viscosity	m^2 /s
ϕ	Porosity of wax deposit, i.e oil fraction in deposit	[–]
π	Pi	[–]
ρ	Density	$\mathrm{kg/m}^3$
ε	Roughness height	m

Latin Symbols

$\dot{m}$	Mass flow rate	kg/s
A_c	Cross-sectional area of flow	m^2
A_s	Area of shell/wall	m^2

C	Concentration fraction of dissolved wax in fluid	[–]
D_h	Hydraulic diameter	m
D_w	Inner pipe diameter subtracted wax layer	m
D_{AB}	Binary mass diffusivity	m^2 /s
D_{ip}	Inner pipe diameter (clean)	m
D_{wo}	Diffusion coefficient of wax in oil	m^2/s
f_D	Darcy-Weisbach friction factor	[–]
F_{wax}	Wax mass fraction of deposit	[–]
h	Heat transfer coefficient	W/m^2 K
J_A	Mass flux	kg/sm^2
k	Thermal conductivity of material	$W \cdot K/m$
K_m	Mass transfer coefficient	m/s
L	Length	m
Nu	Nussuelt number	[–]
P	Wetted perimeter	m
Pr	Prandtl number	[–]
q_r	Heat transfer in radial direction	W
q_{ax}	Heat transfer with respect to temperature difference in axial direction	W
Re	Reynolds number	[–]
Sc	Schmidt number	[–]
Sh	Sherwood number	[–]
T	Temperature	K
t	Time	s
T_{c1}	Temperature of cold fluid inlet i.e. seawater inlet	K
T_{c2}	Temperature of cold fluid outlet i.e. seawater outlet	K
T_{h1}	Temperature of hot product outlet	K
T_{h2}	Temperature of hot product inlet	K
U	Overall heat transfer coefficient	W/m^2K
V	Molar volume	cm^3/mol

v	Velocity	m/s
w_w	Dissolved wax mass fraction $w_w = \rho_{wax}/\rho$	[–]

1. Introduction

1.1 Background

A large part of the worlds remaining oil and gas resources are found in harsh environments such as deep water, ultra-deep water and Arctic conditions. The development of such oil and gas fields require advanced process solutions for wax control, hydrate control, separation, dew point control and transport solutions.

Most reservoir fluids contain heavy paraffinic components that may precipitate as a solid-like material, called wax, if the fluid is cooled down. Wax precipitation may cause operational problems when unprocessed well streams are transported in subsea pipelines.

During the last decade various companies have initiated work on methods for controlled wax deposition and loosening. Such methods can potentially enable or improve production of reservoirs with high wax content, and more particularly enable long distance transport of oils containing paraffin wax. In order to provide solutions to the market, Subsea 7 currently develops proposals for new cold flow technologies that can be introduced to improve the economy of such field developments.

This master thesis builds on work done as a precursor to the master thesis, called a project thesis. Proposals are made for a subsea concentric heat exchanger unit where controlled wax deposition will take place, with cleaning done by pigging. Additionally reviews of various wax deposition models are made, and program simulating the temperature drop and wax deposition in the heat exchanger unit are made.

Work on this subject has earlier been done at the Department of Energy and Process Engineering, NTNU. More specifically Emmanuel Oluwatosin Ajayi[1] has written a master thesis with a similar topic.

1.2 Objective

The main objective of this master thesis is to develop a wax deposition program in MATLAB. Where NeqSim is incorporated into the program. It can be used for simulating the deposition process in the proposed subsea controlled wax deposition unit, for a oil dominated well stream. A comparison of the experimental data and the mathematical wax deposition models are to be done. The comparison will give an insight as to how

accurate the model is. There are several uncertainties surrounding field data, fluid composition etc. Hence, a new user should be able to use and alter the program. With the simulation data and program Subsea 7 should be able to get more insight as to how to further develop the controlled wax deposition unit. Additionally this thesis gives a good overview of the challenges related to cold flow, and a good overview of relevant mathematical equations.

1.3 Report Structure

The problem description, attached in the first couple of pages, divides the thesis into 5 main subsections:

1. Introduction to wax deposition and precipitation
2. Proposal for unit enabling subsea cold flow transport of single phase oil dominated liquid
3. Review of deposition models
4. Development of wax deposition point-model and comparison with experimental data
5. Development of wax deposition program for unit enabling subsea cold flow transport of single phase oil dominated liquid

This thesis is comprised of 9 main chapters, in addition to Introduction, Conclusion and Further Work.

Chapter 2 and 3 is an introduction to wax precipitation, deposition and wax deposition handling. Chapter 4 discusses a proposal for a subsea controlled wax deposition unit, where most of the wax content is precipitated. This yields a possibility to transport the crude oil with minimal wax deposition in the rest of the pipeline. The proposed unit will be further examined throughout the rest of the thesis. Chapter 5 and 6 introduces and discusses various mathematical approaches to calculate heat transfer, mass transfer and wax deposition models. Chapter 7 explains the structure of the MATLAB program that has been created. Chapter 8, 9 and 10 presents the values and findings gathered from the MATLAB program simulations. Chapter 9 additionally contains experimental data. Appendix A contains some of the scripts for the MATLAB program.

The respective subsections are answered in the following chapters:

- Subseciton 1. is answered in Chapter 2 and 3
- Subsection 2. is answered in Chapter 4 and 10
- Subsection 3. is answered in Chapter 5 and 6
- Subsection 4. is answered in Chapter 7, 9 and Appendix A
- Subsection 5. is answered in Chapter 7, 10 and Appendix A

2. Paraffin Wax and Hydrate formation

Paraffin wax is a term used for a mixture of long chained alkane hydrocarbons, n-paraffins, with carbon chain lengths ranging from $C15$ to $C75$[2]. There are also iso-paraffins which are branched molecules, these are, however, usually unstable as wax solids. The general formula is C_nH_{2n}. Paraffin wax is often found in oil and consists of flexible hydrocarbon molecules, which tend to cluster together[3].

Subsea transport of unprocessed multiphase flow at ambient seawater temperature causes a risk of hydrate and wax formation. Hydrate is an ice like solid that can cause blockages in pipelines when water and natural gas combine at high pressure and low temperature. It may be stable up to temperatures of $30\,^{\circ}C$, depending on pressure and the composition of the hydrocarbon. It is common to inject methanol or ethylene glycol, MEG, as freezing point depressant chemicals for hydration control. The problem with using this method is that it is costly, needed in great quantity and one may need additional infrastructure[4].

A possible solution to this problem is the SINTEF developed concept CONversion of Water to Hydrate Particles. This solution takes the free water and converts it into free flowing hydrate particles, and transports it as a hydrate slurry in a cold flowline. This will solve the hydrate deposition and plugging problem as well as lower the risk of corrosion. As hydrate formation is not the focus of this thesis, it will not be further discussed[5].

2.1 Wax Appearance Temperature (WAT) and Pour Point

Temperature is the dominating factor of paraffin solubility. As the liquid mixture is cooled the paraffin components become less soluble. At a certain temperature the highest molecular weighted paraffin precipitates, and the others follow as the temperature is further reduced. The temperature, where the highest molecular weighted paraffin precipitates, is called the wax appearance temperature (WAT), also known as cloud point and paraffin crystallization temperature. The cloud point can be found by studying a liquid sample, and simply cooling the sample until wax appears. The name is given because the sample often becomes opaque at this temperature. TUWAX is a thermodynamic modelling software that defines the WAT as the temperature and pressure at which 0.02 mole per cent of the crude precipitates out in a solid state[2, 3, 6].

Pour point, a parameter used in the industry, is the lowest temperature at which an oil will flow freely under its own weight at specific testing conditions. An example of

a situation where this value may be of interest is during shut-down. The temperature may drop below the pour point and cause complications during start-up[7].

2.2 Wax Deposition

Deposition on pipeline walls can be a big problem in subsea pipelines. The seawater temperature is often below the WAT, causing the pipeline walls to cool the oil and in turn deposit wax on the pipeline wall. This causes a plethora of problems, a worst-case scenario is a total pipeline blockage that can result in a complete shut-down, causing great economic consequences. Wax usually has a greater surface roughness than the pipeline wall and deposited wax reduces the diameter, causing there to be a greater pressure drop. This may result in less throughput minimizing production quantities, and increasing operation pressure. The increase of operation pressure brings about more consumption of power and reduces the safety of the pipeline[8].

2.3 Determining Wax Content

It is of great importance in the petroleum industry to know what the wax content of the crude oil is. This is especially important for production, storage and transportation of waxy crude oils. There are various methods one can use to determine the wax content. Some of these methods are: The standard acetone method and a modified version, which is the method most used in the industry, gas chromatography, pulsed nuclear magnetic resonance (NMR) and density measurement techniques. The acetone method is complex and uses toxins such as toluene and benzene, the gas chromatographic method and pulsed NMR method have poor accuracy and the density measurement technique requires expensive specialised equipment. There has, therefore, been research on more convenient and reliable methods to determine the wax content of crude oils, such a method is the differential scanning calorimetry (DSC), which is a well documented tool to find the characteristics of crude oils[9].

2.4 WAT Measurements

There are several WAT measurement options e.g. visual observation, cross-polarized microscopy, filter plugging, rheometry, differential scanning calorimetry (DSC), densitometry, and spectroscopic methods. The knowledge of the WAT and where it might take place is critical for flow assurance strategies related to petroleum production. It is common to use multiple experimental techniques to confirm the accuracy of the measured WAT[10].

2.5 Subsea Processing

Subsea processing, as the name implies, involves processing non-processed oil subsea. Currently it is more common to transport the product to topside for processing, e.g. to an offshore oil rig. Subsea processing can be beneficial where distances from a platform

to satellite wells are large. This could be because it is not economically feasible to have a platform near each well located far from each other, due to ultra deepwater or due to frigid Arctic waters making long transport of unprocessed oil difficult. High pressure due to the depths makes equipment that is meant to be placed on the seafloor costly and difficult to design. However, by placing equipment on the seabed rather than on a floating platform one has the potential to make oil and gas production substantially cheaper. It can pave the way for greater production and higher cost-efficiency in the offshore oil and gas industry.

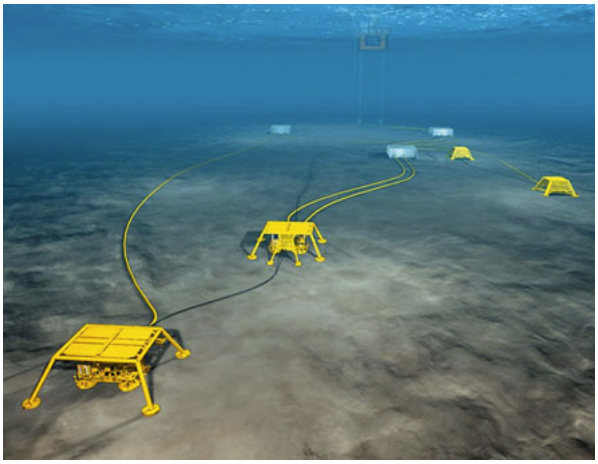


Figure 2.1: The world's first commercial subsea system provided by FMC for the Statoil's Tordis field. Source, offshore-technology.com

By processing subsea it is possible to transport the product great distances without the worry of problems related to multiphase flow, wax deposition and hydrate formation. This will allow the product to be transported long distances at ambient temperature, cold flow. See Section 3.7 for more on cold flow. Seperated gas and water can be re-injected back into the reservoir, and boost production by maintaining reservoir pressure. Placing a re-injection system subsea reduces the infrastructure needed, compared to if the separation takes place on a platform where it needs to be sent back to the well for re-injection. It also allows for a reduction in pipeline diameter, while still transporting the same amount of oil, as the fluid to be re-injected is removed close to the well reducing the water cut.

3. Solutions for Handling Wax Deposition

Most commonly the WAT is higher than the seawater temperature, ambient temperature, leading to precipitation and deposition of wax on the pipeline walls. One of the best ways to avoid the problem of wax deposition is to prevent the precipitation of wax taking place in the first place. Extensive research is being done on chemical injection, to avoid wax precipitation. An example of this is the chemical DS-1607, which is being used 140 kilometres of the coast of Congo. This is an acrylate-based polymer which prevents the formation of wax in the pipelines of this offshore installation[11].

Thermal insulation and heating of pipelines to avoid wax precipitation, or deliberate precipitation of wax are also possible solutions to tackle the problem. However, these solutions might add complexity, not be fully efficient, logistically impractical or not cost effective. Different types of methods for handling wax deposition is further discussed in this chapter. Several of these methods can also be used to avoid the formation of hydrates.

3.1 Pigging

Pigging, which is widely used in the oil and gas industry, is a mechanical solution for maintenance of pipelines. Pipeline pigs can be introduced to the pipeline via a pig launcher and retrieved via a pig trap. Launching a pig near wellheads located several kilometres from human activity can be time consuming and costly. To avoid this an automated pig launching system can be used, e.g. a magazine filled with several pigs that can be deployed remotely.

The pig does not interrupt the product flow, as it is propelled by the flow and the pressure difference on the front and back side of the pig. Pigs are usually cylindrical in shape and can clean pipeline walls by scraping the sides and pushing the debris ahead of it. This is ideal for clearing deposited wax from pipeline walls.

Figure 3.1: Pipeline segment with a cut-out and a pig inside. Source, Flickr.com by Harvey Barrison

There are various tasks the pigs can perform, in addition to cleaning. There is, therefore, a vast variety of pigs intended for different uses. Utility pigs are used for cleaning the pipeline. Sealing pigs are used to separate two different products within a pipeline. Inspection pigs often also known as smart pigs can gather information like; Pipeline diameter, curvature, bends, temperature and pressure, as well as corrosion or metal loss. Speciality pigs, such as plugs, can be used to isolate a section of a pipeline for maintenance work. The deployment of pigs can cause blockages if a pig gets stuck. Some pig types can have a more detrimental effect on the pipeline than others[12].

3.2 Chemical Solution

A way to avoid wax precipitation is to use chemicals, often referred to as wax inhibitors. They work by bonding to the wax crystals and stops further growth, hence, the chemicals need to be added before the crystallisation takes place. The chemicals need to be fine-tuned to each well composition to be effective. This makes it difficult when a number of well streams are combined, which is not uncommon in offshore production. Toluene and xylene, which are aromatic hydrocarbons, are often used to remove wax that has already deposited, but it can also be used as a precautionary measure. There are also dispersants, which acts like soap does on grease and water, where one end attaches to the paraffin and the other to oil or water. This prevents the agglomeration of the wax.

There are great costs related to chemical injection due to the large amounts needed, cost of the chemicals and the additional infrastructure needed. There are also strict safety regulations to prevent chemicals, which may cause severe environmental problems, from contaminating the environment[13].

3.3 Pipeline Burial

Pipeline burial is a method that can be used to avoid that the production fluid, pre-process oil, gets below its WAT. The mass covering the pipeline acts as a thermal

insulator, additionally geothermal heat can help lower the heat losses to the environment. The conductivity of the soil above the pipeline and the burial depth are important parameters for calculating the heat losses. The greater the depth one chooses to bury the pipeline the greater the cost[13].

3.4 Insulation and Pipe in Pipe

As for the pipeline burial solution, insulation keeps the crude oil above the WAT to avoid wax precipitation. With this solution there is more flexibility because the need for burial is negated. Polymer materials are often used in subsea pipelines, these are good because they are tough and do not corrode. Polyurethane and Polypropylene foams used for these coatings have thermal conductivities as low as 0.16 [W/mK].

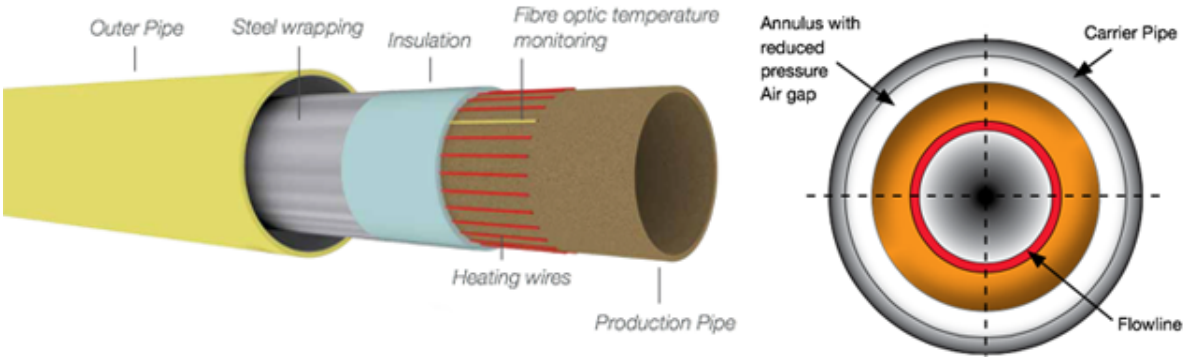


Figure 3.2: Insulated PIP. Source, subsea7.com Deep 7

Another solution is a pipe-in-pipe (PIP) solution, which is as the name implies two concentric pipes one larger than the other, see right-hand side figure in Figure 3.2. In the annular space there can be a vacuum or an insulation material. With this method the insulation material does not need to be able to withstand the harsh offshore conditions, as it is not exposed to the environment. Fibreglass wool is an example of an insulation material that can be used, it can have an extremely low thermal conductivity of about 0.02 [W/mK][13].

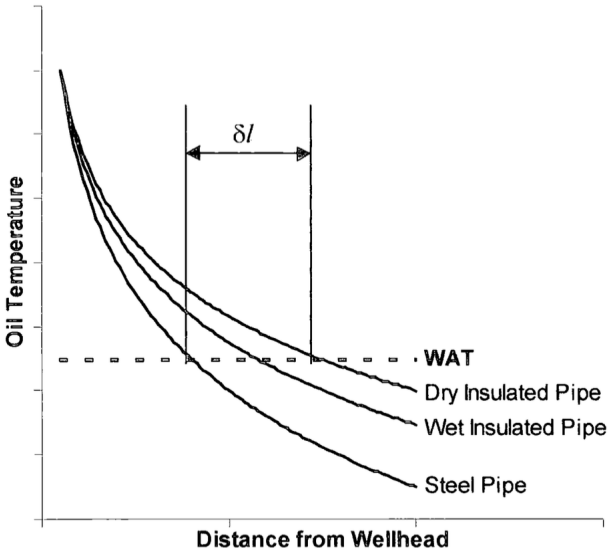


Figure 3.3: Difference in distance before reaching WAT[13]

Insulated pipelines costs a lot more than its non-insulated counterpart, and the WAT is merely moved farther downstream. This can be seen in Figure 3.3, which illustrates the increase in transport length, for various pipes, before reaching WAT. The dry insulated pipe is a PIP pipeline, and the wet insulated pipe is a non-PIP insulated pipeline. As long as the WAT is not reached in the pipeline there should not be a wax precipitation problem, however, there will always be heat losses and this creates a maximum pipeline length before the WAT is reached.

3.5 Heating

To avoid reaching WAT in longer pipelines it is possible to electrically heat it. As shown in Figure 3.4 the electrical heating unit is often located in the annulus in a PIP pipeline. It is also possible to replace the electrical heating unit with a warm fluid flowing through the annulus, which in turn heats the product.

These two methods have several advantages over passive heating solutions. During start-up the cool pipeline takes time to heat, and wax may get deposited on to the pipeline walls. This can be avoided with a heated system by warming the pipelines prior to start-up. Heating helps add more control to the system, e.g. during seasonal temperature changes one can easily regulate the heating needed. There is also the added advantage of the ability to melt deposited wax if there has been a shut-down. However, this system will add great complexity and capital cost during installation and operation.

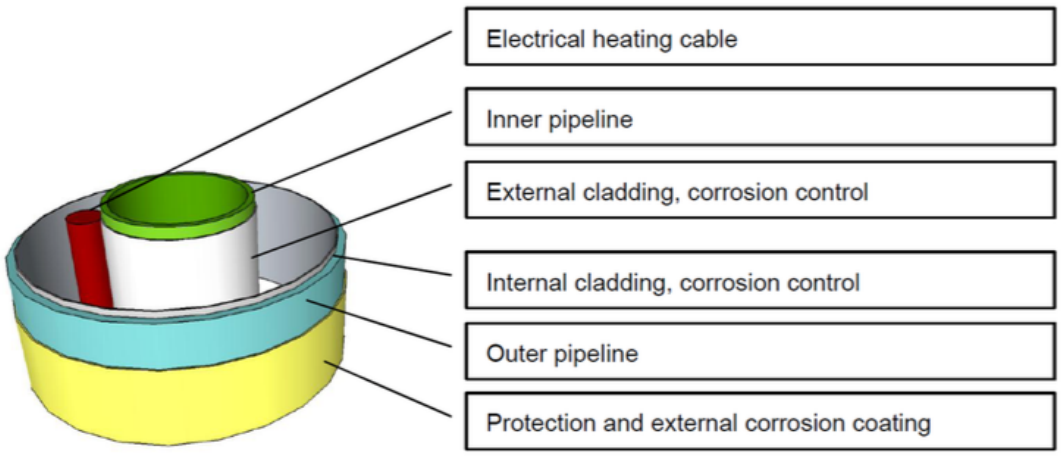


Figure 3.4: Example of a direct electrically heated pipeline[16]

3.6 Surface Coating

As the Teflon non-stick layer on a frying pan prevents food from sticking, a layer of non-stick coating can be added to the pipeline wall to prevent wax from sticking to the wall. A team inspired by the superoleophobic properties of fish scales, which can keep their bodies clean in oil-polluted waters. Have invented a non-wax-stick coating comprised of zinc, oxygen and silicon elements, with proven effects of reduction in deposited wax[14]. The downside to surface coating is that there is no known coating that stops wax from forming completely. This means that there could be a need for an alternative wax removal solution. There is also the possibility of the coating wearing out, e.g. by pigging or the fluid itself, or scratches forming exposing the non-oleophobic pipeline wall.

3.7 Cold Flow Solution

Cold flow is a solution where oil is transported at the ambient seawater temperature. Many of the solutions mentioned above keeps the product fluid hot, i.e. above WAT. This poses challenges when transporting oil long distances or in frigid waters such as in the Arctic. To make cold flow a feasible solution one needs to tackle the problem of wax deposition. There are several possible methods to make this work, some of which are presented in this section.

3.7.1 Statoil's Heat Pulse Technology

As is described in a Subsea 7 internal document: At Statoil's test facilities in Porsgrunn, Norway, experiments were performed with waxy condensate circulating at constant temperature in a test rig. Cooling water, which was in an annulus, surrounded the pipe with the condensate inside. The cooling water temperature was first chosen to be $10\,°C$, where continuous wax deposition in the pipe was observed. Later on the cooling water was

set to be $15\,^{\circ}C$, this reduced the temperature difference between the condensate and the cooling water. The wax build up was observed to be slower. Finally the the water temperature was increased to $20\,^{\circ}C$, eliminating the temperature difference. One day later the deposited wax on the pipeline walls were unexpectedly completely gone. It is important to note that the wax was not completely melted when released.

The explanation of this is that when the pipeline walls are warmed the wax layer's structure, towards the pipeline wall, changes causing the wax to more easily slip off the wall. The structural change decreases the ability to transfer shear force from the fluid through the wax layer and to the wall's surface. Once the shear forces are larger than the adhesive forces of the wax, the precipitated wax solids slip from the wall's surface. This phenomenon was the reason the test rig at Porsgrunn was cleared of wax. This method of cleaning wax has the potential of being used on a larger scale in offshore oil production. The technology has been patented by Statoil[15].

3.7.2 Wax Seeding

Wax seeding is a method where seed particles are introduced to the flow. The introduction of these particles causes nucleation to take place at a higher temperature, i.e. it changes the WAT to a higher temperature. This method can be used to force the wax precipitation to take place in a controlled section, minimise precipitation distance and enable cold flow transport.

3.7.3 Mechanical Solution

Pigging through the entire pipeline, from well to topside, can be more costly and invasive than if the pigging was done in a smaller section. To make pigging in a smaller section possible, one can cool the fluid to ambient temperature in a controlled distance, hence, make the wax deposit in the controlled section. The frequency of pigging in the controlled section needs to be high, but the rest of the pipeline can have a less frequent pigging interval to ensure a clean pipe.

The Statoil heat pulse method, described in Section 3.7.1, is a method that can be complex and costly, as there is substantial investment and research needed. It might, therefore, be advantageous to develop a system with components and technologies that have been used in the industry for decades and proven to work. Such a system is proposed in Chapter 4, where controlled precipitation of wax takes place by cooling, as in the heat pulse method, but removal of the precipitated wax is done by deploying pigs. It is important that the oil is cooled fast in order to minimize the size of the controlled precipitation section.

4. Proposal for a Subsea Controlled Wax Deposition Unit, Enabling Cold Flow Oil Transportation

Wax precipitation is mainly dependent on temperature, and as discussed in the previous chapter, Chapter 3, there are several techniques to deal with this problem. What is chosen to be called "Subsea controlled wax deposition unit (CWDU), Enabling Cold Flow Oil Transportation", is a unit that Subsea 7 with its know-how within subsea installations can produce. This chapter proposes a unit to enable controlled wax deposition. The proposals are essentially subsea countercurrent concentric heat exchangers, which forcibly cools the product fluid to approximately ambient seawater temperature. This is done by passing product through the inner tube and pumping seawater through the annulus. This can be seen in Figure 4.1 as the white concentric pipes. Note that the figure is not to scale and that in reality it would be a lot longer. For this proposal it is assumed that most of the water and gas is separated out of the product fluid prior to entering the unit.

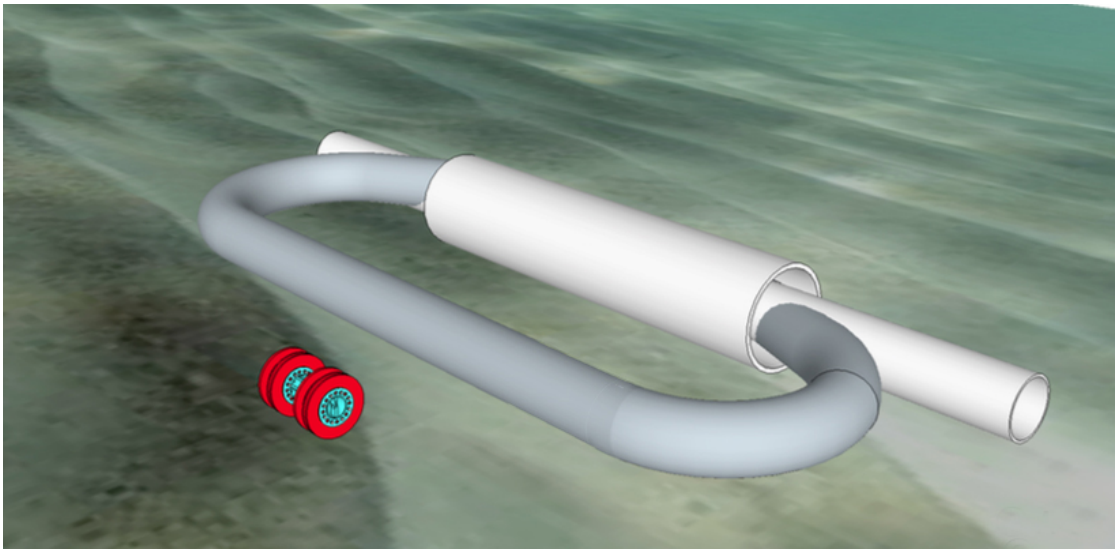


Figure 4.1: Downscaled concentric heat exchanger with a pigging loop and a red pig

When one cools the oil from above WAT to approximately ambient seawater temperature, most of the wax will then have precipitated within this length. This is what happens in the concentric heat exchanger. This unit can, therefore, be seen as the section where controlled wax deposition takes place. There will be no more, or minimal wax deposition for the rest of the pipeline, as it has reached ambient temperature at the exit of the heat exchanger. The heat pulse method, discussed in Section 3.7.1 Statoil's Heat Pulse Technology, can be used to remove the deposited wax on the pipeline unit walls. This technology has, however, not been tested and used to the same extent as pigging for wax removal. This may cause hesitation to implement the technology, due to the low technical readiness level a relatively new technology has. There is also the added problem that the electrical insulation in the concentric heat exchanger has a low thermal conductivity. This means that the heat exchanger length needed to obtain ambient or near ambient temperature will be substantially longer than if this is removed. If this is a hindrance, wax seeding can be added to provoke wax precipitation to take place at an earlier stage.

Pigging is vastly used in the industry, and can be used in the unit to remove the deposited wax. It is possible to pig the entire pipeline, but with the systems proposed in this chapter it is not necessary. Pigging only needs to be done regularly on the controlled section, due to the fact that it is in this unit almost all the wax is precipitated. Wax will still precipitate as long as there is a temperature gradient between the pipeline wall and the fluid bulk, i.e. the pipeline wall is colder than the bulk fluid, and there is more wax to be precipitated. Routine pipeline pigging can be done on the entire pipeline at a lower frequency, to ensure a clean pipeline and pipeline integrity. This can reduce the wear on the entire pipeline and reduce the risk of a pig getting stuck.

To obtain this solution a pigging loop is proposed to be added to the unit, which can be seen as the grey section in Figure 4.1. Reduction in pig deployment costs is possible by developing and implementing a remote pig launcher, compared to e.g. the deployment from ships.

4.1 Proposals

Only ones creativity sets limitations on the different design variations of a controlled wax deposition. The optimal design will vary from field to field, budget etc. There are, however, some similarities for the valves and fittings which can be used in multiple pigging loop design alternatives.

4.1.1 Alternative I

Figure 4.2 shows a schematic representation of a design alternative similar to the simplified 3D image in Fiugre 4.1. Note, the red circles indicates where the pig diverter/valves would be placed.

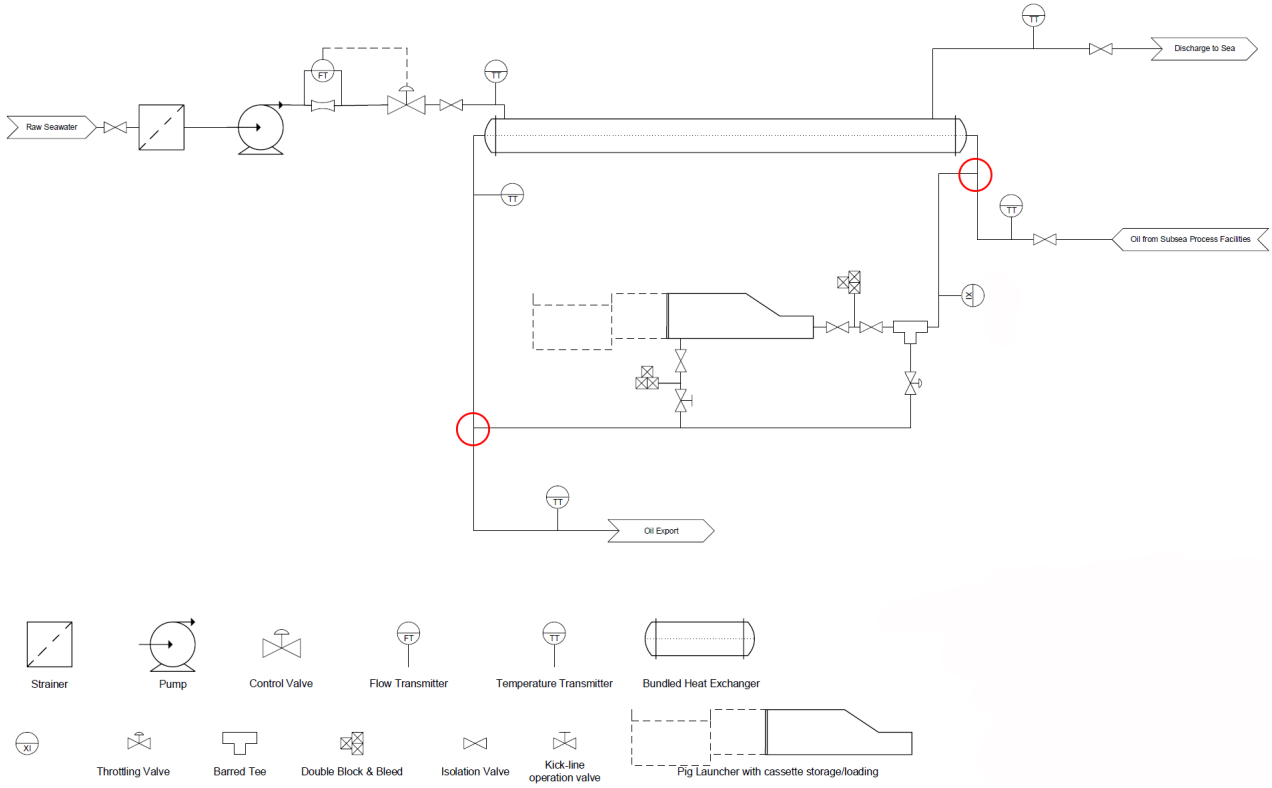


Figure 4.2: A diagram representation of the concentric heat exchanger with a pigging loop

A why fitting, see Figure 4.3, is a component which most likely can be similar for many possible designs. It can be installed on the unit where the two pipeline intersections are, see Figure 4.1 where grey and white pipes meet. A valve made of a plate which can be mechanically retracted and extended into the pipeline flow section can be added to the wye fitting on the down stream pipe intersection. Another option is a plate which can pivot around one of its end points. This is pictorially shown in Figure 4.3 with a red line. This will stop the pig from continuing down the main pipeline, and prevent it from getting stuck. The grey part of the loop in Figure 4.1 is from here on out called the stow section. The upstream pipe intersection can have a permanent, i.e. a non retracting, plate fitted in order to guide the pig and preventing it from getting stuck. This plate will have a round cut-out to let product through. An additional plate, which also can be retracted mechanically, may be installed to stop the pig in the stow section.

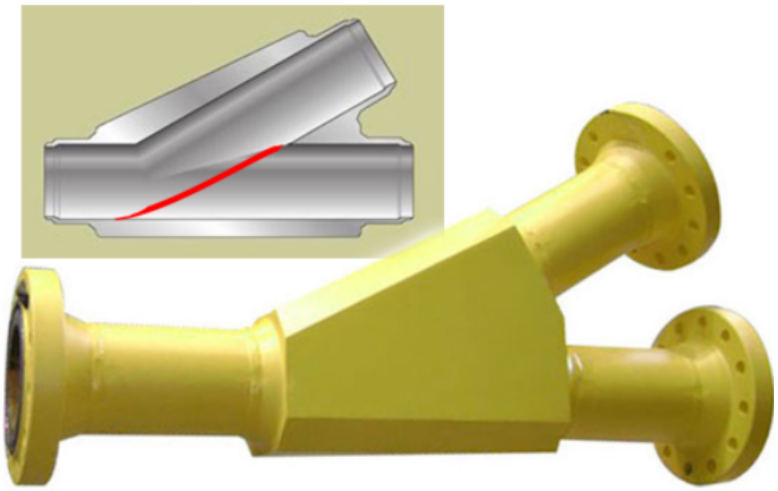


Figure 4.3: Wye fitting made by Oceaneering. The red line illustrate a proposed valve. Source, oceaneering.com wye fitting

The operation of the pig can be explained in two main modes, a regular and a cleaning mode. During the regular mode the pig is stowed in the stow section of the pipeline, with a plate holding it back. The plate located at the downstream intersection of the pipeline loop is retracted to yield maximum production flow. During the cleaning mode this plate is extended into the flow, and the pig is released by the retraction of the plate located in the stow section. The pig is now propelled by the product, which is forced through the loop, due to the partial blockage the extended downstream intersection plate creates. The pig removes the deposited wax from the pipeline walls in the loop and lets the product flow carry it to topside as solids for further processing. For this to work, the downstream intersection plate needs to block some product to allow pressure build-up to propel the pig. However, it is advantageous to allow as much product through as possible to minimize the reduction in production, and minimize pressure build-up. This can be done by the plate having cavities, letting product through while it is extended, but this may cause problems with wax clogging it.

It is important to note that the product is hot and above WAT at the upstream pipeline intersection and cold and below WAT at the downstream intersection. The product is propelling the pig during cleaning mode and will be in the stow section during regular mode. This may cause concern for potential precipitation in the stow section, as the product is not replaced in the stow section. This should, nonetheless, not be of concern because almost all of the wax will have precipitated at the downstream intersection prior to entering the stow section. However, collection of precipitated wax solids in the stow section can be an issue, but the pig should be able to remove this during cleaning mode.

It is important to monitor the wax layer thickness in the unit for several reasons: It acts as a thermal insulator making the oil cool at a slower rate, it generally increases the surface roughness and narrows the pipeline, which decreases throughput and can in a

worst-case scenario block the entire pipeline. Therefore, the unit needs to be pigged at a relatively high frequency to avoid a thick wax layer forming. This will in time cause the pig to get worn-out. A pig launcher and pig trap can be added to the stow section in order to replace worn-out pigs.

4.1.2 Alternative II

The design suggestion in this subsection is quite similar to Alternative I's design. The valves have been moved and the product flows in a loop. This design leaves less space for the pig parking and, hence, better utilises the pipeline space. As can be seen in Figure 4.4 the horizontal parallel sections of the loop can be joined in a bundle. An additional flow line can be incorporated into the bundle, e.g. if the flow is to be transported in the same direction as the inlet flow direction. This is what is illustrated in the picture above. Note that in addition to the product flow, there is an annulus flow with cooling fluid, i.e. seawater. This is not shown in the schematics. The cooling water can enclose the pipe and be pumped in the opposite direction of the product fluid, or enclose both pipe passes.

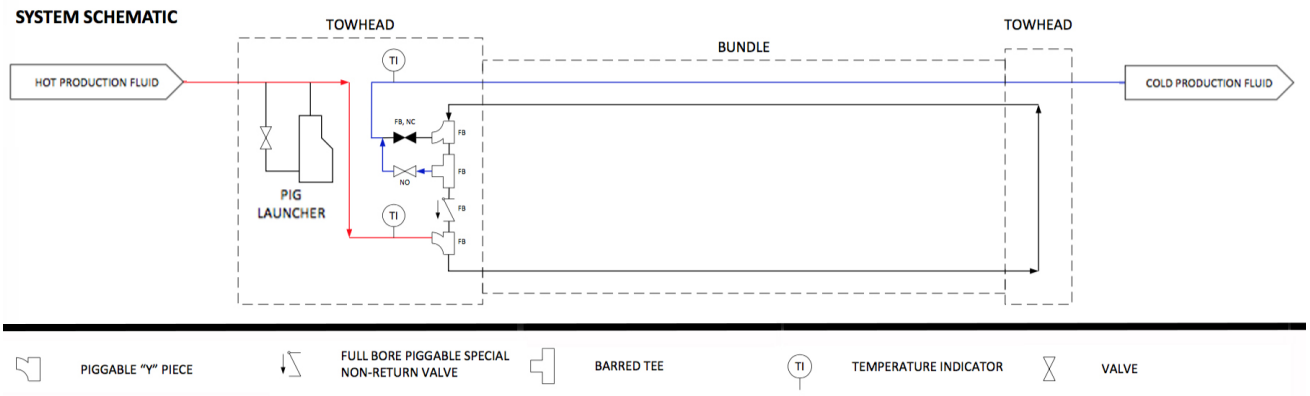


Figure 4.4: P&ID of proposed subsea unit [16]

Figure 4.5 shows the same configuration as Figure 4.4. As can be seen in Figure 4.5 one needs a piggable bend, which can be a part of the towhead. By a piggable bend it is meant that the bend should have a radius big enough to allow a pig to pass. A similar wye valve as discussed in Alternative I can also be used in this design, in addition to a pig launcher and receiver.

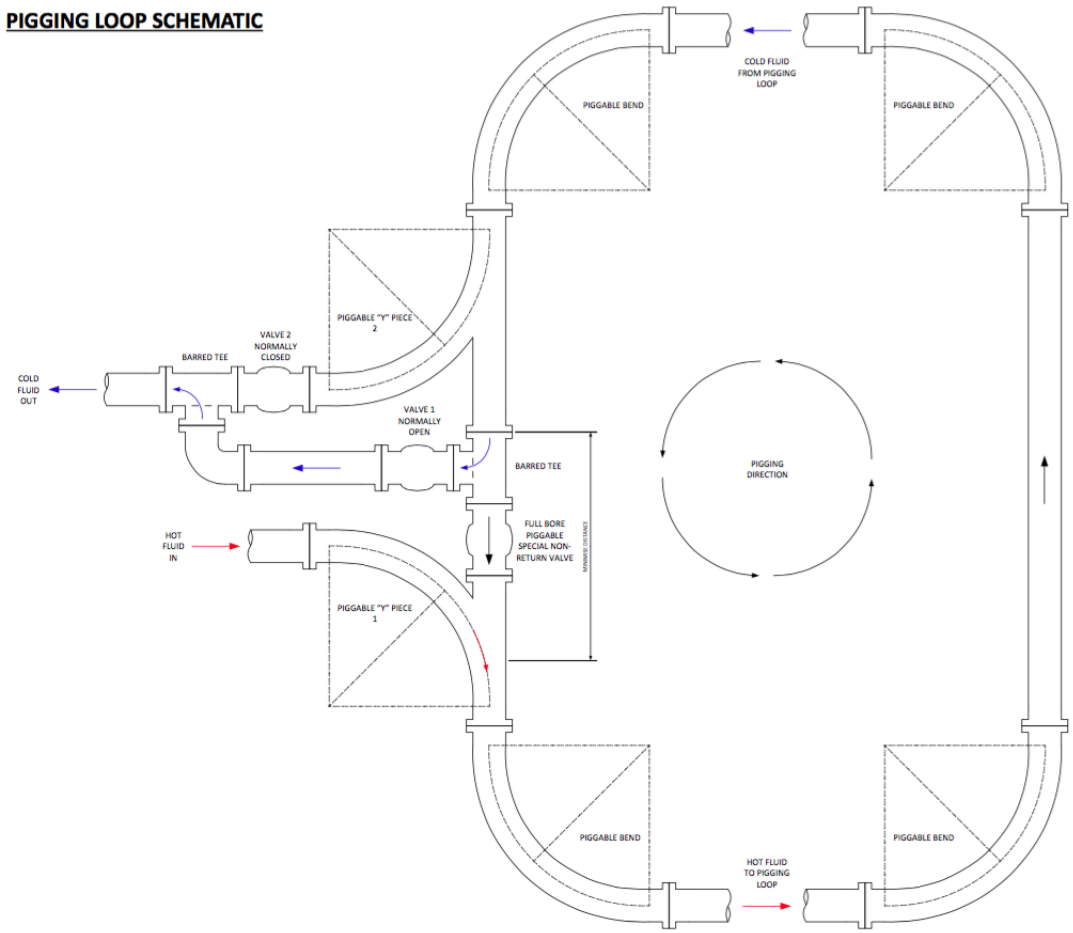


Figure 4.5: Pigging loop Schematic [16]

4.2 Scale Model

A scale model is made at Subsea 7's office in Stavanger, and has the main purpose of testing the pigging loop concept with regards to pigging and valve operation. As the test rig is not created to test wax deposition effects there is no pipe in pipe configuration.

Figure 4.6: Overview of rig [16]

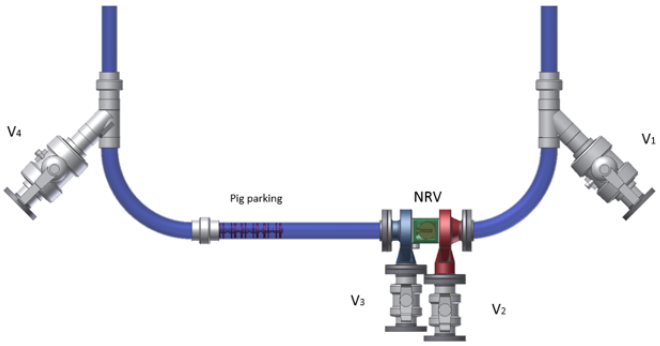


Figure 4.7: Overview of a part the rig[16]

The red section represents the inlet and the blue the outlet.

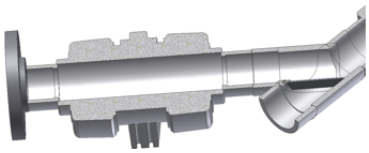


Figure 4.8: A Pig diverter(on the right) with piggable valve(on the left) [16]

The figure above shows a cross-sectional view of the pig diverter valve which allows the pig to continue in the loop or be released.

5. Heat and Mass Transfer

This chapter presents elementary heat and mass transfer that can be used to model the heat loss for the concentric heat exchanger, which was discussed in the previous chapter, Chapter 4. Wax precipitation is mainly temperature dependant, hence, it is of great importance to study the temperature profile in the pipeline to obtain a wax deposition model. Please see the Nomenclature for a description of the the variables given in the equations.

5.1 Heat Transfer

Heat transfer in a concentric heat exchanger involves convection in the hot and cold fluid, and conduction through the wall separating the two liquids.

The radial heat transfer is the same through each layer of substance, as long as there is no accumulation of heat. This can be expressed by two different equations:

$$q_r = UA_s\Delta T_r \tag{5.1}$$

A_s is the area of pipeline wall shell, where the outer wall radius is often used.

$$q_{ax} = \dot{m}C_P\Delta T_{ax} \tag{5.2}$$

The difference between the ΔT's in Equation (5.1) and (5.2) is that ΔT_r is the temperature difference between the hot and cold fluid, in the applicable pipeline section, where as the ΔT_{ax} is the temperature difference for one of the fluids from one point to the other, axially, e.g. temperature for seawater at outlet and inlet. It is assumed that there is no accumulation of heat, this yields $q_r'' = q_{ax}''$ for a pipeline sections. This is true because the amount of energy leaving a section in radial direction is the same as the energy change axially, i.e. the energy change between the sections inlet and outlet.

5.1.1 Dimensionless Numbers

In order to calculate several of the heat and mass transfer values, dimensionless numbers such as the Reynolds number, Prandtl number, Nusselt number need to be found.

The Reynolds number equation is:

$$Re = \frac{\rho v D_h}{\mu} \tag{5.3}$$

Where D_h is the hydraulic diameter:

$$D_h = \frac{4A_c}{P} \tag{5.4}$$

The Reynolds number is the ratio between inertia and viscous forces. If the Reynolds number is small, inertia forces are insignificant relative to viscous forces. The disturbances are then dissipated, and the flow remains laminar. For a large Reynolds number, however, the inertia forces can be sufficient to amplify the triggering mechanisms, and a transition to turbulence occurs[17]. Values for when a flow is laminar, transition or turbulent varies if the flow is, e.g. on a flat plate or in a pipeline. A Reynolds number $Re_D > 4000$ in a pipeline flow is regarded as turbulent. Where the subscript D refers to the pipeline diameter, which in this case is the characteristic length. The Reynolds number will change as the temperature changes because the fluid properties change, e.g. the density usually decreases with increasing temperature.

Reynolds number, with regards to mass flow rate is:

$$Re = \frac{4\dot{m}}{\pi \mu D_h} \tag{5.5}$$

Reynolds number inside annulus, with regards to mass flow rate is:

$$Re = \frac{4\dot{m} D_h}{\pi \mu (D_{out}^2 - D_{in}^2)} \tag{5.6}$$

Where D_{out}^2 and D_{in}^2 is the outside and inside diameter of the annulus, respectively.

Prandtl number equation is:

$$Pr = \frac{C_p \mu}{k} \tag{5.7}$$

The Prandtl number is defined as the ratio between the momentum diffusivity (kinematic viscosity) and the thermal diffusivity. $Pr \ll 1$ means thermal diffusivity dominates, and $Pr \gg 1$ means momentum diffusivity dominates.

Nusselt number equation is:

$$Nu = \frac{hD_h}{k} \tag{5.8}$$

The Nusselt number is the ratio between convective heat transfer and conductive heat transfer.

It can also be expressed using the Dittus-Boelter equation:

$$Nu = 0.023 Re^{4/5} Pr^{n} \tag{5.9}$$

$n = 0.4$ for heating of the fluid, and $n = 0.3$ for cooling of fluid.
Equation (5.9) is valid for:
$0.6 < Pr < 160$
$Re > 10\,000$
$fracLD > 10$

The two equations, Equation (5.8) and (5.9), can be combined and used to find the heat transfer coefficient, h.

5.1.2 Overall Heat Transfer Coefficient and Resistance

The overall heat transfer coefficient, U, in Equation (5.1) can be calculated using equation:

$$U \cdot A_c = \frac{1}{R_{tot}} \tag{5.10}$$

Where:

R_{tot} is the total resistance $[K/W]$

Note that U is dependent on A_c, e.g. if the inner diameter of the pipeline wall is chosen, U will be different than if the outer diameter wall is chosen. It is common to choose the outer wall diameter. This seldom has a major significance on the calculation.

The total resistance can be calculated by adding the convective and conductive resistances. This is pictorially shown in Figure 5.1 below.

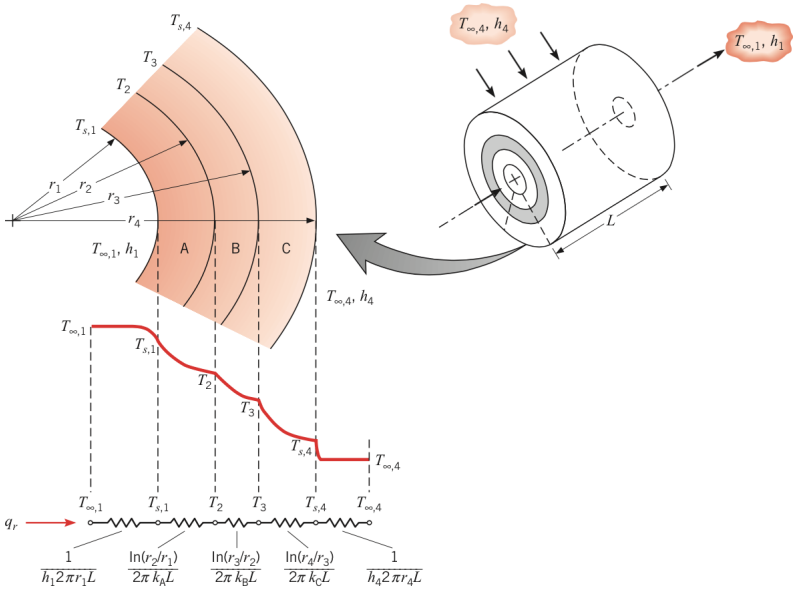


Figure 5.1: Radial Resistance[17]

Figure 5.1 illustrates a pipeline, a pipeline cross-section segment, a temperature profile passing radially from one fluid through the cylindrical wall to the other fluid, a thermal circuits representation of the resistance and each resistance equation.

The equation for convective resistance is:

$$R_{conv} = \frac{1}{h\pi DL} \tag{5.11}$$

The equation for conductive reistance is:

$$R_{cond} = \frac{ln(\frac{D_{out}}{D_{in}})}{2\pi kL} \tag{5.12}$$

Where D_{in} is the inside diameter of the wall and D_{out} is the outside diameter of the wall. A pipeline wall can be comprised of several different materials, if this is the case one can add all the resistance terms for each material layer and get the total conductive resistance.

The total resistance can be found by adding the total convective and conductive resistances.

$$R_{tot} = \Sigma R_{conv} + \Sigma R_{cond} \tag{5.13}$$

5.2 Mass Transfer

There are several equations for mass transfer that are analogous to heat and mass transfer. Chilton and Colburn J-factor analogy is an analogy between heat, momentum, and mass transfer, and is as follows:

$$J_H = J_D = \frac{h}{c_p \rho \overline{v}} Pr^{\frac{2}{3}} = \frac{K_m}{\overline{v}} \cdot Sc^{\frac{2}{3}} \tag{5.14}$$

This analogy can be used to convert the Dittus-Boelter Equation (5.9) to a mass transfer equation, which is similar to Linton and Sherwood correlation:

$$Sh = 0.023 Re^{4/5} Sc^{1/3} \tag{5.15}$$

Where the dimensionless numbers Sherwood and Schmidt respectively are:

$$Sh = \frac{K_m D_h}{D_{AB}} \tag{5.16}$$

$$Sc = \frac{\nu}{D_{AB}} = \frac{\mu}{\rho D_{AB}} \tag{5.17}$$

Note that K_m is the mass transfer coefficient $[m/s]$. Similar to the heat transfer equation these mass transfer equations can yield important information, e.g. about mass diffusivity, which can be used in a wax precipitation model.

6. Wax Deposition Models

Four different mathematical wax deposition models, various effects that causes wax deposition and techniques to measure wax deposition thickness will be discussed in this chapter.

6.1 Wax Deposition Mechanisms

There are different mechanisms that lead to wax being deposited: Molecular diffusion, shear dispersion, Brownian diffusion and gravity settling. These mechanisms are used in several theoretical models. Molecular diffusion is widely accepted as the most important. Shear dispersion is also of some importance, while Brownian diffusion and gravity settling are viewed as less influential[18].

6.1.1 Shear Dispersion

Shear dispersion is the mechanism where precipitated wax, in the fluid, is deposited onto the pipeline wall. Precipitated wax has a tendency to move with the flow. This means that the precipitated wax will move in axial direction, however, wax can get stuck to the pipeline wall or the already deposited wax layer, which is on the wall. Shear dispersion does not create new nucleation, but can cluster together with precipitated wax. This often causes the deposited wax to be softer than deposits created due to molecular diffusion, which is discussed in the next subsection.

Shear dispersion is at its highest effect after most of the wax has precipitated. This means that the fluid usually needs to be considerably below WAT in order to observe the highest effect of shear dispersion. On the other hand greater shear forces can tear the deposited wax off of the wall and reduce the thickness of the deposited wax. This is called shear removal or shear stripping[19].

The equation for shear stress, on the wall or the deposited wax, can be expressed as follows[17]:

$$\tau = \frac{1}{2} f \rho u^2 \tag{6.1}$$

6.1.2 Molecular Diffusion

Molecular diffusion starts taking place once the fluid cools and reaches the WAT, at this point wax precipitates in the solution. The surface of the pipeline wall is an ideal nucleation site for precipitation of wax. A driving force will be a concentration gradient between the wax at the wall and the bulk fluid. A laminar sublayer may cause the temperature in this thin layer between the fluid and the pipeline wall to drop. In turn this layer reaches WAT sooner than for the rest of the bulk.

A radial concentration gradient will develop as long as there is a radial temperature gradient. This is due to the paraffin molecule solubility's high dependency on the fluid temperature. This concentration gradient is the driving force for the dissolved paraffin molecules to be transported towards the pipeline wall, where the dissolved wax concentration is lower. Fick's Law is used to describe the mass transfer rate due to molecular diffusion for binary mixtures, as follows[20]:

$$\frac{dm}{dt} = \rho D_{wo} A_s \frac{\partial w_w}{\partial r} = \rho D_{wo} A_s \frac{\partial w_w}{\partial T} \frac{\partial T}{\partial r} \tag{6.2}$$

where D_{wo} is the diffusion coefficient of wax in oil, w_w is the dissolved wax mass fraction $w_w = \rho_{wax}/\rho$ and A_s is the pipe surface area. The molecular diffusion coefficient, D_{wo} can be calculated by either the Wilke–Chang (1955) or the Hayduk–Minhas (1982) correlations, given in Equation (6.3) and (6.4), respectively[20]:

$$D_{AB} = D_{wo} = 7.4 \cdot 10^{-12} \frac{(\varphi_B M_b)^{0.5}}{\mu_B V_A^{0.6}} T \tag{6.3}$$

$$D_{AB} = D_{wo} = 13.3 \cdot 10^{-12} T^{1.47} \frac{\mu_B^{(\frac{10.2}{V_A} - 0.791)}}{V_A^{0.71}} \tag{6.4}$$

Where φ_B is an association parameter for the solvent B (oil), M_B is the solvent molecular weight$[g/mol]$, μ_B is the solvent viscosity and V_A is the molar volume of solute (wax) $[cm^3/mol]$.

6.2 Pressure Drop Method

A way to model the wax thickness is the pressure drop method. This method uses the fact that wax deposition reduces the hydraulic diameter of the flow line and results in an increase in frictional pressure drop. The frictional pressure drop can be calculated using equation (6.5) and (6.6). This method has shown to be most accurate for turbulent flows. The changing pressure can also influence the fluid properties, e.g. density and viscosity, which can be important to implement in a model[21, 22].

The equation for head loss is:

$$h_f = f_D \cdot \frac{L}{D} \cdot \frac{u^2}{2g} \tag{6.5}$$

Pressure drop can be calculated using the head loss:

$$\Delta P = \rho \cdot g \cdot h_f \tag{6.6}$$

A way to visualise Equation (6.6), is to view ΔP as the pressure needed to hold a column of the fluid in question at a height h_f.

The two equations (6.5) and (6.6), can be combined yielding:

$$\Delta P = f_D \cdot \frac{L}{D} \cdot \frac{\rho u^2}{2} \tag{6.7}$$

f_D, which is the Darcy–Weisbach friction factor, can be expressed using the Swamee–Jain equation:

$$f_D = 0.25 \left[\log_{10} \left(\frac{\varepsilon}{3.7D} + \frac{5.74}{\mathrm{Re}^{0.9}} \right) \right]^{-2} \tag{6.8}$$

Note that the Darcy-Weisbach friction factor is not to be confused with the Fanning friction factor. The Darcy friction factor is four times that of the Fanning friction factor. The diameter will change as wax is deposited and the roughness might change. This will impact the pressure drop value.

6.3 Temperature Change Method

Similarly to the pressure drop method it is possible to model the wax deposition thickness, by comparing temperature values after a time interval. The wax layer acts as an insulating layer, which in turn changes the temperature profile. The thermal conductivity of the wax layer has to be known in order to use this method. The thermal conductivity can, however, change in time due to effect such as ageing, this effect is explained in Section 6.4[15, 22].

6.4 Singh et al. Model

The other models discussed in this chapter only predict the time trajectory of the deposit thickness and do not account for variation of deposition composition with time. Laboratory experiments show that the assumption that the composition of deposit is time invariant is invalid. The wax that is deposited on the pipeline walls can contain trapped oil, causing further wax to be deposited. This increases the wax content of the deposited wax. The process described is known as ageing. This phenomenon can be accounted for in the Singh et. al. model[23].

A system of coupled partial differential equations is generated in order to describe the growth and ageing of the deposit, for laminar and low shear conditions[24]:

$$\frac{d\delta_{wl}}{dt} = \frac{D_{wo}\frac{dC}{dr}\Big|_i(1-\phi(x))}{\rho x} \tag{6.9}$$

$$\frac{dx}{dt} = \frac{D_{wo}\frac{dC}{dr}\Big|_i(\phi(x)2(R-\delta_{wl}))}{\rho\delta_{wl}(2R-\delta_{wl})} \tag{6.10}$$

Where δ_{wl} is the deposit thickness $[m]$, dC/dr is the concentration gradient of the wax in the bulk fluid, x the wax content in the deposit, R the radius of the clean pipe $[m]$, D_{wo} the diffusion coefficient of the wax in oil $[m^2/s]$ and ϕ is the porosity factor found by Cussler which is given by equation:

$$\phi(x) = \frac{1}{1+\alpha^2\left(\frac{x^2}{1-x}\right)} \tag{6.11}$$

Where α is the average aspect ratio of the wax crystals[24].

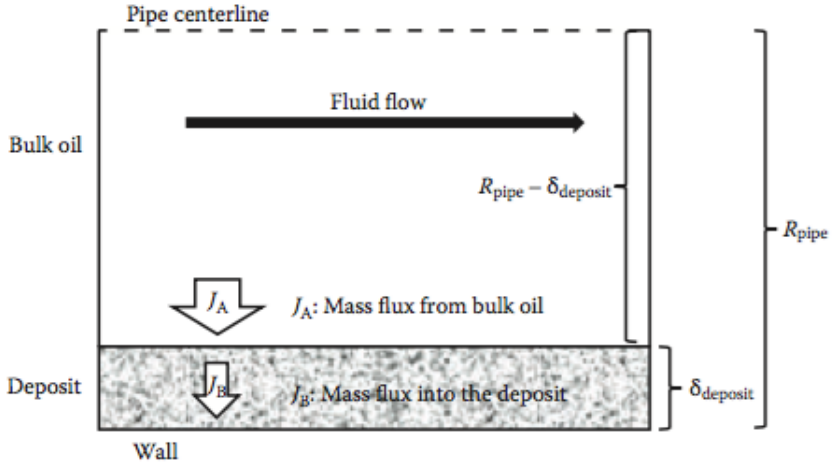


Figure 6.1: Schematic of the mass fluxes of the dissolved waxy components, where J_b illustrates the ageing effect [25]

A different and slightly simplified method to the above mentioned method is used in the MATLAB code discussed in Chapter 7. It is given by the mass flux equations[25]:

$$J_A = -D_{wo}\frac{dC}{dr} = D_{wo}\frac{C(T_{bulk})\rho_o(T_{bulk}) - C(T_{wall})\rho_o(T_{wall})}{\delta_{masstransfer}} \tag{6.12}$$

Where C is the concentration fraction of dissolved waxy components $[-]$, D_{wo} the diffusion coefficient of wax in oil $[m^2/s]$, which can be found by using Equation (6.4), subscript o stands for oil, $\delta_{masstransfer}$ is the thickness of the mass transfer layer $[m]$. NB! $C_{oil}(T_{oil})$ and $C_{wall}(T_{wall})$ are the concentration of dissolved wax at the given temperature, the same is the case for the density, ρ, and the temperature parameters in brackets.

Using the Sherwood number, Equation 5.16, in Equaiton 6.12 yields:

$$J_A = K_m(C_{bulk}\rho_{o,bulk} - C_{wall}\rho_{o,wall}) = \frac{sh}{D} D_{wo}(C_{bulk}\rho_{o,bulk} - C_{wall}\rho_{o,wall}) \tag{6.13}$$

D_wo is the wax-oil diffusion coefficient, which can be found with the use of Equation (6.4).

The definition for mass flux is:

$$J_A = \frac{dm}{dt} \cdot \frac{1}{A_s} \tag{6.14}$$

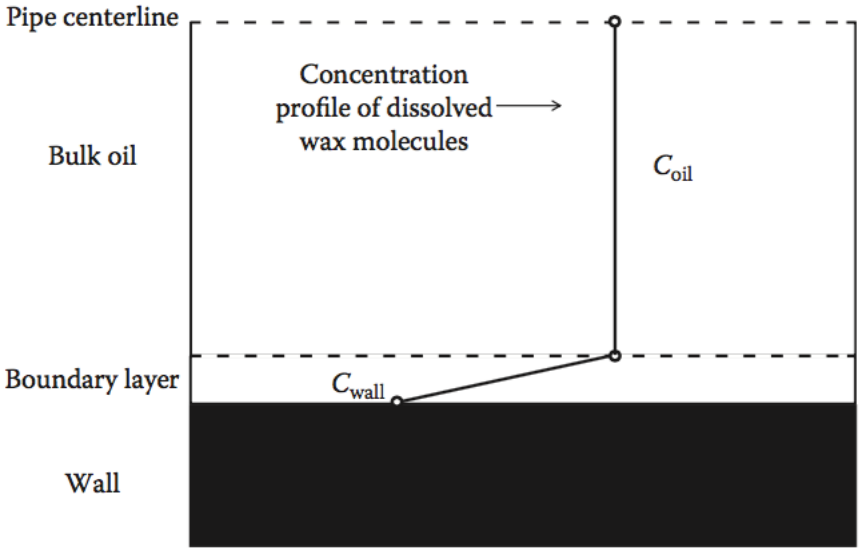


Figure 6.2: Schematic of the concentration profile [25]

The simplified equation for mass flux is when the ageing is not considered, which yields[25]:

$$J_A = \rho_{wl} F_{wax} \frac{d\delta_{wl}}{dt} \tag{6.15}$$

Where subscript wl stands for wax layer, δ_{wl} is the deposit thickness in metres.

The wax mass fraction of deposit ,F_{wax} is given by:

$$F_{wax} = 1 - \phi_{depo} \tag{6.16}$$

Where ϕ_{depo} is the porosity of the wax deposit.

With Equation (6.13), Equation (6.15) and some algebraic manipulation one gets the following equation, which is used in the MATLAB simulation discussed in Chapter 7:

$$\frac{d\delta_{wl}}{dt} = \frac{sh}{D} D_{wo}(C_{bulk}\rho_{o,bulk} - C_{wall}\rho_{o,wall}) \cdot \frac{1}{\rho_{wax}F_{wax}} \tag{6.17}$$

It is important to note that the Singh et al. model does not consider any shear stripping effects, since all of the tests which were conducted, when creating the model, was done under laminar flow conditions [26].

6.5 RRR Model

The Rygge, Rydahl and Rønningsen model, which is also known as The RRR Model, is a multiphase wax deposition model for turbulent flow. It can model wax deposition in pipelines and wells ([27] as cited by [24]).

The model continuously estimates the wax that is precipitated and is then updated. The pipeline is discretised, similarly to what is done in the MATLAB code shown in Appendix A and discussed in Chapter 7. Pressure, mixture composition and the fluid property values are assumed to be constant for each pipeline segment. The pressure drop for each segment is added, yielding the total pressure drop.

Deposition of wax alters the calculation results for the next time period. This is because the pressure and temperature will change as wax is deposited. Due to the fact that wax acts as an insulator and the surface roughness may be greater than that of a clean pipe. Additionally the pipeline diameter decreases and causes higher pressure losses. Deposition is found by: Estimating the diffusivity of wax from the bulk fluid to the pipeline surface, temperature gradient effects and shear dispersion.

The model is comprised of several sub-models: Pressure drop, flow regime, fluid properties and wax deposition models. It uses molecular diffusion and shear dispersion in the modelling of wax deposition. The volume rate of wax deposition due to molecular diffusion for a wax-forming component i is given by:

$$Vol_{wax}^{diff} = \sum_{i=1}^{NWAX} \frac{D_{wo,i}(C_{i,bulk}\rho_{o,bulk} - C_{i,wall}\rho_{o,wall})S_{wet}}{\delta_{sub}\rho_{wax}} 2\pi rL \tag{6.18}$$

Where C_i is the concentration fraction for of wax components in the fluid $[-]$, S_{wet} is the fraction of the wetted circumference, N_{WAX} is the number of wax components, ρ_{wax} is the density of wax component in i $[kg/m^3]$, r is the current inner pipe radius $[m]$, δ_{sub} is the thickness of the laminar sub-layer $[m]$ and L is the length of the pipe section $[m]$. $D_{wo,i}$, which is the diffusion coefficient $[m^2/s]$, can be found using the Hayduk-Minhas correlation given in Equation (6.4) [28].

The laminar sublayer, δ_{sub}, is given by Blasius flat plate equation [29]:

$$\delta_{sub} = 58 \cdot D_w \cdot Re_o^{-\frac{7}{8}} \tag{6.19}$$

The expression is not applicable for laminar flow.

The volume rate of already precipitated wax deposit due to shear dispersion, which is not to be confused with shear stripping, can be estimated by the following equation[28]:

$$Vol_{wax}^{shear} = \frac{k^* C_{wall} \dot{\gamma}_{wall} A_s}{\rho_{wax}} \tag{6.20}$$

Where k^* is the shear deposition rate constant, C_{wall} is the volume fraction of precipitated wax in the oil at the inner wall temperature, $\dot{\gamma}_{wall}$ is shear rate at the wall, A_s is the surface area available for deposition and ρ_{wax} is the average density of the wax. The shear dispersion mechanism is often assumed to be negligible[30].

The total rate of increase in thickness of the wax layer accounting for both diffusion and shear dispersion, in metres per second, is found from the expression[28]:

$$\frac{d\delta_{wl}}{dt} = \frac{Vol_{wax}^{diff} + Vol_{wax}^{shear}}{F_{wax} 2\pi r L} \tag{6.21}$$

The model assumes that all the wax transported to the wall will stick to the surface at temperatures below WAT, i.e. there is no release mechanisms in the model. In reality there will most likely be shear stripping in the pipeline causing the wax to loosen, especially when the flow is turbulent [24].

Combinding Equation (6.18) and Equation (6.21) without adding the shear diffusion effects yields:

$$\frac{d\delta_{wl}}{dt} = \left[\sum_{i=1}^{NWAX} \frac{D_i (C_{bulk} \rho_{o,bulk} - C_{wall} \rho_{o,wall}) S_{wet}}{\delta_{sub} \rho_{wax}} \right] \frac{1}{F_{wax}} \tag{6.22}$$

The RRR model has been applied to several single- and multi-phase systems. According to a paper published by Rygge et al. [28] two cases were presented, where wax deposition

and pressure drop was simulated. There was a good correspondence between the simulated and observed pressure values. It is, however, concluded that further improvements are needed to improve the reliability of the wax deposition simulation.

6.6 Matzain Model

Unlike the Singh et al. model and the RRR model the Matzain model incorporates shear stripping in addition to molecular diffusion and shear dispersion, in its wax deposition model. Shear stripping contributes to the removal of wax, adding this to the model might give a more realistic prediction of the wax thickness layer, compared to the Singh et al. and RRR model. Matzain et al. modelled the change in the deposit thickness as derived from Equation (6.2):

$$\frac{d\delta_{wl}}{dt} = \frac{\Pi_1}{1+\Pi_2} D_{wo} \left[\frac{dw_w}{dT} \frac{dT}{dr} \right] \tag{6.23}$$

Where $\frac{d\delta_{wl}}{dt}$ wax deposition thickness rate on the wall $[m/s]$, D_{wo} can be found by Equation (6.3) or (6.4), w_w is the dissolved wax mass fraction $w_w = \rho_{wax}/\rho$ and r is the radial distance from the pipe centre [m]([31]).

In order to include effects not covered by the standard diffusion equation, Matzain et al. added empirical correction terms, Π_1 and Π_2. Π_1 is an empirical relation for the rate enhancement due to trapped oil in the wax layer. It also accounts for any other positive deposition rate enhancements. Π_2 is an empirical relation for the rate reduction due to shear stripping. The empirical correction terms are expressed by[29]:

$$\Pi_1 = \frac{C_1}{1 - C_0/100} \tag{6.24}$$

$$\Pi_2 = C_2 N_{SR}^{c_3} \tag{6.25}$$

Where the empirical constants C_1, C_2 and C_3 are found to be 15.0, 0.055 and 1.4, respectively, for single-phase and two phase flow. C_0 is the porosity effect coefficient, which defines the amount of oil trapped in the wax layer, and N_{SR} is a dimensionless variable depending on the flow pattern [31].

$$C_0 = \phi_{depo} = (1 - F_{wax}) = 100 \left(1 - \frac{N_{RE,f}^{0.15}}{8} \right) \tag{6.26}$$

$$N_{RE,f} = \frac{\rho_o v_o D}{\mu_{o,wall}} \tag{6.27}$$

N_{SR} for single-phase flow:

$$N_{SR} = \frac{\rho_o v_o \delta_{wl}}{\mu_{o,wall}} \tag{6.28}$$

Where $\mu_{o,wall}$ is the viscosity near the wall, i.e. the oil-wall or oil-wax interface. Subscript o stands for oil. D, in Equation (6.27), is the inner pipeline diameter subtracted the wax layer. δ_{wl} in Equation (6.28) is the wax layer thickness.

As can be seen the shear stripping effect is modelled with a dependency on the wax layer thickness, flow conditions and fluid properties. It is modelled as a deposition rate reduction, directly depending on the diffusion rate, rather than as a separate phenomenon[29].

The deposition rate which is to be used in the MATLAB code discussed in Chapter 7 is given by the diffusion equation[29]:

$$\frac{d\delta_{wl}}{dt} = \frac{\Pi_1}{1+\Pi_2} \cdot \frac{D_{wo}(C_{bulk} - C_{wall})}{T_{bulk} - T_{wall}} \cdot \frac{dT}{dr} \tag{6.29}$$

The thermal gradient of the laminar sub-layer is calculated by using the following estimate:

$$\frac{dT}{dr} = \frac{(T_{bulk} - T_{wall}) \cdot h_{wall}}{k_o} \tag{6.30}$$

Combining Equation (6.29) and Equation (6.30) yields:

$$\frac{d\delta_{wl}}{dt} = \frac{\Pi_1}{1+\Pi_2} \cdot \frac{D_{wo}(C_{bulk} - C_{wall}) \cdot h_{wall}}{k_o} \tag{6.31}$$

Where k_o is the oil conductivity $[W/mK]$, and C is the wax concentration fraction $[-]$.

6.7 Heat Analogy Model

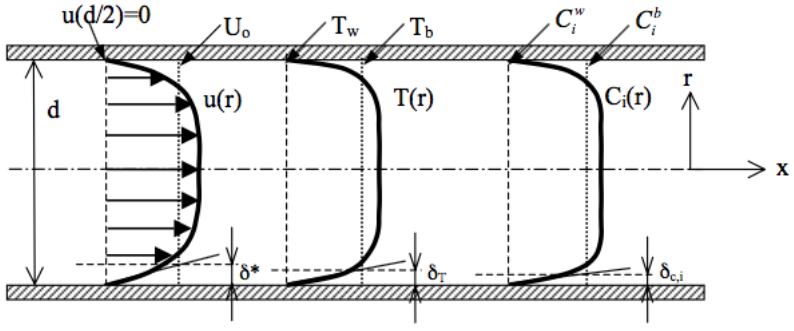


Figure 6.3: Velocity, temperature and concentration profiles [29]

The heat analogy model is applicable for both laminar and turbulent flow. Velocity, temperature and concentration profiles and there related quantities are schematically shown in the figure above. The temperature and concentration profile will often have the same form. The velocity profile may, on the other hand, often differ from the two other profiles. Properties such as: viscosity, density, thermal conductivity and heat capacity impact the flows' profiles.

The Lewis number can be rewritten using the Chilton-Colburn J-factor analogy, discussed in Section (5.2).

$$Le = \frac{Sc}{Pr} = \frac{k_o}{\rho_o C_p D_{wo}} \tag{6.32}$$

Where the Schmidt number, Sc, is given in Equation (5.17) and the Prandtl number, Pr, is given in Equation (5.7). The subscipt o stands for oil.

Algebraic manipulation of the Chilton-Colburn J-factor, Equation (5.14) yields the following equation:

$$\frac{Pr}{Sc} = \left(\frac{\rho_o C_p K_m}{h_{wall}}\right)^{\frac{3}{2}} \tag{6.33}$$

This yields:

$$\frac{1}{Le} = \frac{Pr}{Sc} = \left(\frac{\rho_o C_p K_m}{h_{wall}}\right)^{\frac{3}{2}} \tag{6.34}$$

Combining Equation (6.32) and Equation (6.34), do some algebraic manipulation and solve for the mass transfer, K_m, yields:

$$K_m = \frac{h_{wall}}{\rho_o C_p Le^{\frac{2}{3}}} \tag{6.35}$$

Note that the mass transfer, K_m, is capitalised and is not to be confused with the thermal conductivity, k. The exponent of the Lewis number is $\frac{2}{3}$ when the flow is turbulent, $0.6 < Sc < 3000$ and $0.7 < Pr < 160$. If the flow is laminar then the exponent should be removed, i.e. changed to one.

The mass transport rate of a component i through the boundary layer to the wall can be found by the following equation [29]:

$$\frac{\delta_{wl}}{dt} = K_m \cdot (C_{i,bulk} - C_{i,wall}) \tag{6.36}$$

The equation above does not take deposition reduction effects such as shear stripping into account. The shear stripping parameters from the Matzain model (see Section 6.6) is, therefore, added into the equation above:

$$\frac{d\delta_{wl}}{dt} = \frac{\Pi_1}{1+\Pi_2} \cdot K_m \cdot (C_{i,bulk} - C_{i,wall}) \tag{6.37}$$

Diffusion enhancement from the Matzain model is not required for the heat analogy model, therefore, the C_1 value discussed in the Martzain model is set to 1.0 instead of 15.0.

6.8 Wax Removal Model

A model for wax removal when pigging can be expressed as follows:

$$\dot{m}_{pig} = -|U_{pig}| m_{wxw} A_c \eta_{pig} \tag{6.38}$$

Where $\dot{m}_{pig}$ is the solid wax mass transport rate for a control volume to the wall, hence, the negative sign. A control volume will be the pipeline segment, which is discussed in Chapter 7. U_{pig} is the pig velocity and m_{wxw} is the average wax mass on the wall per control volume $[kg/m^3]$[29].

$$F_{wbf} = C_{pw}\tau_y(1 - F_{wax})\delta_{wl}\pi D_{clean}\eta_{pig}(1 - \Phi) \tag{6.39}$$

Where F_{wbf} is the wax breaking force [N], C_{pw} is the tuning factor for forces induced on the pig due to removal and transport of wax [-], τ_y is the yield stress of the wax layer [Pa], and Φ is the pig form factor [-].

The form factor is a measurement of the "cutting efficiency" of the pig. Usually the cutting tool will have a form factor close to 1. This means that the shape of the cutting devices will easily be able to remove wax. A disc or flexible sphere pig will have a much lower form factor, which is close to zero.

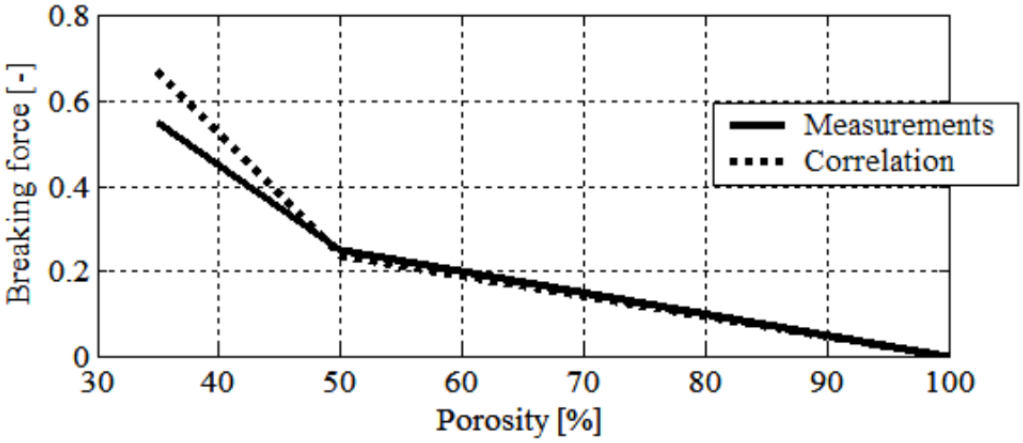


Figure 6.4: Breaking force for varying porosity, for both experimental findings and theoretical [29]

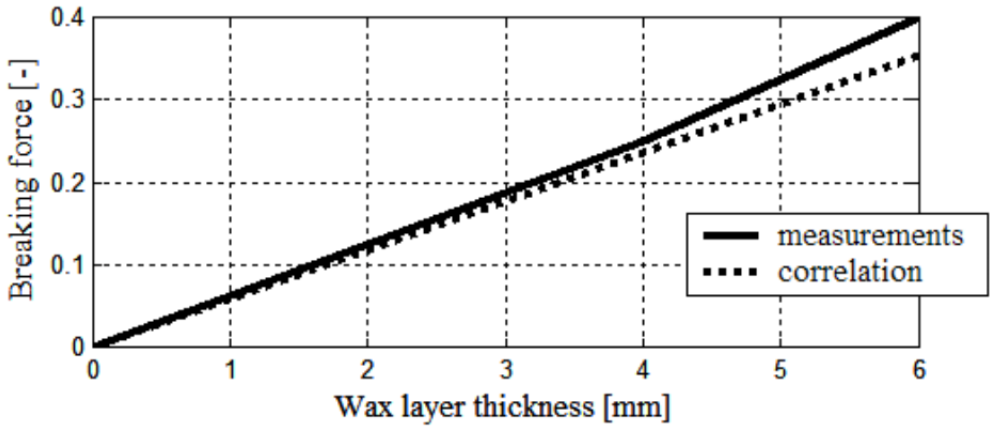


Figure 6.5: Breaking for for varying wax layer thickness, for both experimental findings and theoretical [29]

The values which are used in Figure 6.4 and Figure 6.5 are as follows: wax removal efficiency is set to 100%, the tuning factor is set to 0.00315, pig form factor to 0 and the diameter to 0.0762 m.

6.9 Wax Deposition Models Summary

The four wax deposition models and its mechanisms that are discussed in this chapter is summarised in the table below.

Table 6.1: Deposition models and its mechanisms

Wax deposition model	Wax Deposition Mechanisms:	Mechanisms not included:
RRR	- Molecular diffusion - Shear dispersion	- Shear stripping - Ageing
Matzain Heat Analogy	- Molecular diffusion - Shear dispersion - Shear stripping	- Ageing
Singh et al.	- Molecular diffusion - Ageing	- Shear stripping

7. Thermodynamic and Wax Deposition Model

A Thermodynamic Model, written in MATLAB, is used to determine the minimum length needed to cool the fluid to a temperature near the ambient seawater, when there is no wax deposited. Knowing the minimum length needed to cool the product fluid to ambient temperature, in a clean pipe, gives an indication of how long a heat exchanger unit, like the one discussed in Chapter 4, needs to be.

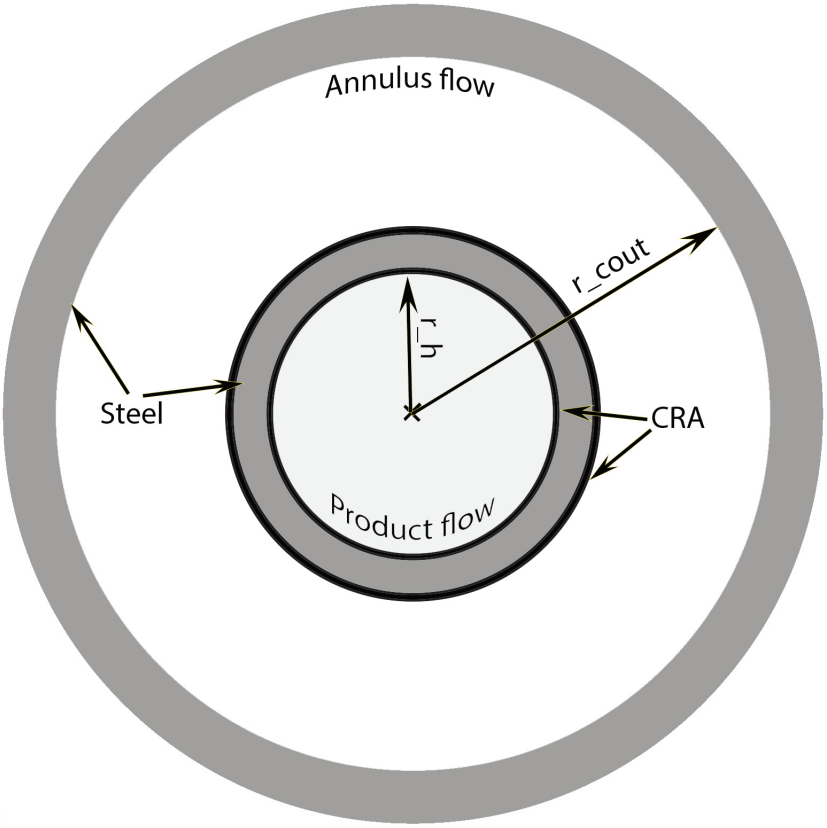
The precipitation and deposition is closely tied to the temperature of the fluid, hence, the Thermodynamic Model has to be implemented into the Wax Model. The concentration gradient due to different temperature near the wall of the pipe and the bulk, is the driving force of molecular diffusion. The deposited layer of wax acts as an insulating layer. Therefore, after some wax has deposited the Thermodynamic Model, i.e. the temperature profile and the related fluid property values, have to be updated.

This chapter explains the logic behind the MATLAB scripts located in Appendix A.

7.1 Thermodynamic Model

The MATLAB program performs various calculations for the concentric counter current heat exchanger, which was introduced and discussed in Chapter 4.

Hot product, where the majority of the water has been separated out, flows inside the pipe and seawater is pumped through the annulus. The temperature profile is found for the production fluid. This is of great importance as the wax deposition mainly relies on temperature change. In addition the script calculates values such as: length needed to cool the production fluid to a given temperature relative to ambient seawater temperature, distance to WAT and pressure drop. These calculations can form a basis to calculate the wax deposition thickness and pigging frequency needed.



Geometrical information:		
r_h	Inner wall of hot product flow	0.15 metre
t_cra_h and t_cra_c	CRA thickness	$5 * 10^{-}3$ metre
t_steel	Inner steel pipe thickness	$30 * 10^{-}3$ metre
r_cout	Outer wall of cold annulus flow	0.5 metre

Figure 7.1: Concentric pipeline (not to scale) and table of geometric values

CRA in Figure 7.1 stands for corrosion resistant alloy and is added to the pipeline to hinder corrosion. The thickness of the outer steel wall, i.e. the outside of the entire heat exchanger, is not needed. This is because it is assumed that there is no heat transfer from the outer steel wall to the surrounding seawater. Note that this is for the Thermodynamic Model calculations when it is not added to the Wax Model.

7.1.1 Boundary Conditions

In order to calculate the temperature changes some values need to be known, these are called boundary conditions. The boundary conditions for the system is chosen to be the product inlet temperature, seawater inlet temperature and product outlet temperature. In addition, values such as mass flow rate for both fluids, geometric, conductive and other fluid properties are assumed to be known, or given by NeqSim, see Section 7.4 for more on NeqSim.

7.1.2 Axial Temperature Model of Concentric Heat Exchanger

In the script the pipeline is divided into several smaller segments with a length of dz, in order to calculate the temperature profile for the entire concentric heat exchanger. This is done by discretizing several of the equations discussed in Chapter 5. A way of thinking of the discretized model is to think of each segment as a new pipe. The first segment outlet values are the second segment inlet values, and so on till the desired boundary condition is reached.

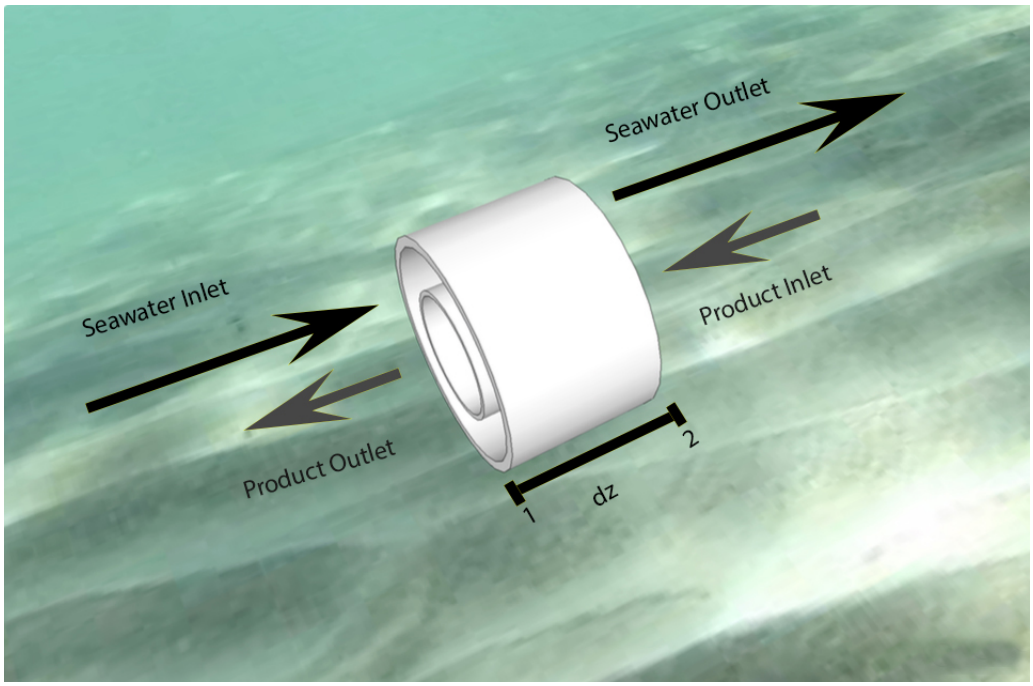


Figure 7.2: A segment, dz, of the concentric counter current heat exchanger

To illustrate how the code works, see Figure 7.2, one can look at the end segment of the pipeline where the cold seawater flows into the annulus and the hot product, which has cooled down, exits. Equations (5.1) and (5.2) are rewritten for this specific case:

$$q_r = \frac{1}{R_{tot}}(T_{h1} - T_{c1}) \tag{7.1}$$

Equation subscripts c stands for cold seawater, h for hot product, and 1 and 2 are positions on the pipeline segment shown pictorially in Figure 7.2. Equation (7.1) yields a heat transfer value, q_r, which is there due to the temperature difference between the two fluids. Further heat transfer between the annulus seawater and the surrounding seawater is assumed to be zero. The convective and conductive resistances are summed to obtain R_{tot}, as is done in equation (5.13). The resistance is recalculated for each step, i.e. each pipeline segment.

$$q_{ax} = \dot{m}_h C_{Ph}(T_{h2} - T_{h1}) \tag{7.2}$$

The fact that $q_r = q_{ax}$ can be used to obtain the unknown T_{h2}. This can similarly be done for the seawater annulus flow, to obtain the next step temperature, T_{c2}. For the next step; The first segment's T_{h2} will be equal to the next segment's T_{h1}.

The script's first segment is the inlet flow of the seawater and the outlet flow of the hot product. The script stops adding new segments and calculating new temperature values when the inlet temperature boundary condition for the hot product is reached. Multiplying number of segments and the step size, dz, yeilds the heat exchanger length.

7.2 Wax Model

The Thermodynamic Model is further developed and incorporated into a wax deposition program. There are different deposition models added to the program; Singh et. al., RRR, Matzain and the Heat Analogy model. It calculates various parameters to determine things such as, wax deposition thickness for a controlled deposition unit, described in Chapter 4. This section discusses the wax deposition program and describes how the code is structured. Below is a flow chart of the MATLAB script for the Wax Model.

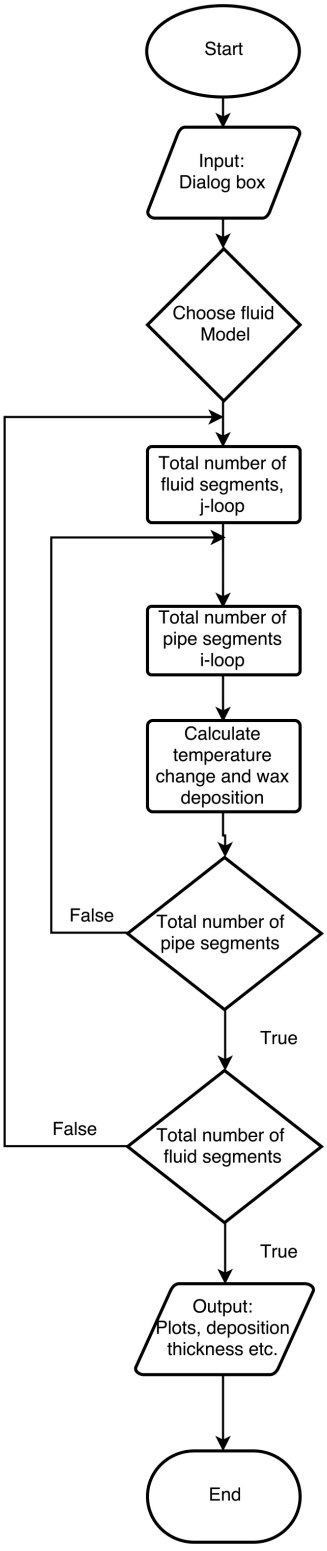


Figure 7.3: Flow chart of the Wax Model
%label

For the wax simulations it is assumed that the length is known, contrary to the Thermodynamic Model where the length is not known. In practical terms it means that one now can regard the product inlet as the first step for the discretized pipe. It is additionally assumed that there is no heat accumulation in the cooling water that runs through the annulus. Heat is also transported from the cooling water to the cooler outside seawater, and the water heat capacity is higher for water compared to the product fluid. A potential gain in cooling water heat would, therefore, not be high. However, The impact on the simulations time, if added, would be. This is because one would have to update the cooling water temperature profile for each step in addition to the product's.

The program is split into several scripts to make the program easier to navigate, e.g. for debugging. The majority of the scripts used can be found in Appendix A.

NB! This section attempts to verbally explain the structure and thought behind the MATLAB code. The best understanding will most likely be gathered by also studying the code it self, which is located in Appendix A.

7.2.1 Methodology

As discussed in the previous section the pipe is split into several segments. It is assumed that properties within each segment are constant, e.g. the temperature of a sliver of fluid located inside a pipe segment is assumed to be the same near the entrance and near the outlet side of the segment. A lot of the wax simulation code is built upon the Thermodynamic Model code. This means that a lot of the equations, such as Equation (7.1) and (7.2), are reused in the wax simulation code. There are also new parameters added such as the important wax resistance parameter, which is added to the heat transfer equation, R_{wax}. This value is equal to zero where there is no wax deposit, e.g. above WAT or after pigging. The wax has a low thermal conductivity relative to the steel wall.

From the various wax deposition models discussed in Chapter 6, one can see that the period of time a given quantity of the product fluid is in a pipe segment is correlated to how much wax is deposited in that pipe segment. If there is a low mass flow rate then the fluid is travelling slower through the pipe, compared to if it were to have a higher flow rate. This allows more wax to be transported to the wall. Here the concentration gradient is the driving force. Additionally slower flowing fluid is cooled more letting more wax precipitate in the given pipe segment. This is assuming that there is wax to be precipitated within the temperature range.

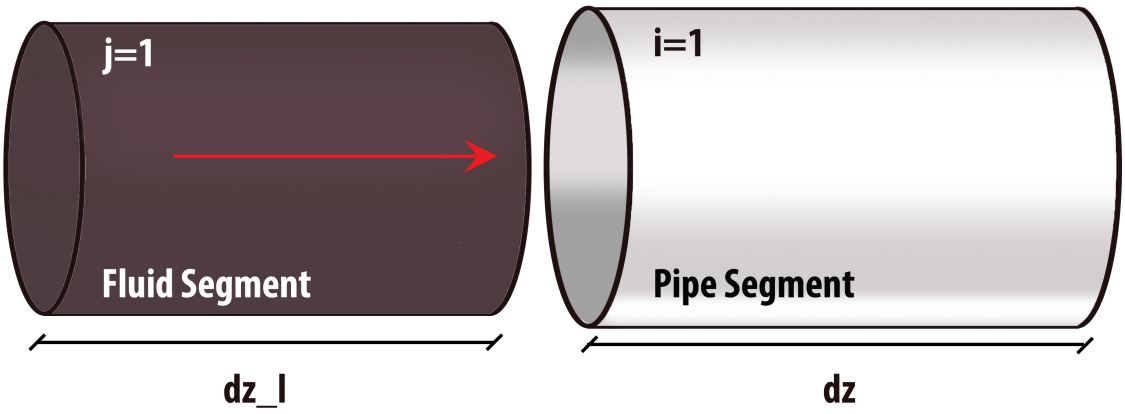


Figure 7.4: Fluid segment and pipe segment

A fluid segment is now defined as a segment of fluid travelling through the pipe line, see figure 7.4. The fluid segment dimensions are defined in order to calculate its mass, volume, and in turn how much time a given fluid quantity is within a pipe segment. The fluid segment is defined to be a certain length, in this case the default is set to be equal to the pipe segment length. There does, however, not have to be any correlation between the fluid length and the pipe length. One can view the fluid segments as fixed control volumes and the pipe segments as moving control volumes.

Pipe segments that are long are not advisable, because this will result in the true temperature between the two far ends being greater. Fluid segments that are long are also not advisable, due to the fact that the temperature profile is not updated before the entire fluid segment has passed the pipe segment. This is not a problem if there is no wax deposition. When there is deposition, however, the temperature profile of a long fluid segment will be less frequently updated. Due to the insulation effects caused by wax, the less frequently updated temperature profile will yield a less accurate model. When simulating longer time periods it might be advantageous to use a longer fluid segment. As this reduces the number of calculations which needs to be made, hence, reduces the computational power and in turn the time it takes for the simulation to finish.

The program does not allow the fluid segments to be smaller than the pipeline segments. There is an if statement that checks this and outputs an error where the user is asked to input a value where the fluid segment is equal to or greater than the pipe segment length, if this is not the case. The reason is the way the time calculations are made and the heat transfer calculations are structured. It is possible to make a code where there is no such restriction, however, for these simulations a fine pipeline grid is more important for the accuracy than a fine fluid grid.

The fluid segment has an initial given length, and an initial cross sectional area equal to the clean pipeline cross section. It has a set mass, volume and a set mass flow rate given by the inlet conditions. There is precipitation of wax within the fluid segment

when it reaches a section of the pipe where it has cooled to the WAT. At this point the fluid starts to precipitate wax and deposit it on the pipeline wall. Now there is a thin layer of wax deposited on the pipe wall, this alters the heat transfer properties, as wax acts as an insulator. The next fluid segment entering the pipe segment, which now has the deposited wax, is not cooled as much as it would have been in the case of a clean pipe. Hence, the WAT for this particular fluid segment might now be one pipe segment farther downstream. It is assumed that the flow is highly turbulent and that there is good mixing, but note that the WAT is reached near the wall earlier than for the bulk fluid. As wax is deposited on the wall the mass flow changes, this change is minuscule and can be assumed to be negligible, but can be added to improve the model accuracy.

As mentioned in Chapter 6 the main deposition mechanism is molecular diffusion, and a driving force is the radial concentration gradient between the wax at the wall and the bulk fluid. This concentration gradient enables transport of wax from the bulk to the wall. The reason for the concentration gradient is the temperature difference. The radial temperature profile is pictorially shown below.

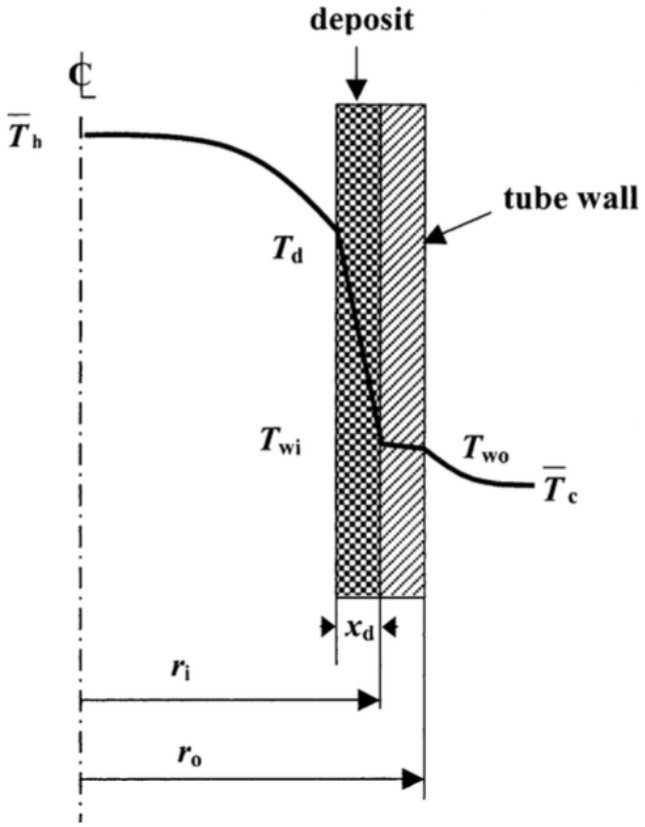


Figure 7.5: Temperature profile with deposit [17]

T_c, in the figure, is the annulus fluid's temperature and T_h is the product's bulk temperature.

7.2.2 MATLAB Loops

The MATLAB program arranges the fluid and pipe segments in two loops, where the pipe segment is denoted by i, and i=1 is the product flow inlet. The distance from inlet to a certain pipe segment can be calculated by multiplying i with dz, where dz is the segment length in metres. The fluid segment is denoted by j. The MATLAB code has a nested loop i.e. one loop within the other, one for j and one for i. The pipeline segment loop, i, is inside the fluid segment loop, j. Thus all the pipe segments are calculated for a certain fluid segment, and then all the pipe segments for the next fluid segment is calculated etc. Note that the first pipe segment is the product inlet, unlike the Thermodynamic Model where the inlet is regarded as the last pipe segment. The reason for this is that, unlike the Thermodynamic Model, for the Wax Model the pipeline length is assumed to be known.

The i and j loop is arrange in a matrix. To illustrate this, a matrix is written for the temperature. A vertical step corresponds to a pipeline segment step and a horizontal step corresponds to a change in the fluid segment.

$$\begin{bmatrix} T_{(1,1)} & T_{(1,2)} & T_{(i,j)} & \dots \\ T_{(2,1)} & T_{(2,2)} & T_{(i,j)} & \dots \\ T_{(3,1)} & T_{(3,2)} & T_{(i,j)} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

When pipe segment, i, is equal to the total pipe segments, i.e the end of the pipe, then all the calculations for the given j, fluid segment, has been calculated. The deposition along the pipeline that a certain fluid segment has deposited, will be a part of the calculations for the next fluid segment's temperature and heat transfer profile calculations. It is important to note that if one looks at e.g. the 2nd fluid segment ($j = 2$) then the 3rd ($j = 3$) will be the next fluid segment, i.e. upstream in relation to the 2nd fluid segment. The arrangement of the loops make the program calculate one fluid segment's flow through the pipeline at a time. Hence, the time between e.g. The 50th pipe segment and 1st fluid segment ($i = 50$, $j = 1$) and the 50th pipe segment and 2nd fluid segment ($i = 50$, $j = 2$) is only the time it takes a fluid segment to cross a pipe segment. This loop arrangement is made so that all wax deposition that happens due to a fluid segment is updated for the next fluid segment.

7.2.3 Mass Balance

For every pipe segment step, for a particular fluid segment, the wax deposited can be subtracted from the flow. Ideally all the wax that can precipitate within the temperature range, i.e. product inlet temperature to near ambient seawater temperature, should have precipitated within the unit. A big portion of the wax dissolved in the product fluid will, therefore, no longer be present at the product exit. This means that the mass flow rate will be slightly changing downstream. However, there is also shear stripping that

will add wax to the stream. If there is an increase in the deposition layer then the mass at the outlet will be slightly lower than at the inlet.

Ideally all of the wax that can precipitate should have precipitated within the unit length to avoid deposition outside the unit. Because of the insulating effects of the wax the length of the unit has to be longer than that of the minimum length calculated in the Thermodynamic Model, if minimum to no wax is to be precipitated outside the unit. Another way to look at it is that a given mass of the fluid is entering the pipe segment and a certain amount is exiting, the difference is the amount of deposited wax. Note that there will also be shear stripping, which will be a part of the mass flow exiting the pipe segment.

7.2.4 Elapsed Time Calculation

The time it takes the front side of the fluid segment to travel from the front to the end of the pipe segment, is given as the $time\_seg\_to\_next\_PIPEseg$ in the MATLAB code in Appendix A. The time it takes the entire fluid segment to pass through the inlet of a pipe segment is given as the $time\_seg\_to\_next\_FLUIDseg$ in the MATLAB code. These values will be the same if the length of the fluid and pipe segments are equal.

The time it takes all of the fluid segments to pass through the entire pipeline is calculated by summing all of the $time\_seg\_to\_next\_FLUIDseg$ for the first fluid segment and then add the remaining fluid segments $time\_seg\_to\_next\_FLUIDseg$ for the last pipe segment. The sums of these values will yield the run time. Similar calculations can be made for the time it takes for a given fluid segment to reach a certain point in the pipe. Simply assume the pipeline exit to be at the point which is to be calculated, and do the same calculations, as previously mentioned, for the desired number of fluid segments. E.g. if the fluid segment is $j = 300$ and one wants to find out the time elapsed for the segment to travel from inlet to pipe segment $i = 1000$, one can add all the elapsed times for the first fluid segment's flow through each pipe segment. I.e the time it takes the first fluid segment to travel from pipe segments $i = 1$ to $i = 1001$, and then add the $time\_seg\_to\_next\_FLUIDseg$ for the last pipe segment for the remaining 299 fluid segments. Note that when the 300th fluid segment is at pipe segment $i = 1000$ then the first fluid segment will approximately be at the 1300th pipe segment if the fluid and pipe segment lengths are equal.

If one chooses to model e.g. 300 fluid segments, and one sets each fluid segment to be one-metre-long, the practical equivalent will be to open a valve and let 300 meters of product fluid pass, then turn the valve off. The 300-meter-long fluid is now assumed to be one continuous fluid travelling through the pipeline. The time it will take from opening the valve to closing can be calculated and is denoted $time\_valve\_open$ in the MATLAB in Appendixit A.

If the elapsed time of e.g. experimental trials are to be simulated it is possible to first run the Wax Model with a few fluid segments. Then it is possible to calculate the number of fluid segments needed to get the desired time by checking the

time_seg_to_next_FLUIDseg or the *time_valve_open*. By choosing the number of fluids segments rather than elapsed time one has more control over how many calculations are made and in turn how much time the program takes to complete the simulation.

7.2.5 Radial Heat Transfer

When calculating the radial heat transfer, q_r, when the fluid segment is longer relative to the pipe segment, one has to make two calculations for radial heat transfer. One for the heat loss for the fluid within the pipe segment and one for the heat loss for the entire fluid segment length, $q_{rfluidseg}$. To get the $q_{rfluidseg}$ one can multiply the pipe and fluid segment length ratio, i.e. fluid segment length divided by pipe segment length and multiply this ratio with the heat loss for the fluid within the pipe segment. The reason for this is that the q_r value is used to calculate the axial temperature difference, $q_r = q_{ax}$ as discussed in Section 5.1. The wax deposition values are calculated for the entire fluid segment that has flowed through the pipe segment. However, one has to calculate the temperature for the fluid which enters the new pipe segment, which is done using the q_r value. Adding the $q_{rfluidseg}$ for all the pipe and fluid segments will yield the total heat loss for the product fluid in the unit.

7.2.6 Pigging after fluid segments

The wax simulation does not simulate wax removal by the pig. It is, however, thought that a pig removes all of the deposited wax and it is located upstream of the last fluid segment. After the removal the deposition process starts over again, this can be defined as a deposition-cleaning cycle. This means that the peak wax thickness, which has to be removed by the pig, will be given as the last deposition thickness values. Because it is assumed that the pig removes all of the deposited wax, all of the deposition-cleaning cycles will be the same. If not all of the wax is removed by the pig, then previous simulation values have to be used for the next deposition-cleaning cycle.

If there is only one pig in the controlled deposition unit. The minimum time for one deposition-cleaning cycle is the time it takes a pig, which is the same time as a fluid segment, to travel through the loop back to the start position.

7.2.7 Narrowing Pipe

When wax is deposited the pipe diameter is reduced. This lowers the cross sectional area elongates the fluid segments and increases the fluid's flow velocity, due to the constant mass flow rate. The net result is that the time it takes the fluid segment to pass through the pipe will be the same when varying the pipe diameter with a constant mass flow rate. One can assume this effect to be negligible if the deposit is low. Due to these facts the time is calculated by only using the clean pipe cross sectional area.

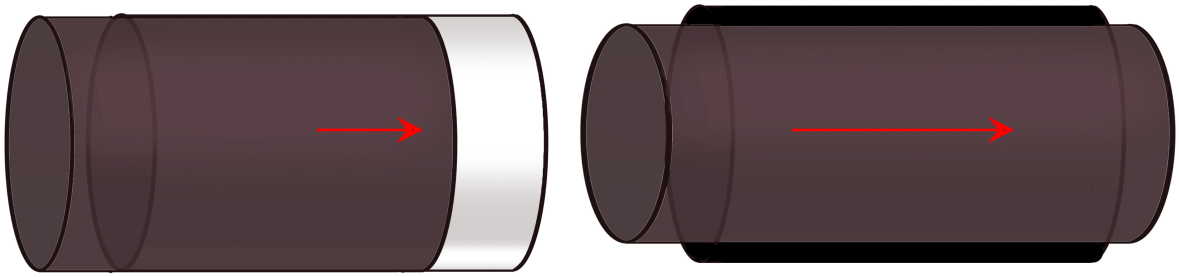


Figure 7.6: Regular fluid segment (left). Fluid segment elongated due to wax deposition (right)

The figure above illustrates the elongation of the fluid segment due to wax deposition. The elongation of the fluid segment could also affect the simulation if the down stream annulus temperature varies a lot. As the fluid segment will cover a longer stretch, hence, there will potentially be a bigger temperature difference than what is calculated. These effects are assumed as negligible.

7.3 Assumptions and Shortcomings

The fluids in the simulation is assumed to have Newtonian properties, however, oil with high wax content can be non-Newtonian below WAT, due to the presence of wax crystals[6]. The bulk fluid temperature is used to yield fluid properties, it is assumed that there is adequate amount of mixing, turbulence. The turbulence makes the fluid temperature more uniform in radial direction. Figure 7.7 illustrates what the laminar velocity profile looks like. The velocity profile is less concave for turbulent flow. The WAT is also found using the fluid bulk temperature, but there will most likely be precipitation of wax near the wall earlier, as the temperature is lower than the bulk and reaches WAT before. Effects such as the laminar sublayer, entrance effects, no-slip condition are all effects that are not taken into account in the simulations.

To simplify the Thermodynamic Model it is assumed that there is no heat loss between seawater in the annulus and the ocean seawater, i.e. seawater outside the heat exchanger. Due to the fact that it is advantageous to cool the oil as fast as possible, as it shortens the heat exchanger length, it is wise to let as much heat as possible be transferred from the annulus seawater to the ocean seawater. This means that in reality the hot oil can be cooled quicker, and the WAT can take place sooner than what the Thermodynamic Model predicts.

In turbulent flow the fluid is more uniformly cooled due to the mixing properties of turbulent flow, this in turn cools the fluid more rapidly, which in this case is desirable. This can be seen in the equations in Chapter 5: Higher Reynolds number $\rightarrow$ higher Nusselt number $\rightarrow$ higher heat transfer coefficient $\rightarrow$ higher Heat transfer. As opposed to laminar flow, which has a lower Reynolds number and usually a lower heat transfer.

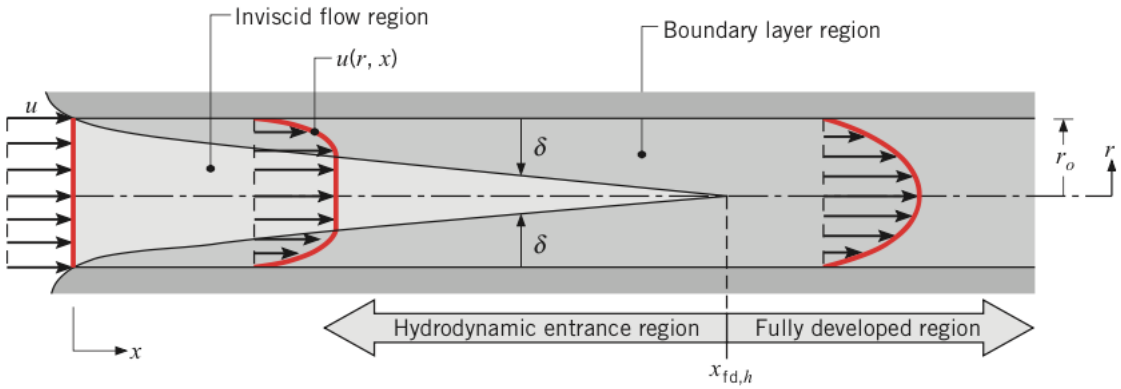


Figure 7.7: Laminar, hydrodynamic boundary layer development in a circular tube[17]

It is computationally strenuous to calculate a new temperature profile for the annulus flow for every fluid segment and pipe segment where wax is deposited. The seawater Cp is high compared to the oil. This means that the temperature will not change as much as the temperature in the product for every watt of heat transferred. For the Wax Model it is assume that the temperature for the seawater annulus flow is constant at ambient seawater temperature. This means that there is no heat accumulation in the annulus. As mentioned above the Thermodynamic Model yields the minimum required length. This gives a good estimate as to how long the heat exchanger unit has to be to cool a hot fluid to a given temperature, when there is no wax deposited. As will be shown in Chapter 8 the temperature does not increase much when it is assumed that all of the heat is contained in the annulus. This means that in reality the annulus fluid temperature will be lower than what is estimated. Therefore, it can be assumed that the seawater temperature profile in the annulus is constant for the wax deposition program.

When one is making the assumption that there is no heat accumulation in the annulus cooling water, one indirectly assumes that the mass flow is high enough to not change the annulus temperature significantly. This can be checked by running the Thermodynamic Model simulation and seeing how much the outlet temperature is, when there is no heat loss to the sea.

7.4 NeqSim

A program called NeqSim, which stands for Non-Equilibrium Simulator, was implemented in the MATLAB code.

NeqSim is a dynamic process simulator, programmed in Java, specially designed to handle non-equilibrium situations. Common non-equilibrium processes include absorption, distillation, multiphase flow in pipelines, drying processes, hydrate formation and heat exchange. NeqSim also handles traditional equilibrium process calculations (equilibrium separators, equilibrium streams)[32]. In the simulations given in Appendix A, NeqSim

was used to obtain various fluid dynamic properties, such as density, heat capacity and conductivity, for the fluid composition given in Table 9.4. NeqSim is not implemented for the pumped seawater, because the fluid properties for seawater will not change much. The fluid properties are found by inputting temperature and pressure values in NeqSim.

7.4.1 NeqSim Tuning and Regression

The fluid composition data shown in Table 9.4 was added to the NeqSim simulation using MATLAB via an Excel document, the MATLAB script can be seen in Appendix A. The script converts the values from the Excel document into matrices. For other compositions it is possible to edit the excel document to fit the new compositional data.

The wax precipitation values given in NeqSim is tuned to better match experimental findings, which can be found in Table 9.3. The graph below shows the experimental values, a regression of these values, the tuned NeqSim values and a regression of the tuned values.

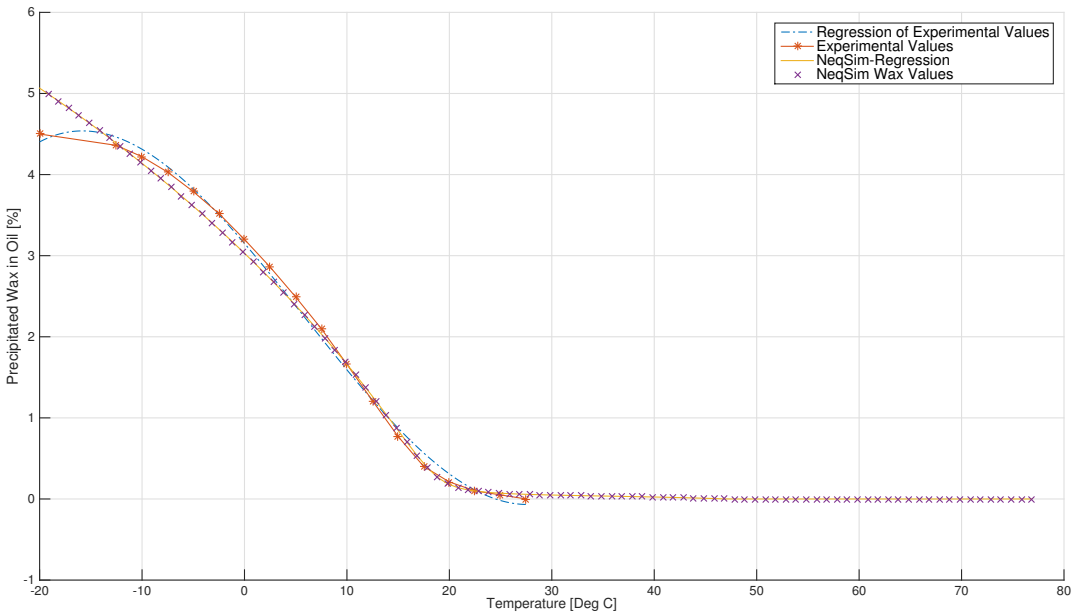


Figure 7.8: NeqSim wax precipitation values

The tuning was done by adjusting the fugacity, triple point and parameters A, B and C given by the the Wax Model, Pedersen (1995) as modified by Rønningsen et al. (1997) and based on the following assumptions: Only C7+ components can form wax, and only part of each C7+ carbon number fraction contributes to the wax formation. The wax forming fraction essentially corresponds to then n-paraffin fraction. The mole fraction

of wax-forming components contained in a given carbon number fraction having the average molecular weight M_i and the average density ρ_i follows the expression[33]:

$$z_i^s = z_i^{tot}\left[1 - (A + B \cdot M_i) \cdot \left(\frac{\rho_i - \rho_i^P}{\rho_i^P}\right)^C\right] \tag{7.3}$$

Where z_i^{tot} is the total mole fraction of carbon number fraction i. A, B, and C are empirical constants determined from experimental wax precipitation data. ρ_i^P is the density of a normal paraffin of the same molecular weight as carbon number fraction i. The following expression is used for the n-paraffin density in [g/cm3] [33]:

$$\rho_i^P = 0.315 + 0.0675\ln(M_i) \tag{7.4}$$

As is shown in Figure 7.8 it is also possible to make a regression of the experimental data and not use NeqSim. Such a regression might cause problems. The regression of experimental values, as can be seen from the above mentioned figure, goes above 4.5 wt% wax. This is not a problem because the simulations will not go below 0°C. The values near zero wt% wax on the other hand is used, and this value goes below zero. This is corrected by writing an if statement that checks and setts any value below zero wt% wax equal to zero.

All the wax deposition models, presented in this thesis, use dissolved wax as the parameter when calculating wax deposition thickness. The dissolved wax can be derived by the precipitated wax values. Max dissolved wax is set to be 4.5wt% for the fluid used. One can, from Figure 7.8, see that the experimental values peak at about 4.5 wt% wax precipitation. This is why 4.5wt% is used for this fluid. To find the dissolved wax one can simply subtract the deposited wax from the defined maximum possible precipitated wax, i.e. 4.5 wt% for the fluid discussed in this thesis. There is a discrepancy for the experimental values and the NeqSim values at low temperatures, however, the simulations done does not go down to these low temperatures. Due to the fact that the difference between the wall concentration and bulk concentration is used it does, in fact, not matter what the maximum dissolved wax value is defined as.

When regressions are made in MATLAB it is important to note that oscillations may occur. This is the case for the regression done for precipitated wax, when the values reaches zero. This can be solved by setting the values equal to zero above WAT.

Tuning was also done for the fluid viscosity values to better match the experimental data. NeqSim uses the friction theory to find viscosity and tunes the data to fit the experimental values.

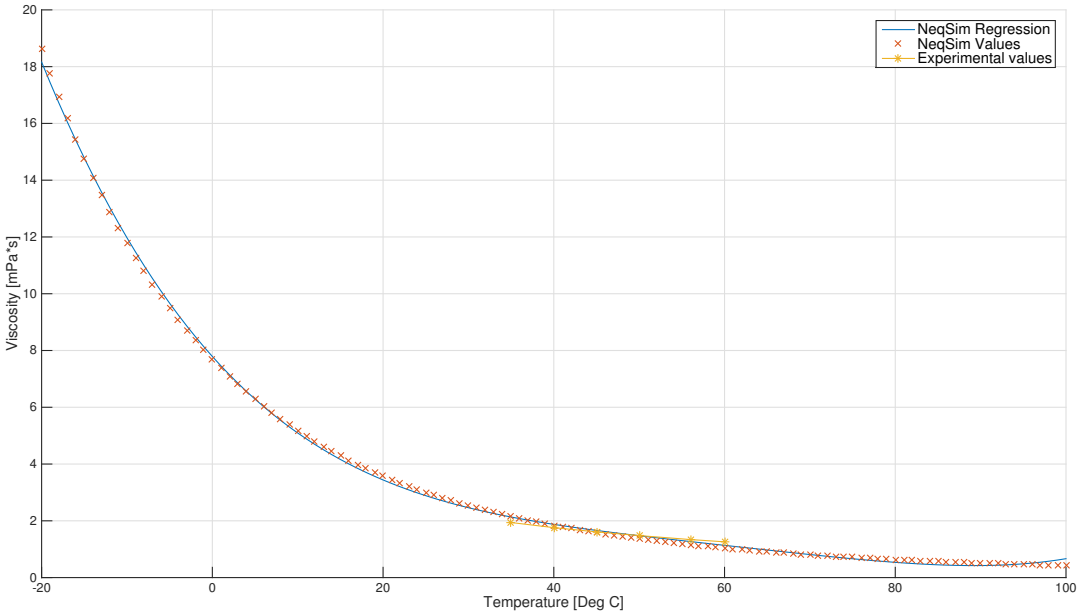


Figure 7.9: NeqSim viscosity values

The reason for creating a regression for several of the values given by NeqSim is to increase the speed of the code, by not having to get values from NeqSim for every change in system conditions. Which for this simulation will be the change in temperature. There is also the added benefit of gaining more control over the program by having a better overview of the values given by NeqSim, and the fact that the program can run without being connected to the NeqSim database. The regression values are stored in .txt files and a .mat file, which are loaded into MATLAB when the simulation is run.

The thermal conductivity, heat capacity, density and molar volume for precipitated wax, was also found using NeqSim and regressed. There respective values can be seen from the graphs below.

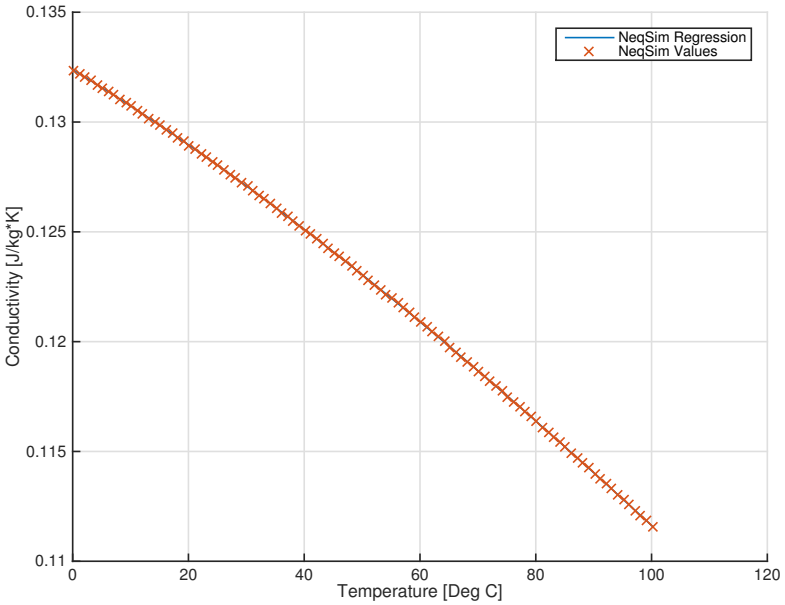


Figure 7.10: NeqSim thermal conductivity values

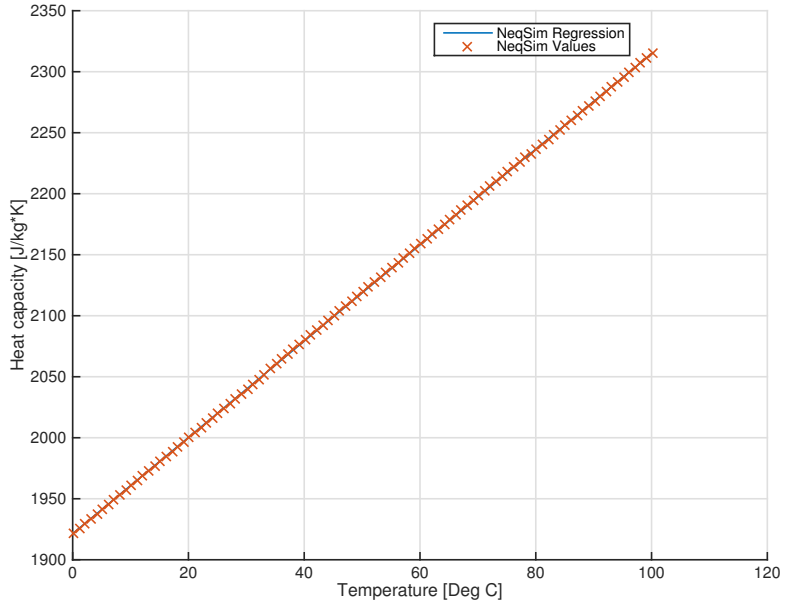


Figure 7.11: NeqSim heat capacity values

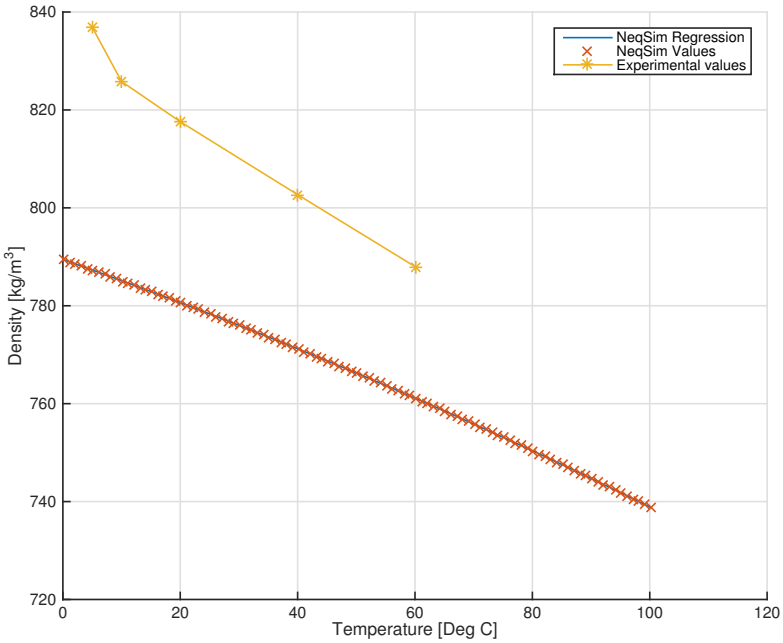


Figure 7.12: NeqSim Density values

The experimental values and the NeqSim values shows some deviation, the difference is, however, not substantially big relative to each other .

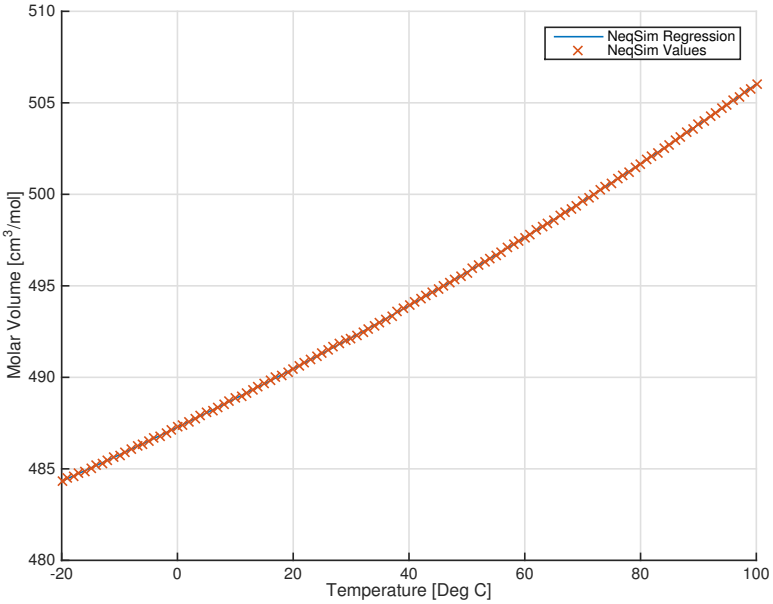


Figure 7.13: NeqSim wax molar volume values

One of the parameters that is used to calculate the diffusion coefficient is the wax molar volume, V_A. The wax molar volume values for a varying temperature is found using NeqSim. As can be seen from Figure 7.13, the value does not change much with varying temperature. It does, nonetheless, have an impact on the diffusion coefficient. A higher molar volume yields a lower diffusion coefficient, which in turn yields a lower wax deposition. This can easily be seen from the Wilke-Change Equation (6.3). As the molar volume decreases when the temperature decreases, it slightly alters the curvature of the deposition plot. If one looks at the plus fraction, i.e. the last value in Table 9.4, one can calculate that the wax molar volume is about 1970 $[cm^3/mol]$. Such a high value is not given by NeqSim.

8. n-Decane Thermodynamic Model Simulation Results

To easily show how the Thermodynamic Model works, one can do simulations for a simple fluid composition. This is done in this chapter, where n-decane (nC10) is used. The model predicts the minimum length needed to cool the fluid to near ambient seawater temperature, when there is no wax deposited. One can also see the Thermodynamic Model that is integrated into the Wax Model by viewing the temperature profile.

8.1 How to Use Code and Simulation Results

This section will explain how the Thermodynamic MATLAB scripts are to be used, and the results it outputs.

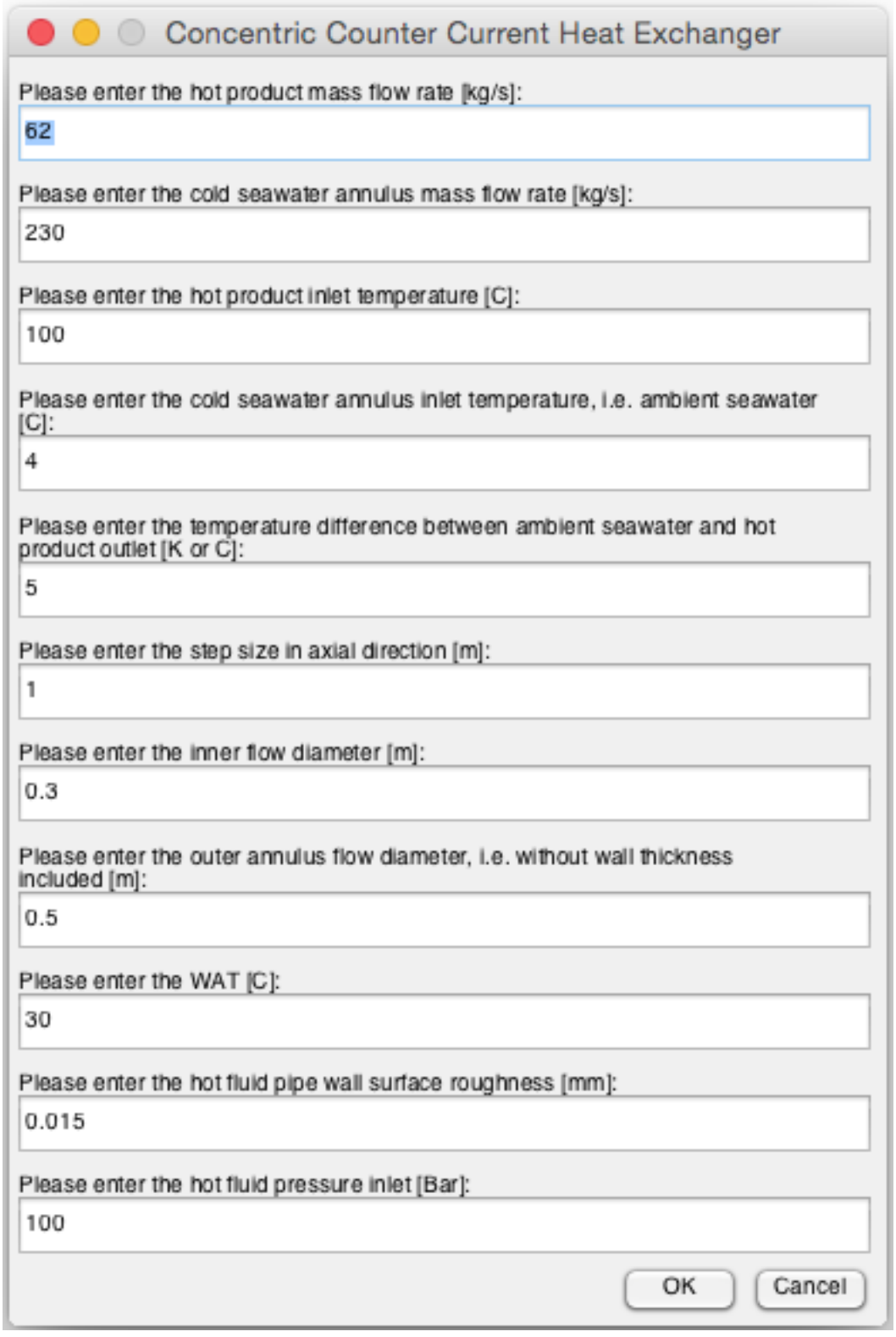


Figure 8.1: Dialogue box with its default values

Figure 8.1 shows the input dialogue box that is opened when the dialogue box script is run. The dialogue box contains default values, which can be edited by the user. The plots shown in Figure 8.2 and 8.3 is outputted when the default values shown in Figure 8.1 is run.

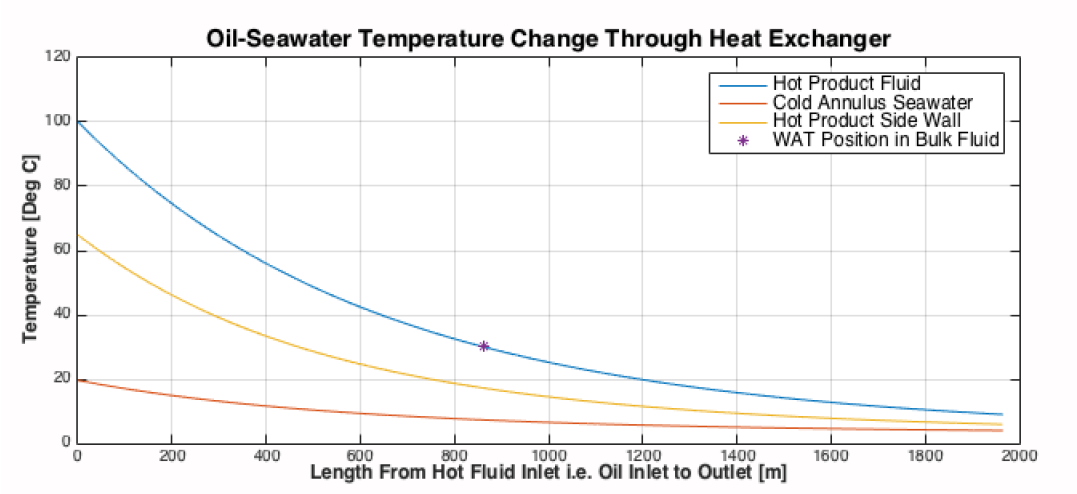


Figure 8.2: Temperature profiles and WAT position

The WAT, which is set to 30 $°C$ for the bulk product fluid, takes place at approximately 862 metres from the hot product inlet. The heat exchanger has to be approximately 1966 metres long in order to cool the hot product down to a temperature of 5 $°C$ above ambient seawater, i.e. 9°C for this case. MATLAB yields a Reynolds number, for the product flow, approximately between $2 \cdot 10^5$ and $6.5 \cdot 10^5$, which means the flow is turbulent (turbulent if $Re_D > 4000$).

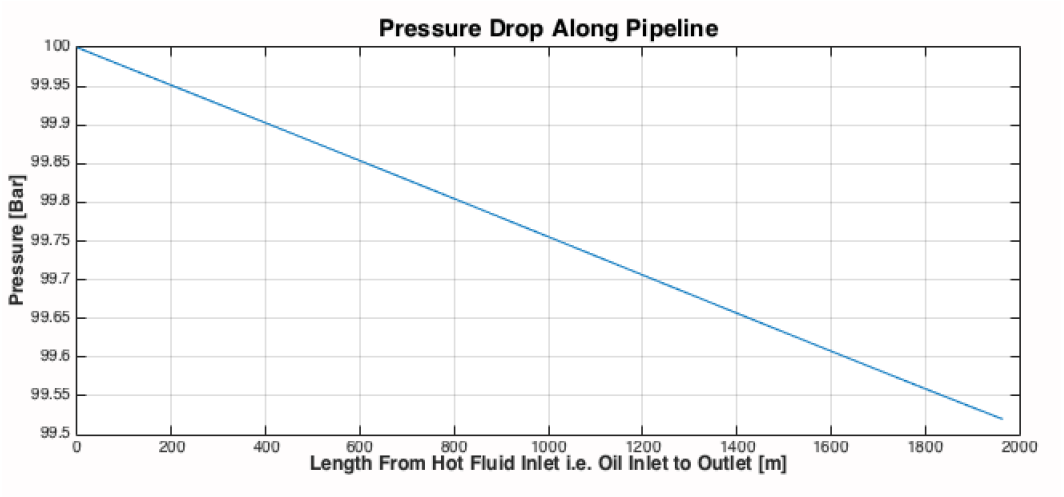


Figure 8.3: Pressure drop

As can be seen in Figure 8.3 the pressure drop is low. However, it is important to note that the calculations done do not factor in effects such as bends, valves, welding seams etc. The frictional losses will also increase when wax is deposited. The low pressure loss is negligible.

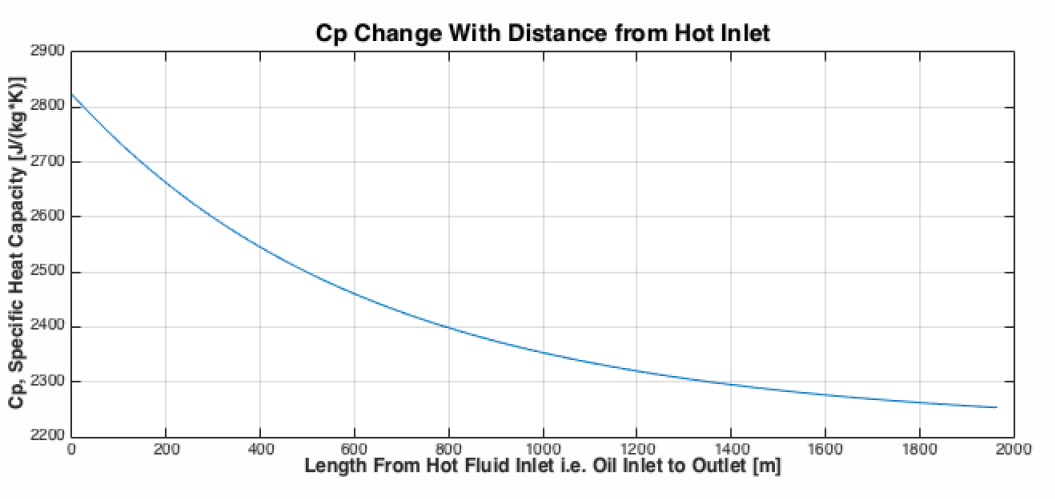


Figure 8.4: NeqSim varying the specific heat capacity for the hot product fluid

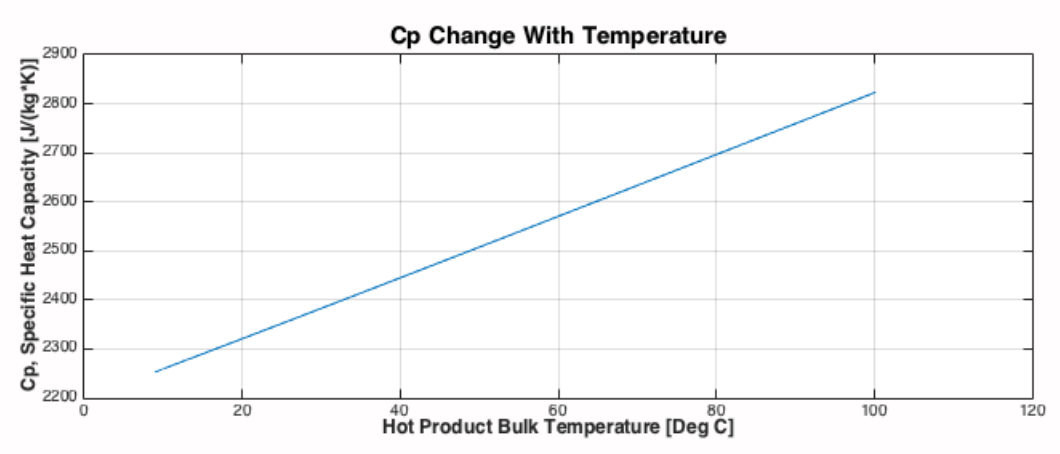


Figure 8.5: NeqSim varying the specific heat capacity for the hot product fluid

Figures 8.4 and 8.5 show how NeqSim varies the specific heat capacity, C_p, value with temperature for the nC10 fluid, i.e. the product fluid.

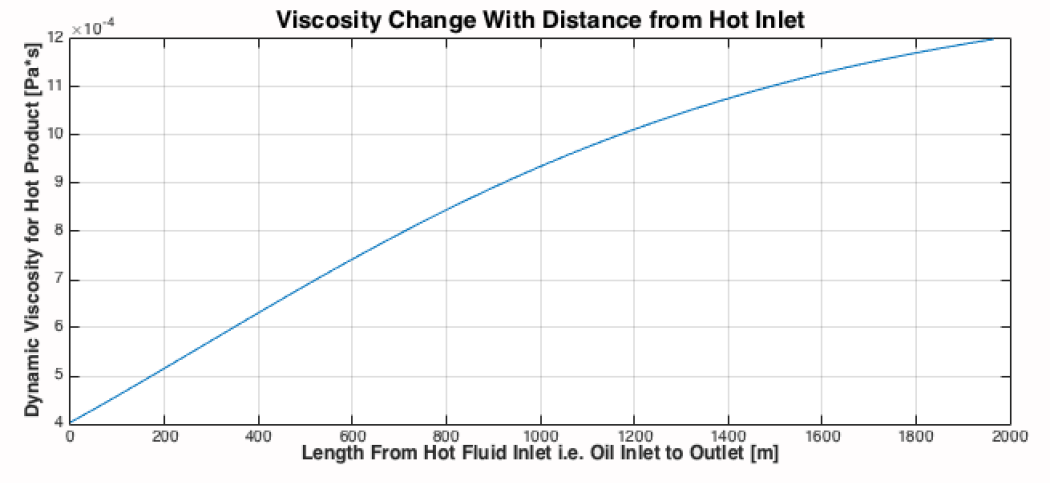


Figure 8.6: NeqSim varying the viscosity for the hot product fluid

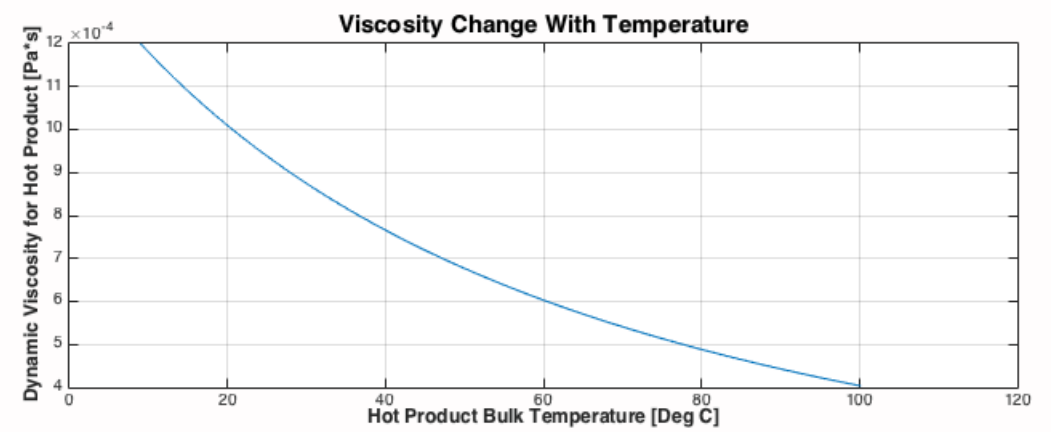


Figure 8.7: NeqSim varying the viscosity for the hot product fluid

Figures 8.6 and 8.7 show how NeqSim varies the viscosity value with temperature for the nC10 fluid, i.e. the product fluid.

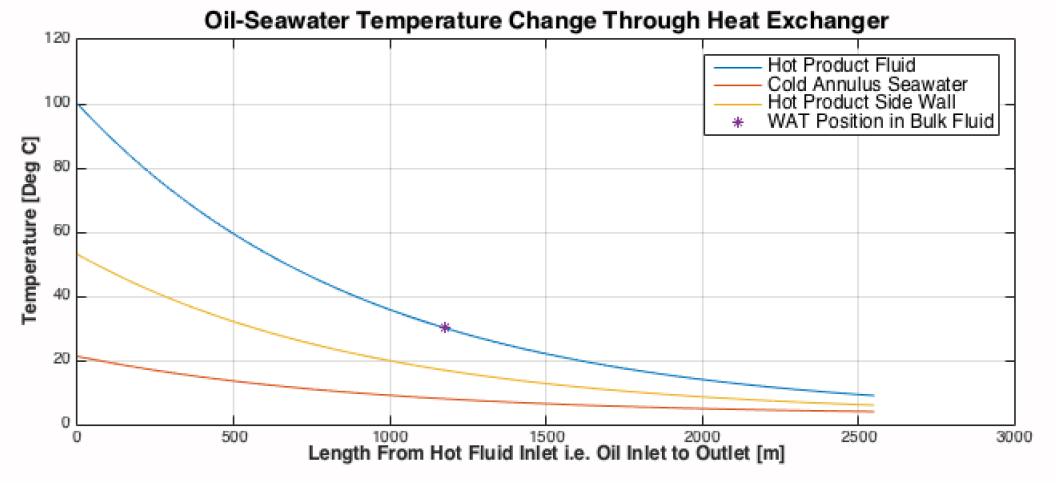


Figure 8.8: Temperature profiles for a constant specific heat capacity $C_p = 2800$ and $Viscosity = 12 \cdot 10^{-4}$

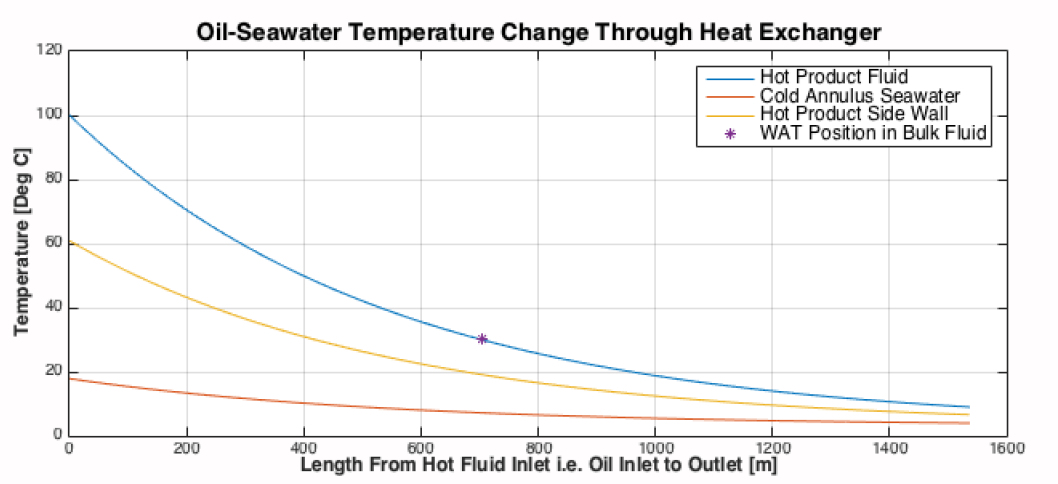


Figure 8.9: Temperature profiles for a constant specific heat capacity $C_p = 2250$ and $Viscosity = 4 \cdot 10^{-4}$

Figures 8.8 and 8.9 show the effects of removing NeqSim from the code and keeping the viscosity and specific heat capacity constant. The WAT position, and the heat exchanger length needed to cool the product fluid to the desired temperature changes. The two figures show two cases with constant specific heat capacity and viscosity. Figure 8.8 has the worst cooling properties and Figure 8.9 has the best cooling properties. The varying thermal conductivity for the product fluid can also have a big impact on the length needed to cool the fluid. Higher specific heat capacity means that more energy per kilogram is needed to warm the fluid one degree Kelvin, as compared to a lower specific heat capacity.

9. Point Model Simulations and Comparisons to Experiment

In this chapter Statoil experiments are presented and they are simulated using four different mathematical wax deposition models: Singh et al., RRR, Matzain and The Heat Analogy Model. All these models are discussed in Chapter 6. The purpose of the point model simulations are to find a good correlation between the little experimental data that is available and the simulations. The wax deposition model that yields the best coherence with the experimental data will be used to simulate the Subsea Model, i.e the full-scale model, presented in Chapter 4.

The four wax deposition model simulations are joined into four separate tables with important values such as mass flow rate, runtime (valve open time), simulated wax layer etc. Additionally a tuning factor, C, is added to tune the simulation to experimental values. All of the ten times four simulations were done with a fluid segment length of 50 metres, i.e. 100 times longer than the pipe itself. This was done in order to reduce simulation time.

The simulations are only done for the fluid composition given in Table 9.4. The reason for this is that it is difficult to get access to both fluid composition data and corresponding experimental deposition data values. Without deposition data, a model can not be verified. This is because one does not have data, for the given composition, to compare the simulations to. The program is, however, made so that one easily can add a new fluid composition to an existing excel file and do simulations for the new fluid.

9.1 Point Model Experiment

A vast amount of the experimental data is kept secret by companies. Most data is only presented in an incomplete manner. This has made it difficult to find data. Fortunately Statoil has shared some of its data. This can potentially be used to verify future models and the make-up of the fluid composition can be used for the simulations. Note: If any of the data presented in this chapter is to be published, permission should be granted by Statoil prior to publication.

The following data presented in the tables in this chapter are values for what is chosen to be called Fluid A. Fluid A has a WAT of $30.1°C$ and a wax dissolution temperature (WDT) of $43.0°C$. The WDT is defined as the temperature at which all precipitated wax

has been dissolved on heating the oil. Table 9.1 shows as expected decreasing density with increased temperature.

Table 9.1: Density Data

Temperature $[^{\circ}C]$	Density $[kg/m^3]$
5	836.9
10	825.8
20	817.6
40	802.7
60	788.0

Table 9.2 shows values for viscosity with changing temperature for a shear rate of 1000 $[1/s]$ and 500 $[1/s]$ respectively.

Table 9.2: Viscosity Data

Temperature $[°C]$	Pressure $[Bara]$	Shear rate $[1/s]$	Exp. Viscosity $[mPa \cdot s]$
60.00	1.52	1000	1.260
56.00	1.51	1000	1.340
50.00	1.47	1000	1.480
45.00	1.44	1000	1.610
40.00	1.43	1000	1.760
35.00	1.39	1000	1.940
33.00	1.38	1000	2.010
28.00	1.35	1000	2.220
25.00	1.33	1000	2.400
20.00	1.30	1000	2.710
15.00	1.28	1000	3.200
10.00	1.25	1000	4.240
5.04	1.23	1000	8.620
4.03	1.21	1000	10.500
28.00	1.32	500	2.260
25.00	1.32	500	2.420
20.00	1.30	500	2.790
15.00	1.27	500	3.290
10.10	1.25	500	4.130
5.05	1.22	500	6.680

Table 9.3 shows the amount of wax in weight per cent that has precipitated. The total wax content in the fluid is in this case 4.5 $wt\%$. Note that the value for concentration fraction, C, used in the Matzain Model, discussed in Section 6.6, is the dissolved wax fraction in the product fluid.

Table 9.3: Wax Precipitation Data, Total Wax Content 4.5 $wt\%$

Temperature		Precipitated Wax
$[°C]$	$[K]$	$[wt\%]$
32.5	305.5	0
30.0	303.0	0
27.5	300.5	0.00
25.0	298.0	0.05
22.5	295.5	0.10
20.0	293.0	0.21
17.5	290.5	0.40
15.0	288.0	0.77
12.5	285.5	1.21
10.0	283.0	1.66
7.5	280.5	2.10
5.0	278.0	2.49
2.5	275.5	2.86
0.0	273.0	3.20
-2.5	270.5	3.52
-5.0	268.0	3.79
-7.5	265.5	4.03
-10.0	263.0	4.22
-12.5	260.5	4.36
-20.0	253.0	4.50

Table 9.4: Gas Chromatography Analysis Data for Fluid A. Due to intellectual property rights the values are not shown.

Fraction [−]	wt% [−]	mole% [−]	MW $[g/mol]$	Density $[kg/m^3]$
C1	-	-	-	-
C2	-	-	-	-
C3	-	-	-	-
ISO-C4	-	-	-	-
N-C4	-	-	-	-
ISO-C5	-	-	-	-
N-C5	-	-	-	-
C6	-	-	-	-
C7	-	-	-	-
C8	-	-	-	-
C9	-	-	-	-
C10	-	-	-	-
C11	-	-	-	-
C12	-	-	-	-
C13	-	-	-	-
C14	-	-	-	-
C15	-	-	-	-
C16	-	-	-	-
C17	-	-	-	-
C18	-	-	-	-
C19	-	-	-	-
C20	-	-	-	-
C21	-	-	-	-
C22	-	-	-	-
C23	-	-	-	-
C24	-	-	-	-
C25	-	-	-	-
C26	-	-	-	-
C27	-	-	-	-
C28	-	-	-	-
C29	-	-	-	-
C30	-	-	-	-
C31	-	-	-	-
C32	-	-	-	-
C33	-	-	-	-
C34	-	-	-	-
C35	-	-	-	-
C36+	-	-	-	-

The values shown in Table 9.5 are found by running experiments in a concentric heat exchanger. As shown in Figure 9.1 below, the oil is run in the internal pipe and water flows in the opposite direction in the annulus. This is similar to the subsea concentric heat exchanger discussed in Chapter 4.

Figure 9.1: Concentric countercurrent heat exchanger

The length of the pipe is 0.5 m. The diameters for the oil pipes are: ID: 52.5 mm, OD: 60.3 mm. The diameters for the water annulus flow are: ID: 102.3 mm, OD: 114.3 mm. The oil pipe thermal conductivity is 45 $[W/(mK)]$.

Table 9.5: Experimental Wax Deposition

Test $[-]$	T_{oil} $[°C]$	T_{water} $[°C]$	Q_{oil} $[m^3/hr]$	$\dot{m}_{oil}$ $[kg/s]$	Q_{water} $[m^3/hr]$	$\dot{m}_{water}$ $[kg/s]$	Time $[hr]$	Wax Deposit $[mm]$
1	20	10	25	5.68	15	4.17	65	0.53
2	20	10	21	4.70	15	4.17	100	0.65
3	20	10	15	3.41	15	4.17	100	0.82
4	20	10	10	2.27	15	4.17	100	1.05
5	20	10	5	1.14	15	4.17	100	1.81

As is expected we can see that there is more wax deposition as the oil volumetric flow rate is decreased. This is due to the fact that the test with lower volumetric flow rate has a lower velocity, causing the fluid to cool more. This in turn makes the fluid deposit more wax.

9.2 Fluid Property Values

The model contains several uncertain fluid property values. Changing these values may severely affect the result of the deposition. Values such as heat capacity, Cp, is given by NeqSim and may not be accurate. As long as the values are within reasonable range it should not greatly affect the deposition values.

Values such as viscosity can change the shear stripping value. A lower viscosity increases the Π_2 value in the Matzain and Heat analogy wax model, see Equation (6.25). I.e. a lower viscosity increases the shear stripping.

9.2.1 Reynolds Number

The Reynolds number given by the simulation is approximately $4 \cdot 10^4$ and $8 \cdot 10^3$ for Test 1 and Test 5 respectively, see Table 9.5. This means that the flow is turbulent (turbulent if Re_D >4000). It is important that the flow is turbulent, so that there is adequate amount of mixing and the temperature in the fluid becomes more uniform. This makes the use of the bulk temperature in the calculations more accurate.

9.2.2 Diffusion Coefficient

The diffusion coefficient, D_{AB}, for the Test/Run 1 simulation yields about $2.7 \cdot 10^{-10}$ m^2/s. The value is not experimentally found, but Hayduk-Minhas correlation, Equation (6.4), is rather used to find the values. The Coefficient is a factor in the wax deposition thickness calculation and has a big impact on the deposition values. The diffusion coefficient of the waxy components in the oil typically ranges from 10^{-10} to 10^{-9} m^2/s[25]. A credible source at Statoil has also confirmed that the values for the coefficient is within the correct range.

9.3 How to Run The Wax Model

Similarly to what is described in Chapter 8, the user inputs values in the dialogue box shown in the figure below or uses the default values. As can be seen there are a few new input values, e.g. the number of fluid segments desired and its length, which indirectly corresponds to the elapsed time. It is also possible to not restrain the amount of fluid passing through the pipe, but rather stop the code when a certain wax deposition thickness is reached, however, this is chosen not to be done. One has more control over the program when the user chooses the simulation's resolution i.e. the number of segments.

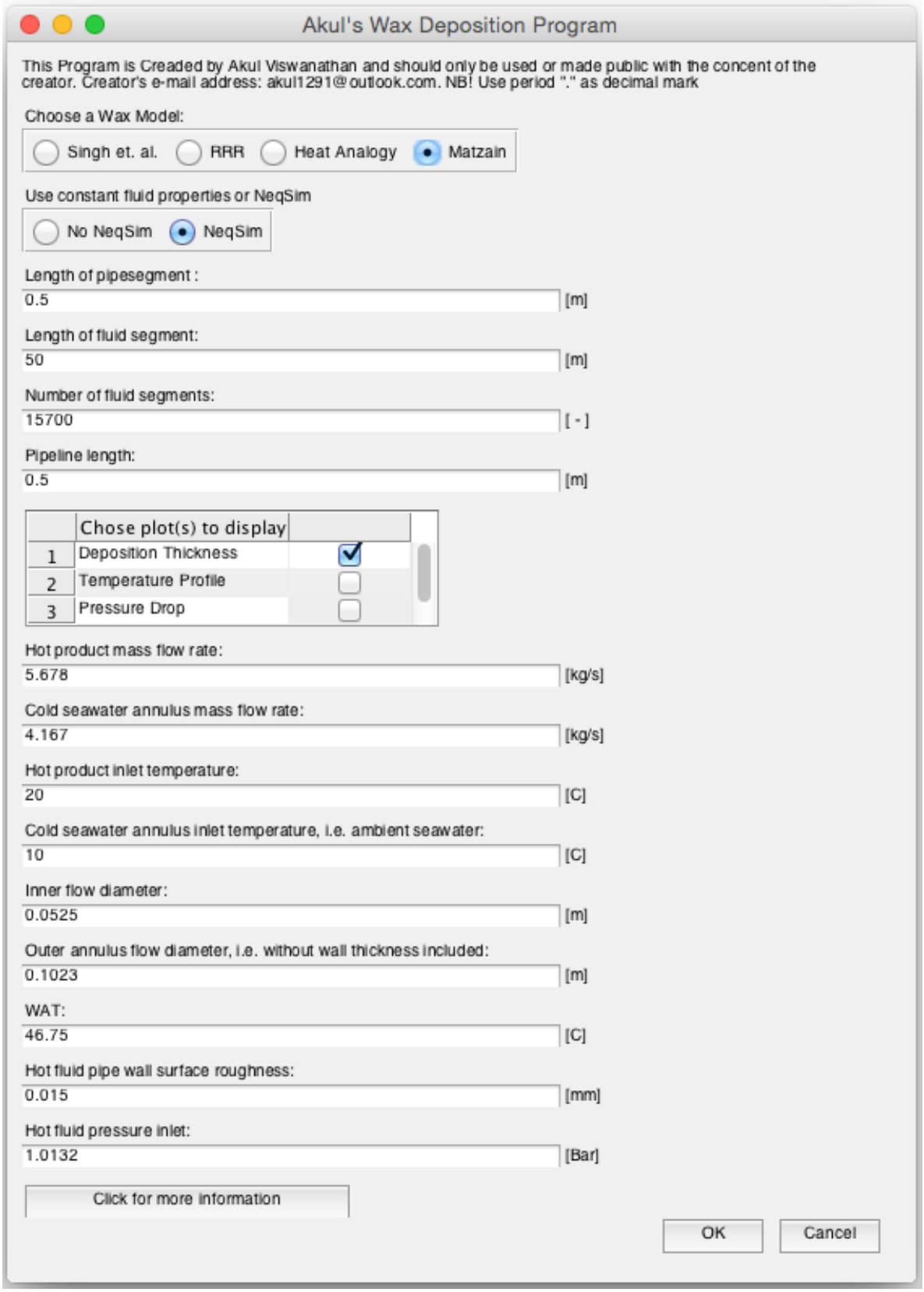


Figure 9.2: MATLAB Point Model dialogue box with default values

As can be seen in the figure above, one can choose between four different wax deposition models by clicking on a radio button: Singh et al., RRR, Heat Analogy or the Matzain model. One can also choose between using NeqSim or not using it. Additionally there is a section where one can choose which plots to display after the simulation has completed.

In order to calculate the number of fluid segments to input, when the time and fluid segment flow through time are known is given by:

$$Number\ of\ Fluid\ Segments = \frac{Valve\ Open\ Time\ [s]}{Fluid\ Segment\ Flow\ Through\ Time\ [s]} \tag{9.1}$$

The fluid segment flow through time can easily be found by running the simulation for a few fluid segments, and reading the output value $Time\_seg\_to\_next\_FLUIDseg$ in the MATLAB code. This value corresponds to the time it takes a fluid segment to pass through the inlet of the pipe segment.

9.4 Model Tuning

None of the models presented reproduce wax deposition, for varying conditions, accurately. It is, therefore, possible to tune the deposition values to better fit the experimental data. There are several ways to tune the wax deposition models. The models are chosen to be tuned by multiplying the deposition equation by a correction factor, C. This will in fact be the same as multiplying the diffusivity coefficient by the correction factor.

Run 1, which corresponds to the test 1 experimental values in Table 9.5, is tuned. The reason this is chosen to be the basis for the tuning is the fact that the Reynolds number is closest to the Subsea model's Reynolds number. The Reynolds number can be used to compare scaled and non-scaled values for fluid dynamic problems, and can be used to determine dynamic similitude between two different cases.

9.5 Singh et al. and RRR Point Simulation

This section presents the Singh et al. and the RRR Point model Simulation results. The findings are discussed below the tables and figures. The reason Singh et al. and the RRR wax models are put in the same section is that the simulation results can be explained in a similar manner.

The tables below show ten different simulation results using the Singh et al. wax model and ten separate ones for the RRR model. The Run number corresponds to the Test number in the experiments conducted by Statoil, see Table 9.5.

Table 9.6: Singh et al. Point Wax Deposition Simulation

Run [-]	Fluid Seg. [-]	Valve Open Time [hr]	$\dot{m}_{oil}$ [kg/s]	$\dot{m}_{water}$ [kg/s]	C [-]	Wax Layer [mm]	Experimental Wax Layer [mm]
1.0	$1.57 \cdot 10^6$	65	5.68	4.17	1.00000	BLOCKED	0.53
1.1	$1.57 \cdot 10^6$	65	5.68	4.17	0.00055	0.53	0.53
2.0	$2.00 \cdot 10^6$	100	4.70	4.17	1.00000	BLOCKED	0.65
2.1	$2.00 \cdot 10^6$	100	4.70	4.17	0.00055	0.68	0.65
3.0	$1.45 \cdot 10^6$	100	3.41	4.17	1.00000	BLOCKED	0.82
3.1	$1.45 \cdot 10^6$	100	3.41	4.17	0.00055	0.66	0.82
4.0	$9.65 \cdot 10^5$	100	2.27	4.17	1.00000	BLOCKED	1.05
4.1	$9.65 \cdot 10^5$	100	2.27	4.17	0.00055	0.62	1.05
5.0	$4.85 \cdot 10^5$	100	1.14	4.17	1.00000	BLOCKED	1.81
5.1	$4.85 \cdot 10^5$	100	1.14	4.17	0.00055	0.49	1.81

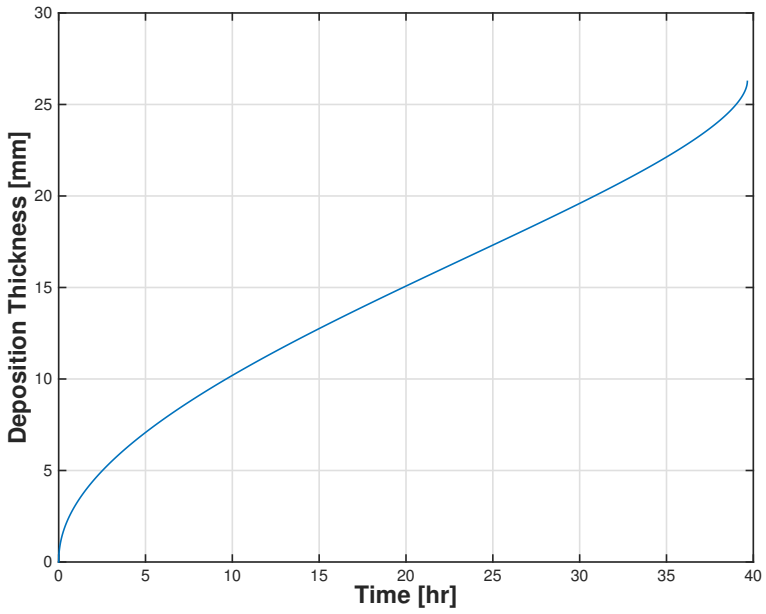


Figure 9.3: Run 1.0 Singh et al. Point Simulation, 50 metres fluid segment length. Note that the pipe has been completely blocked.

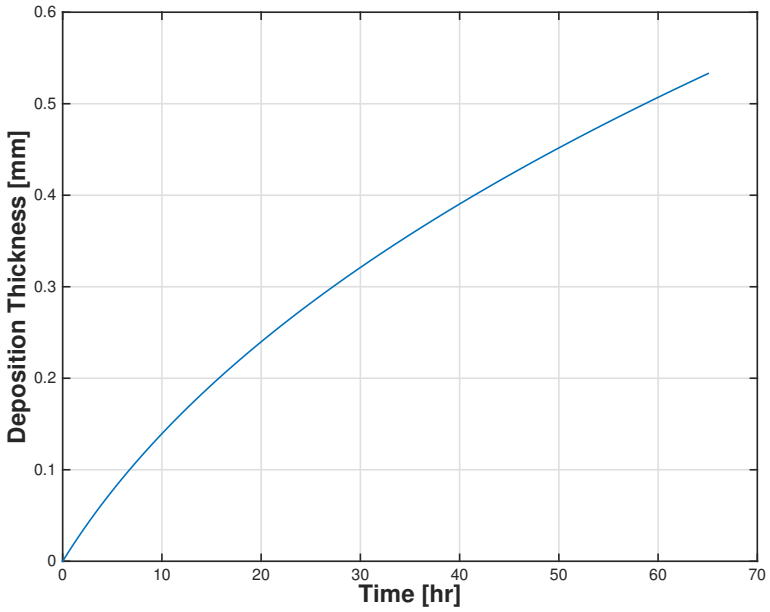


Figure 9.4: Run 1.1 Singh et al. Point Simulation, 50 metres fluid segment length

Table 9.7: RRR Point Wax Deposition Simulation

Run [-]	Fluid Seg. [-]	Valve Open Time [hr]	$\dot{m}_{oil}$ [kg/s]	$\dot{m}_{water}$ [kg/s]	C [-]	Wax Layer [mm]	Experimental Wax Layer [mm]
1.0	$1.57 \cdot 10^6$	65	5.68	4.17	1.000	7.74	0.53
1.1	$1.57 \cdot 10^6$	65	5.68	4.17	0.006	0.53	0.53
2.0	$2.00 \cdot 10^6$	100	4.70	4.17	1.000	9.71	0.65
2.1	$2.00 \cdot 10^6$	100	4.70	4.17	0.006	0.67	0.65
3.0	$1.45 \cdot 10^6$	100	3.41	4.17	1.000	9.65	0.82
3.1	$1.45 \cdot 10^6$	100	3.41	4.17	0.006	0.64	0.82
4.0	$9.65 \cdot 10^5$	100	2.27	4.17	1.000	9.61	1.05
4.1	$9.65 \cdot 10^5$	100	2.27	4.17	0.006	0.58	1.05
5.0	$4.85 \cdot 10^5$	100	1.14	4.17	1.000	9.58	1.81
5.1	$4.85 \cdot 10^5$	100	1.14	4.17	0.006	0.44	1.81

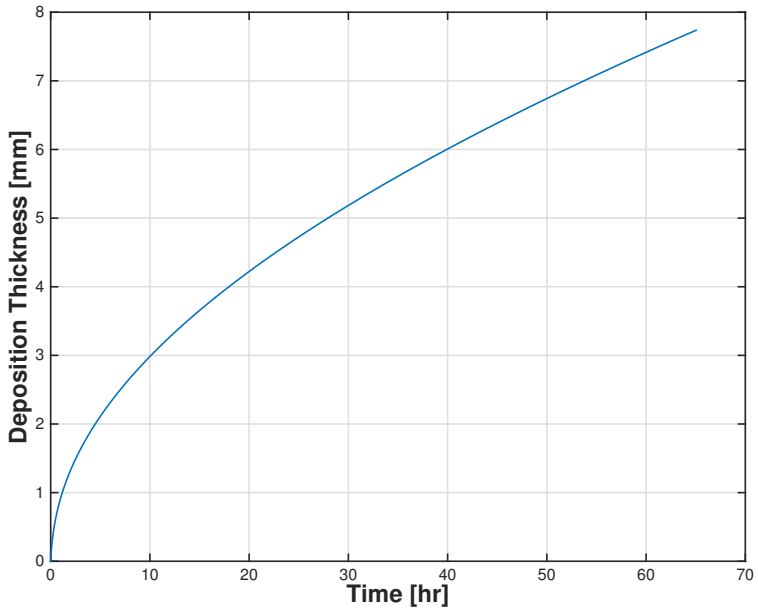


Figure 9.5: Run 1.0 RRR Point Simulation, 50 metres fluid segment length

Note that the fluid segment numbers, given in the tables above, are for a fluid segment length equal to the pipe segment length. In this case this is 0.5 m. This means that if the fluid segment length is 100 times longer, i.e. 50 m, then the values have to be divided by 100 in order to obtain the same valve open time.

As can be seen in Table 9.6 the five values that does not have a tuning factor simulates a blocked pipe. This means that the Singh et al. model severely overpredicts the amount of deposited wax. The RRR model does not simulate a blocked pipe, but also overpredicts the amount of deposited wax. It is highly unlikely that the tuned Singh et al. and RRR wax deposition values match experimental values for different valve open times, for a given Run. This is, among other things, because there are no shear stripping effects taken into account. All of the figures in this section would have had a very different shape had this been added.

For the Singh et al. model an extremely high correction factor has to be added to Run 1.0 in order to obtain the correct wax deposition compared to the experimental value. When the correction factor is added to Run 1.1 then this correction factor will be correcting too much for a case with a lower mass flow rate. Due to the fact of there not being shear stripping effects present. E.g. for Run 3.0 there is more wax deposited compared to Run 1.0. This is because the fluid is cooled more when the flow is slower. However, when Run 1 is used as the basis for the tuning, then as there is no shear stripping the tuning factor will be too low. This will make the simulation output a lower wax layer for lower mass flows. This trend can be seen in Table 9.6, and is the reason why the simulated wax layer consistently gets lower compared to the experimental wax

layer as the mass flow rate decreases. This is also the case for the RRR model, see Table 9.7. However, the simulation without a tuning does, unlike the Singh et al. model, not block the pipe. This also means that the correction factor has to be a lot lower.

Note that the creation of the Sing et al. model was done under laminar flow conditions [26]. These flow conditions are turbulent. One might think that the correction factor will have a linear relation to the deposited value. However, as can be seen this is not the case. When multiplying by e.g. 0.006 in the RRR simulations, one does not get the same value for wax deposition as if the non-tuned value is multiplied by 0.006. A reason is that each calculation impacts the next calculation, so that a big correction made may have less of an overall impact on the end wax layer thickness outcome. This is also applicable for the Matzain and Heat Analogy models, which are presented in the section below.

9.6 Matzain and Heat Analogy Point Simulation

This section presents the Matzain and the Heat Analogy Point model Simulation results. The findings are discussed below the tables and figures. The reason Matzain and the Heat Analogy wax models are put in the same section is that the simulation results can be explained in a similar manner.

Table 9.8: Matzain Point Wax Deposition Simulation

Run	Fluid Seg.	Valve Open Time	$\dot{m}_{oil}$	$\dot{m}_{water}$	C	Wax Layer	Experimental Wax Layer
[-]	[-]	[hr]	[kg/s]	[kg/s]	[-]	[mm]	[mm]
1.0	$1.57 \cdot 10^6$	65	5.68	4.17	1.000	1.88	0.53
1.1	$1.57 \cdot 10^6$	65	5.68	4.17	0.013	0.53	0.53
2.0	$2.00 \cdot 10^6$	100	4.70	4.17	1.000	2.30	0.65
2.1	$2.00 \cdot 10^6$	100	4.70	4.17	0.013	0.67	0.65
3.0	$1.45 \cdot 10^6$	100	3.41	4.17	1.000	2.66	0.82
3.1	$1.45 \cdot 10^6$	100	3.41	4.17	0.013	0.78	0.82
4.0	$9.65 \cdot 10^5$	100	2.27	4.17	1.000	3.19	1.05
4.1	$9.65 \cdot 10^5$	100	2.27	4.17	0.013	0.93	1.05
5.0	$4.85 \cdot 10^5$	100	1.14	4.17	1.000	4.30	1.81
5.1	$4.85 \cdot 10^5$	100	1.14	4.17	0.013	1.27	1.81

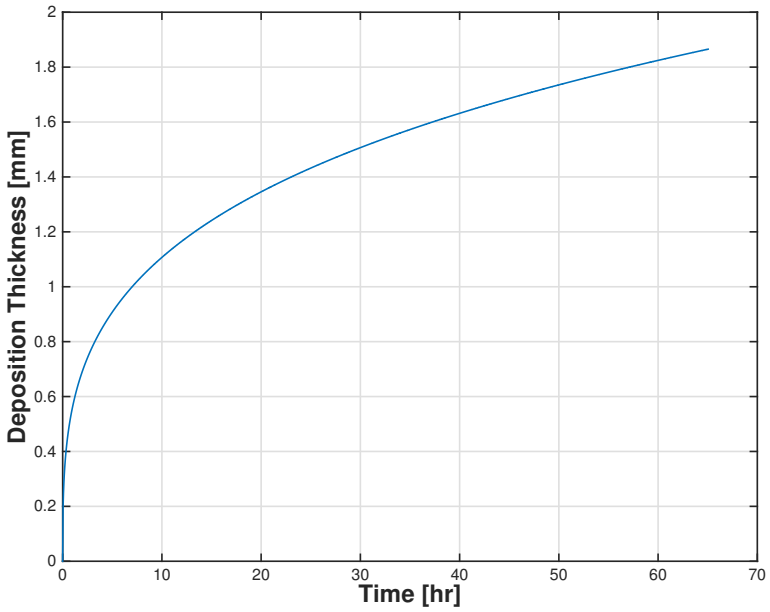


Figure 9.6: Run 1.0 Matzain Point Simulation, 0.5 metres fluid segment length

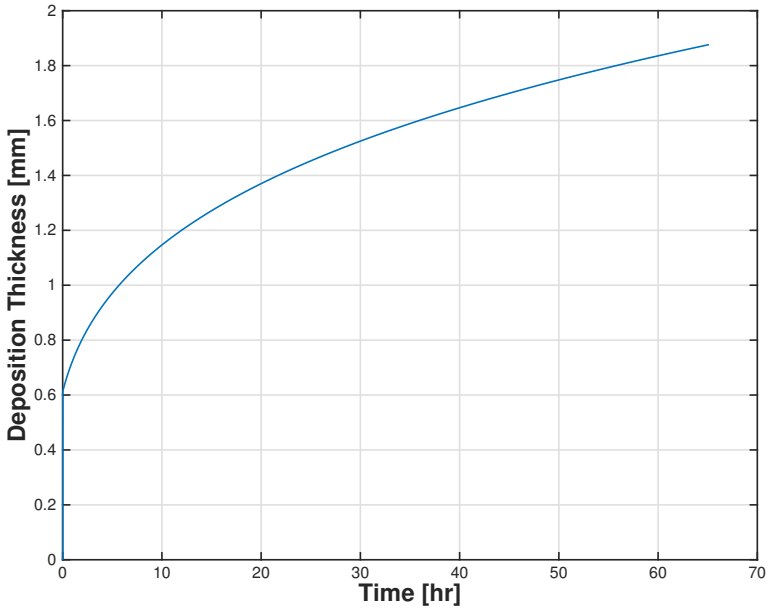


Figure 9.7: Run 1.0 Matzain Point Simulation, 50 metres fluid segment length

Table 9.9: Heat Analogy Point Wax Deposition Simulation

Run [-]	Fluid Seg. [-]	Valve Open Time [hr]	$\dot{m}_{oil}$ [kg/s]	$\dot{m}_{water}$ [kg/s]	C [-]	Wax Layer [mm]	Experimental Wax Layer [mm]
1.0	$1.57 \cdot 10^6$	65	5.68	4.17	1.000	1.47	0.53
1.1	$1.57 \cdot 10^6$	65	5.68	4.17	0.034	0.53	0.53
2.0	$2.00 \cdot 10^6$	100	4.70	4.17	1.000	1.81	0.65
2.1	$2.00 \cdot 10^6$	100	4.70	4.17	0.034	0.67	0.65
3.0	$1.45 \cdot 10^6$	100	3.41	4.17	1.000	2.09	0.82
3.1	$1.45 \cdot 10^6$	100	3.41	4.17	0.034	0.77	0.82
4.0	$9.65 \cdot 10^5$	100	2.27	4.17	1.000	2.51	1.05
4.1	$9.65 \cdot 10^5$	100	2.27	4.17	0.034	0.93	1.05
5.0	$4.85 \cdot 10^5$	100	1.14	4.17	1.000	3.39	1.81
5.1	$4.85 \cdot 10^5$	100	1.14	4.17	0.034	1.24	1.81

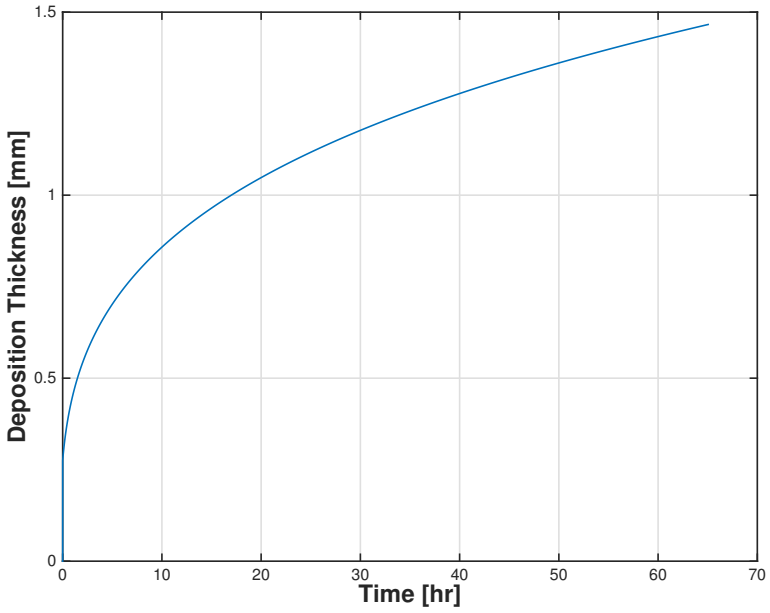


Figure 9.8: Run 1.0 Heat Analogy Point Simulation, 50 metres fluid segment length

As can be seen in Figure 9.6 and 9.7, changing the fluid segment length does not alter the deposition thickness much after 65 hours . The difference in thickness between the

two values is about 0.5%. The main difference between the two models is the computing power needed. For the 0.5 metre fluid segment simulation it takes about an hour to simulate and for the 50 metre fluid segment it takes about three minutes. This time will vary according to computational power, but gives a good indication to why choosing a longer fluid segment is advantageous. For shorter time scales, on the other hand, this time saving will not be as apparent and it will be less accurate. Additionally, as can be seen from Figure 9.7, there is a jump from the first fluid segment, where the deposition is zero, to the second fluid segment, where the deposition is about 0.6 mm. This is because the wax reducing does not take effect before there is some wax deposited. When the fluid segment length is set to be 50 metres then the first update to the simulation that there has been wax deposited, is after the initial 50 metres fluid segment has passed.

If the C1 constant in the Matzain model, Equation (6.24) which is an empirical relation for the rate enhancement due to trapped oil, is changed from 15 to 1. Run 1.0 deposition value changes to 0.87 mm. This shows us that it is also possible to tune the Matzain and Heat Analogy models by changing one or more of the empirical constants discussed in Section 6.6. The C1 value is set to 1 as default for the Heat Analogy model. This makes the wax thickness jump for the first deposition, in Figure 9.8, less noticeable.

The Matzain and Heat Analogy models are, like the models in the previous section, tuned on the basis of Run 1.0. It is therefore, interesting to see whether the experimental wax layer values match the other tuned simulation cases. One can see from the two Tables above that the two model's simulations do in fact match the experimental values quite well, unlike the Singh et al. and the RRR model discussed in the section above. There is an under prediction when the flow rate gets lower, this may be due to the shear stripping effects being too low. This is similar to what was discussed in the previous chapter regarding lower simulated wax deposition values for lower flow rates.

9.7 Model Comparison and Selection

An objective for this chapter is to find the best suited wax deposition model for simulation of the controlled wax deposition unit model. The graphs shown in the figures below are some of the simulations from the two consecutive sections above.

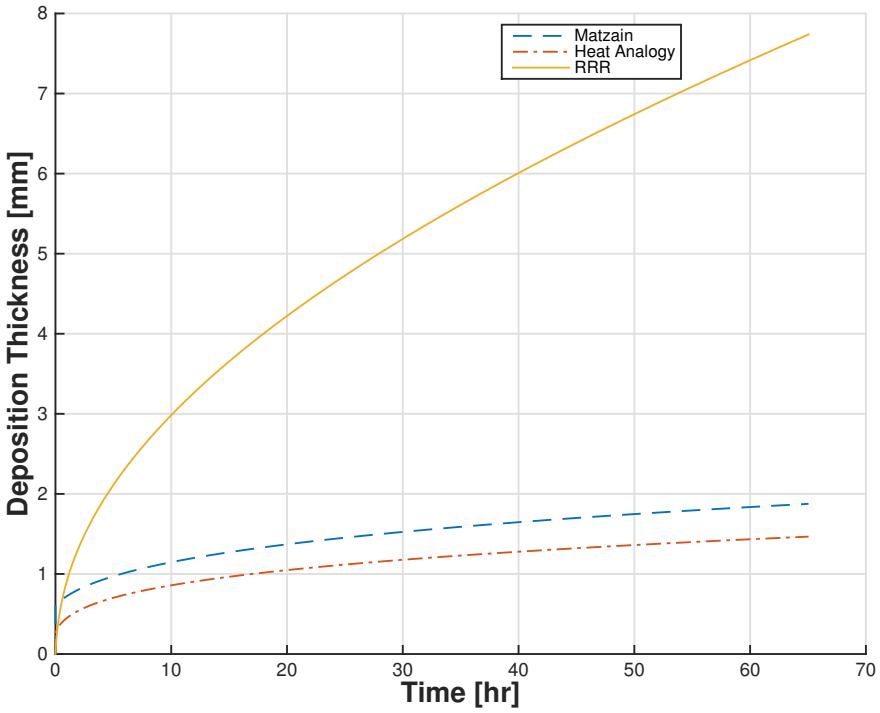


Figure 9.9: The Matzain, Heat Analogy and RRR's Run 1.0 (no tuning) simulations

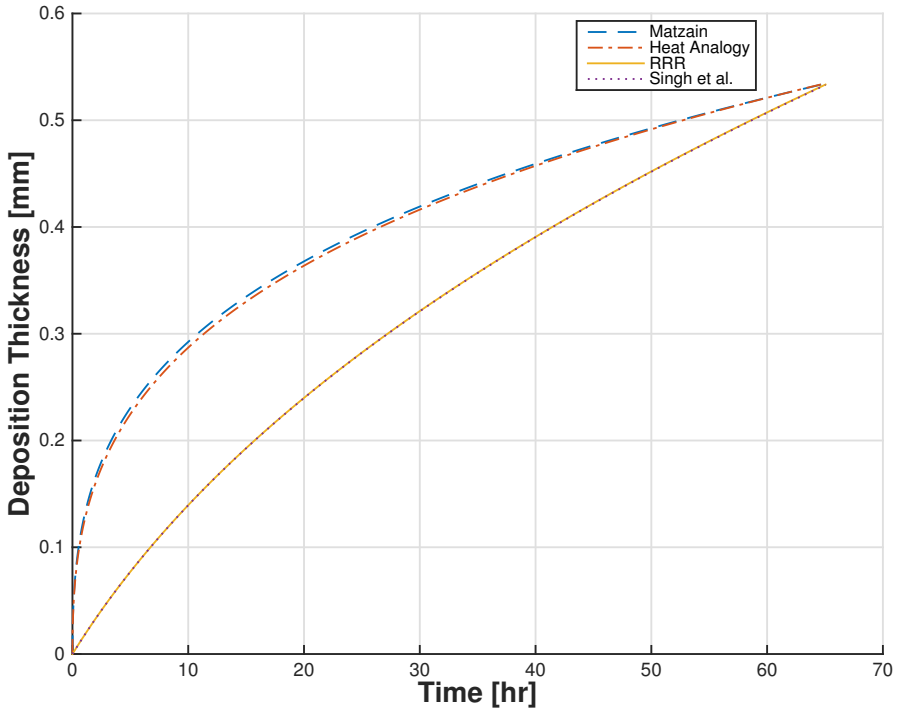


Figure 9.10: The Matzain, Heat Analogy, RRR and Singh's Run 1.1 (with tuning) simulations

Figure 9.9 clearly shows that the RRR model overpredicts the wax deposition a lot compared to the Matzain and the Heat Analogy models. This is due to the lack of shear stripping in the RRR model.

Figure 9.10 shows the tuned deposition values for the four different wax models. As these values are tuned to fit the end point it is obvious that this point will be correct. However, as can be seen the path the models take are not the same. Unfortunately only a value for each experiment is known, so it is not possible to know if other values for a different time than the end point match the experiment. Ideally one could have tuned the deposition model to fit the experimental values to more than one point. As this is not possible due to lack of data, this is not done. It is, however, possible to look at the tuned model results for wax deposition when changing fluid conditions, i.e. the various runs.

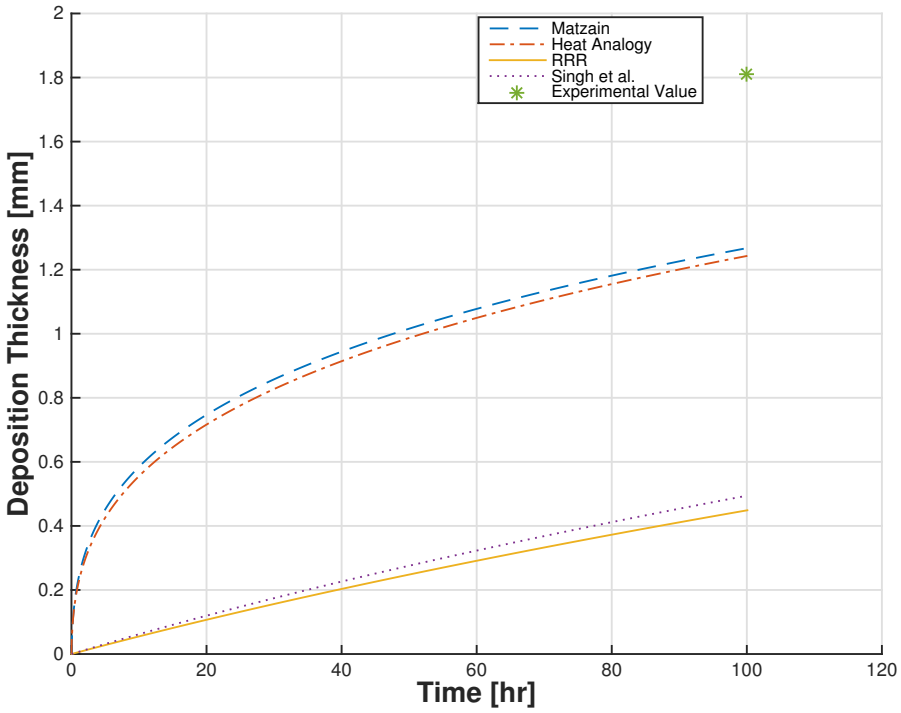


Figure 9.11: The Matzain, Heat Analogy, RRR and Singh's Run 5.1 (with tuning) simulations in one plot

Figure 9.11 shows tuned deposition simulation values for a new flow condition. Note that the tuning is done on the basis of Run 1.0. One can see that the simulation that is closest to the experimental results are the Matzain and Heat Analogy model. However, the error is quite substantial even for the best results.

There are a few graphs of experimental results conducted at Statoil, Porsgrunn. Unfortunately the fluid composition is unknown. A direct comparison can, therefore, not be made. However, a look at the curvature of the graph will give some insight as to how the simulation's graph curvature should look.

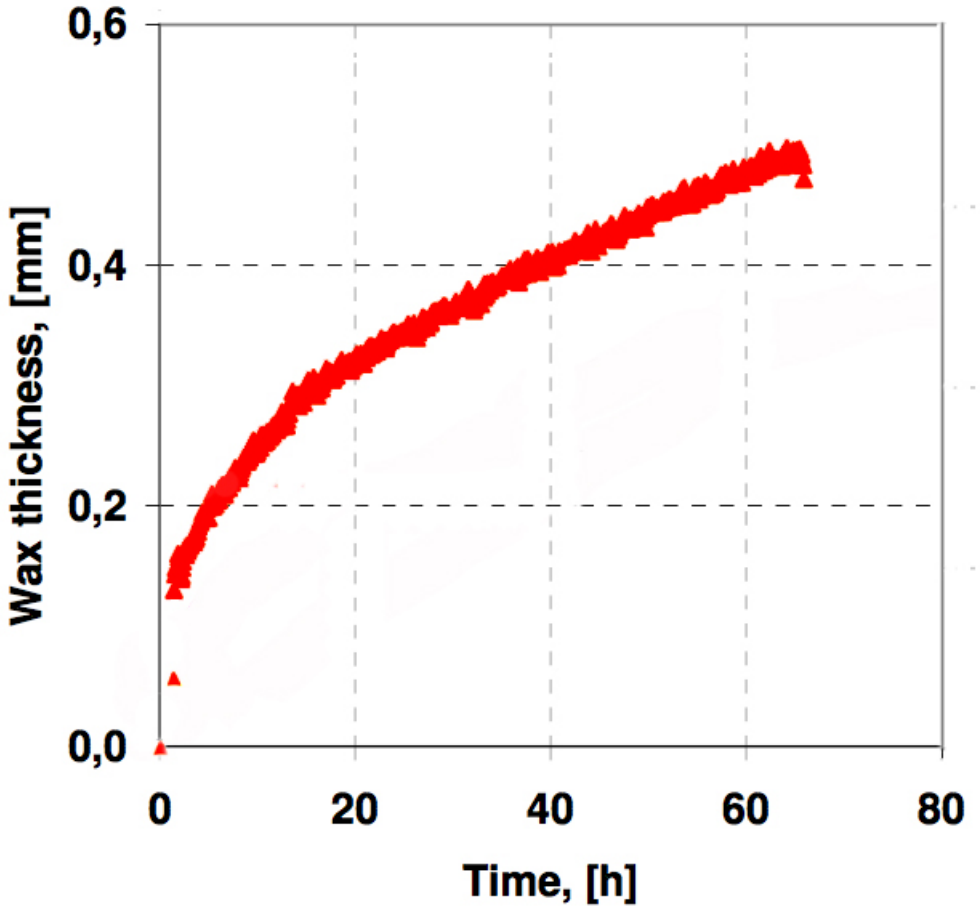


Figure 9.12: Experimental Data from Statoil, Porsgrunn. With Fluid values similar to Run 1 [24]

Looking at Figure 9.10, one can see that the curvature of the Matzain and Heat Analogy model looks most similar to the curve in 9.12.

From all the gathered information one can make the conclusion that the the best simulation results are produced when using the Matzain or Heat Analogy model. A note can also be made on the fact that one can see that the values for the various models are systematically higher when simulated. By comparing to Rosvold's OLGA results and other papers. However, what is most important going forward is, if the model follows the experimental values when changing the fluid values and for varying time. Both the Matzain and the Heat Analogy model does this to a certain degree.

10. Subsea Controlled Wax Deposition Unit Simulations

Due to the findings in the previous chapter the tuned Matzain wax model is chosen to be used for the next simulations. In this chapter wax simulations for the subsea controlled wax deposition unit (CWDU), discussed in Chapter 4, is further discussed. As has been previously discussed the CWDU simulation are comprised of what essentially are several point models put together. The temperature range and, therefore, also the fluid properties changes more. This makes the Reynolds number vary more, which slightly lowers the tuning accuracy. This is due to the fact that simulation is tuned for specific flow conditions. However, with the experimental values available this is the best and most accurate tuning which can be done. Another possibility is to change the CWDU model's flow values to better coincide with the experimental values. This is chosen not to be done.

10.1 Unit Location

The CWDU is located on the seabed. To gain a better understanding of where it is positioned, both in terms of the process and physically, the figure below was made. The image was put together with three different images[34, 35, 36].

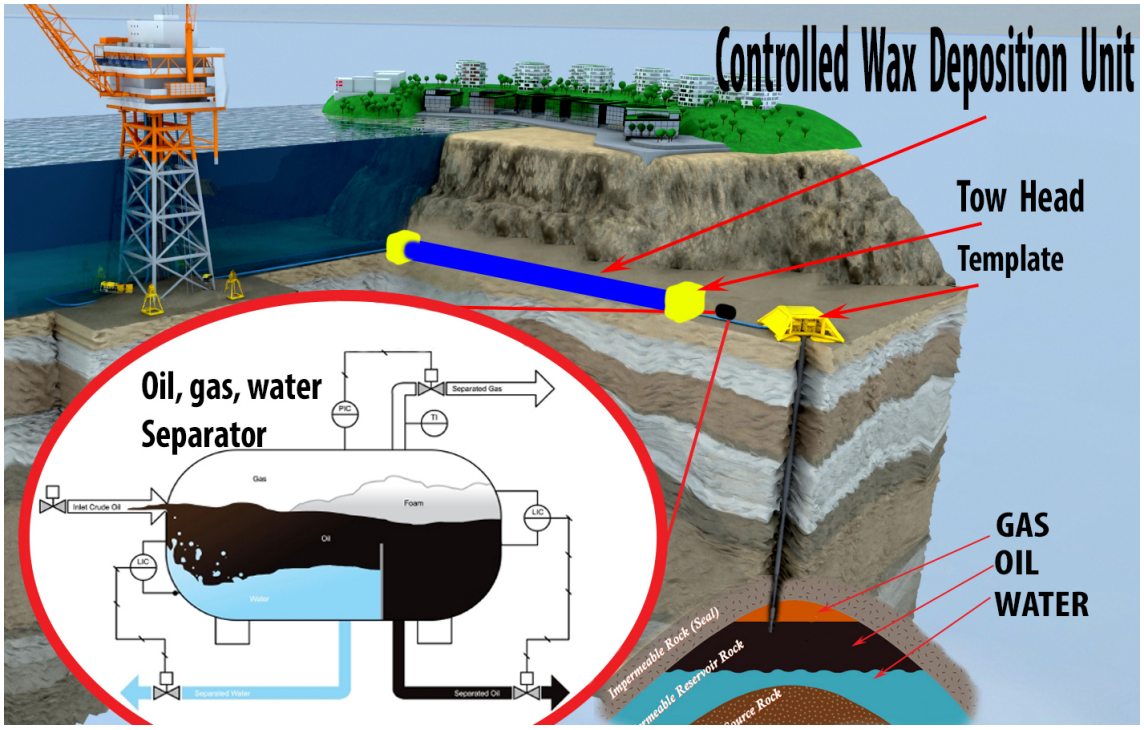


Figure 10.1: Placement of various subsea components

It is ideal that most of the water and gas is separated out of the fluid. This avoids multiphase flow in the CWDU. Gas will also have a bad effect on the cooling performance in the unit. The water and gas can possibly be reinjected into the reservoir, lowering the reservoir's pressure drop. The product which exits the CWDU can be transported at ambient temperature, also known as cold flow, to top side. At topside the precipitated wax can be separated and removed.

10.2 Fluid Property Values

What is described in Section 9.2 is also applicable for the CWDU simulations. However, for these simulations there is a greater temperature range and the fluid properties vary more.

The Reynolds number is in the range $4 \cdot 10^4$ to $4 \cdot 10^5$ and is closest to the point model simulation Run 1's Reynolds number, which has a Reynolds number $4 \cdot 10^4$. Ideally experiments with a higher Reynolds number should also have been available.

The diffusion coefficient is in the range $1.5 \cdot 10^{-10}$ to $1.3 \cdot 10^{-9}$ m^2/s. As mentioned in Section 9.2 this is not experimentally found, but is within the correct range.

10.3 How to Run The Wax Model

Note, this section contains similar information as Section 9.3, however, the default values shown in the figure below are different.

Similarly to what is described in Chapter 8, the user inputs the dialogue box shown in the figure below or uses the default values. As can be seen there are a few new input values, e.g. the number of fluid segments desired and its length, which indirectly corresponds to the elapsed time. It is also possible to not restrain the amount of fluid passing through the pipe, but rather stop the code when a certain wax deposition thickness is reached, however, this is chosen not to be done. One has more control over the program when the user chooses the simulation resolution i.e. the number of segments.

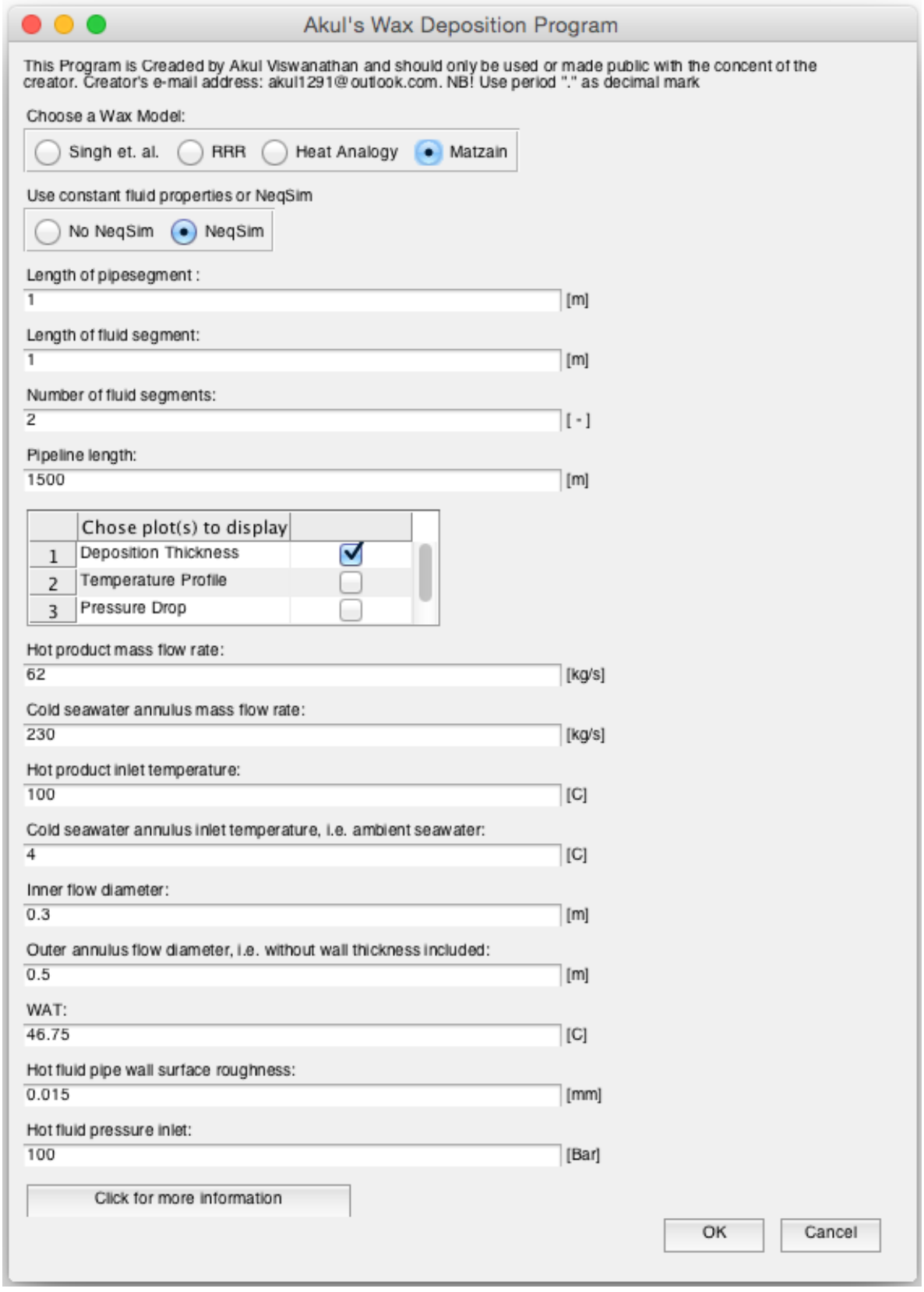


Figure 10.2: MATLAB dialogue box with default values

As can be seen in the figure above, one can choose between four different wax deposition models by clicking on a radio button: Singh et al., RRR, Heat Analogy or the Matzain model. One can also choose between using NeqSim or not using it. Additionally there is a section where one can choose which plots to display after the simulation has completed.

In order to calculate the number of fluid segments to input, when the time and fluid segment flow through time are known is given by:

$$Number\ of\ Fluid\ Segments = \frac{Valve\ Open\ Time\ [s]}{Fluid\ Segment\ Flow\ Through\ Time\ [s]} \tag{10.1}$$

The fluid segment flow through time can easily be found by running the simulation for few fluid segments, and reading the output value $Time\_seg\_to\_next\_FLUIDseg$ in the MATLAB code. This value corresponds to the time it takes a fluid segment to pass through the inlet of the pipe segment.

When the simulation is running a 'waitbar' runs to show how much of the simulation has completed, see the figure below. This is of great help when running simulations that take much time.

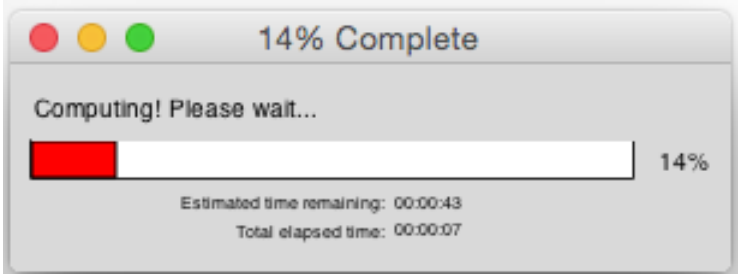


Figure 10.3: Waitbar, which shows how much of the simulation that has completed

10.4 Simulation

Various simulations are done by using the default values, which are shown in Figure 10.2. The pipe length was chosen to be 6000 m and the fluid segment number altered to fit the different elapsed times wanted. Below is a Figure of such a simulation, where each line represents 12 hours of elapsed time (vale open time). The conductivity for wax deposition is set to be a constant of 0.2 $W/(m \cdot K)$.

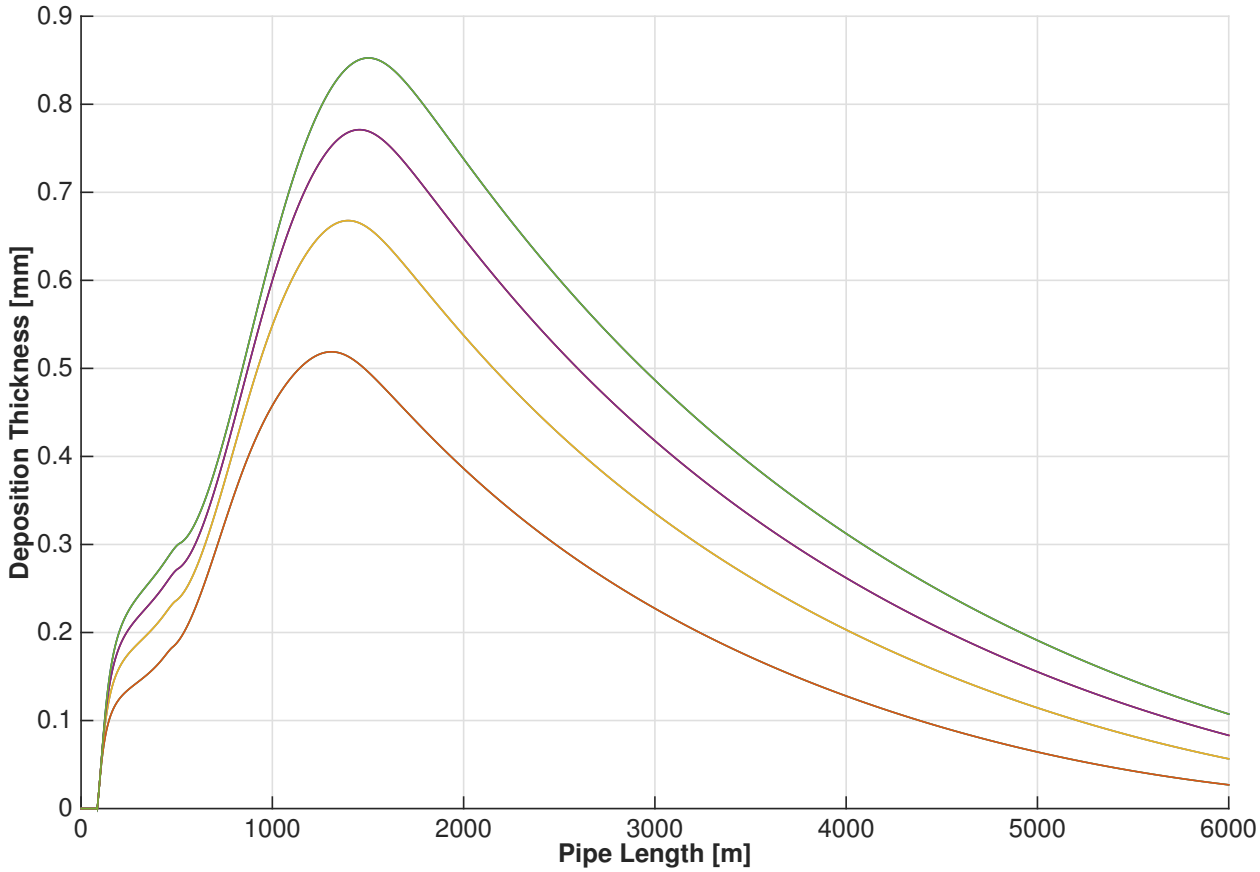


Figure 10.4: Subsea unit wax simulation. Each line represents 12 hours elapsed time.

The bottom most wax deposition line, for the figure above, is the first 12 hours after a pig has cleaned the CWDU, and each next line is an addition of 12 hours, without cleaning.

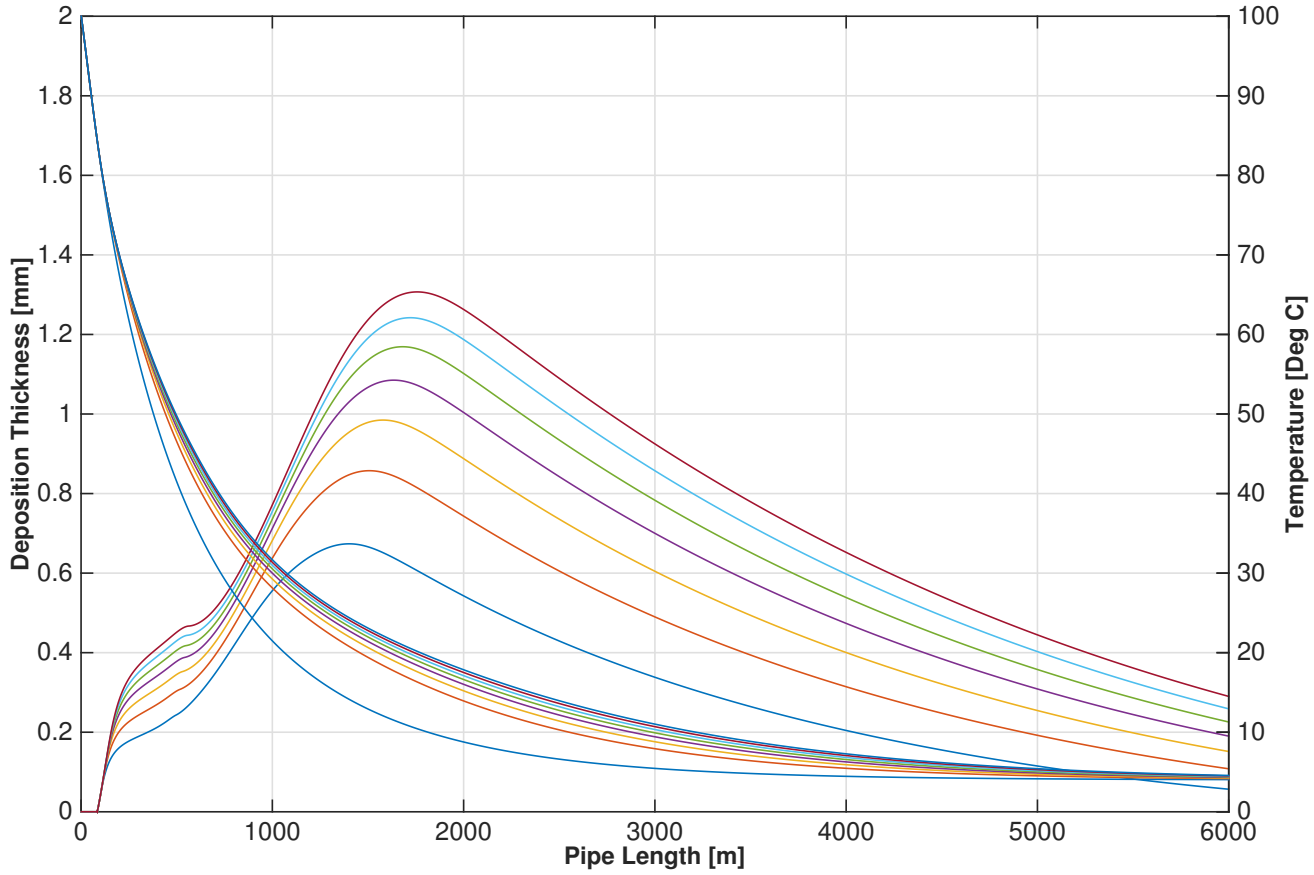


Figure 10.5: Subsea unit wax simulation. Each line represents 24 hours elapsed time. Both bulk temperature and wax layer is plotted.

The bulk temperature lines, in the figure above, is the line that starts at the top left. The inlet temperature is set to be 100°C. The bottom most temperature line is the Thermodynamic Model temperature profile, i.e. the temperature profile when there is no wax. By this line it is possible to see that the minimum length to cool the product fluid to near ambient temperature is about 3 km. The bottom most wax deposition line is the first 24 hours after a pig has cleaned the unit, and each next line is an addition of 24 hours. As can be seen the Figure 10.4's second line and Figure 10.5's first line which represents 24 hours elapsed time are the same. The simulation is chosen to be done for a 6000 m long pipeline. If one for instance wants a 4000 m pipe length then one can simply look at the graph from 0 to 4000 m.

The deposition plot shape is mainly influenced by the temperature profile and the wax precipitation values, shown in Figure 7.8. The measured WAT is between 27.5 and 25.0°C, this can be seen from Table 9.3. NeqSim on the other hand gives a small

amount of wax precipitation at high temperatures, which can be seen in Figure 7.8. The effect of this can be seen in Figure 10.5 where there is wax deposition at higher temperatures. However, it is important to note that the wall temperature is lower than the bulk temperature. The WAT given by NeqSim is about 47°C.

The reason the first 24 hours has such a higher wax deposition compared to the other 24 hour lines is due to the mathematics behind the Matzain model, more specifically the shear stripping. When there is more deposited wax then there is more shear stripping, i.e. in the beginning less wax is loosened, when there is less wax that has deposited. Hence, the bigger steps and more deposition, between the same time periods when less time has elapsed. How accurate this is, is difficult to conclude with the little experimental data that is available.

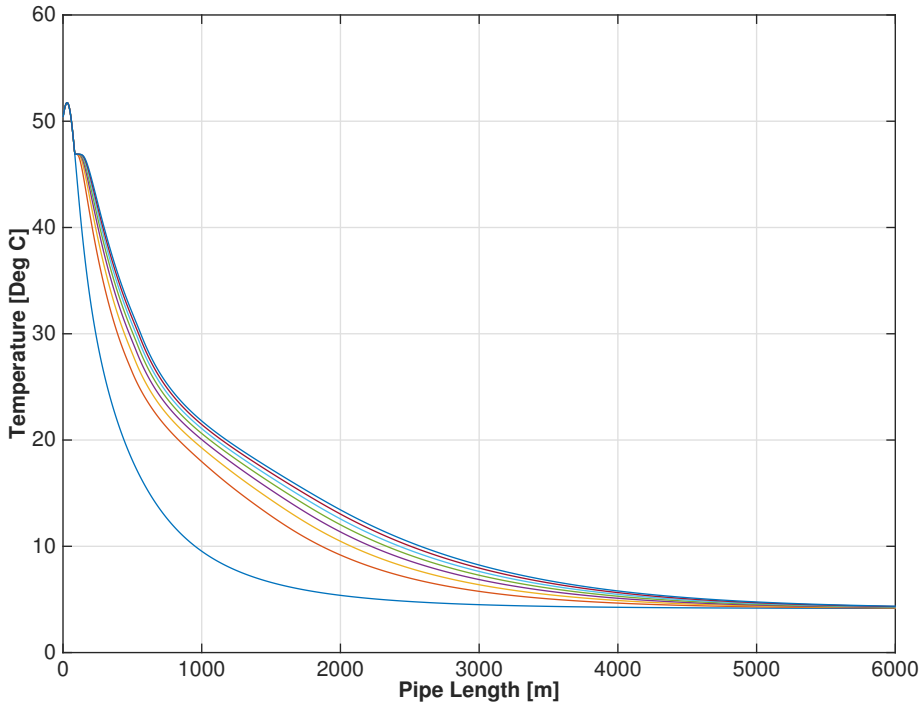


Figure 10.6: Subsea unit wall temperature. One week total elapsed time.

The lines in the figure above represent the prodcut fluid's wall temperature. The bottom most line is the wall temperature when there is no wax present. The next lines are each an additional elapsed time of 24 hours. The wall temperature is an important part of the wax deposition calculations. Because the temperature difference between the bulk and the wall is the reason for the wax concentration gradient in the pipe. This allows for transport of precipitated wax. The slight upwards movement of the wall temperature, on the graph above near the inlet, is caused by the slight inaccuracy of the regression of the viscosity given by NeqSim. This can be seen in Figure 7.9 where the blue line

goes slightly upwards between 80 and 100°C. This is, however, above WAT and will not impact the simulations much. It is also important to note the assumption that the wall temperature is not the same as the bulk at the inlet. The Figure below shows the viscosity values for the same one week simulation. Here the unfortunate aforementioned dip in viscosity can clearly be seen and the viscosity value's temperature dependency.

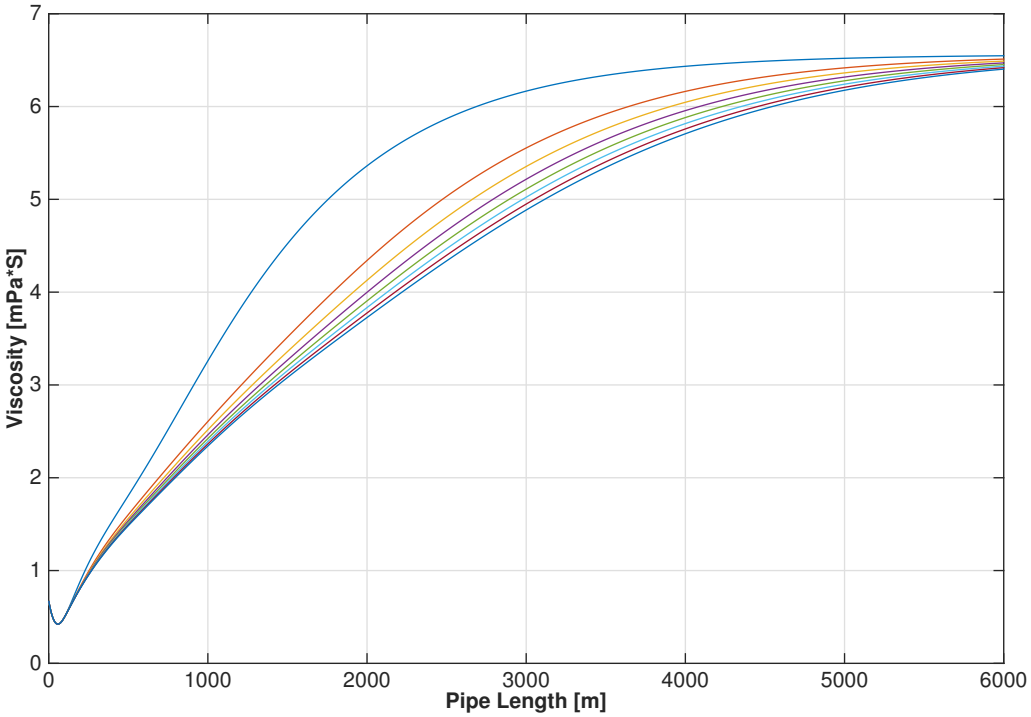


Figure 10.7: Viscosity one week wax simulation

The top most line in the figure above is the viscosity before any wax is deposited and each next line is 24 hours of run time.

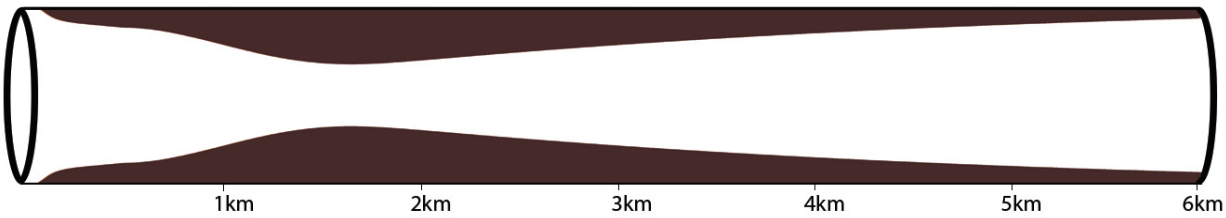


Figure 10.8: Magnified wax thickness, made using the 7 day deposition thickness curve. Thickest point being 1.3 mm.

To illustrate how the wax deposition will look. The deposition curve after an elapsed time of 7 days, was used to create the figure above. I.e. the top most red line in Figure 10.5. The thickest wax deposition point is about 1.3 mm. Here it is assumed that there is symmetric deposition, which will not be completely accurate due to gravity effects. However, it gives a good illustration as to what the deposition may look like.

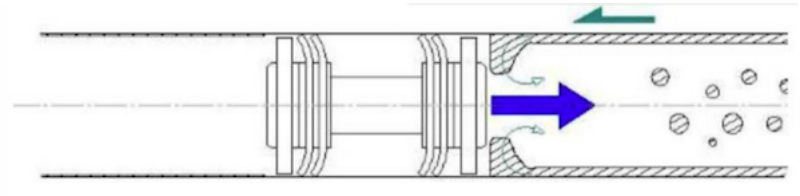


Figure 10.9: Wax removal with bypass pig[37]

The figure above shows a schematic of a bypass pig removing wax. This is what happens when the pig goes around in the loop and removes wax. It is assumed that the pig removes all of the wax. This means that the simulation values given by the program, is for the run time for one deposition cycle before a pig removes the wax.

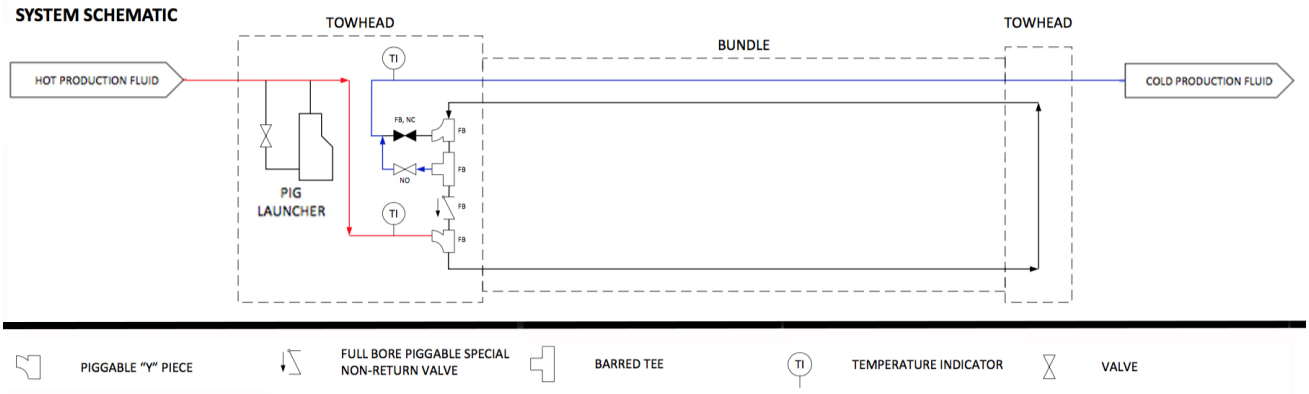


Figure 10.10: P&ID of proposed subsea unit [16]

What configuration is chosen for the CWDU does not change the simulation results. The P&ID shows a possible configuration. The red line is the inlet and the blue the outlet. This means that the red line is the zero position and the blue line e.g. 6 km point. If the pipe length of 6 km is desired, then the unit length will be approximately half of that. The proposed system shown in the P&ID schematic is a good configuration that utilises the pipe area well.

10.5 Weight of Precipitated and Deposited Wax

By Figure 7.8, it is possible to estimate the wt% range to approximately be 3wt%. This means that if the mass flow rate is 62 kg/s then 1.86 kg/s of this is wax that can

precipitate, some of which is deposited.

The mass of the total wax deposited after the simulated 7 days can also be calculated using the script given in Appendix A. This yields a mass of $1.2 \cdot 10^7$ kg, if the wax has a density of 909 kg/m^3, which is spread along the pipe as is shown in Figure 10.8.

10.6 Other Scenarios

There are several other scenarios other than the main one presented in this chapter. Some of these are briefly presented and discussed in the subsections below.

10.6.1 Different Length

As has been previously mentioned one can look at the simulation for a 6 km long pipeline and still have results for a e.g. 4 km long pipeline. Below is a figure that shows an area marked in a red box. These deposition lines illustrates some of the wax that will deposit outside the unit if the length is chosen to be 4 km. Rather than having a pigging frequency of 7 days, one should evaluate increasing the frequency. By looking at the figure below a pigging frequency of 2-3 days seems reasonable.

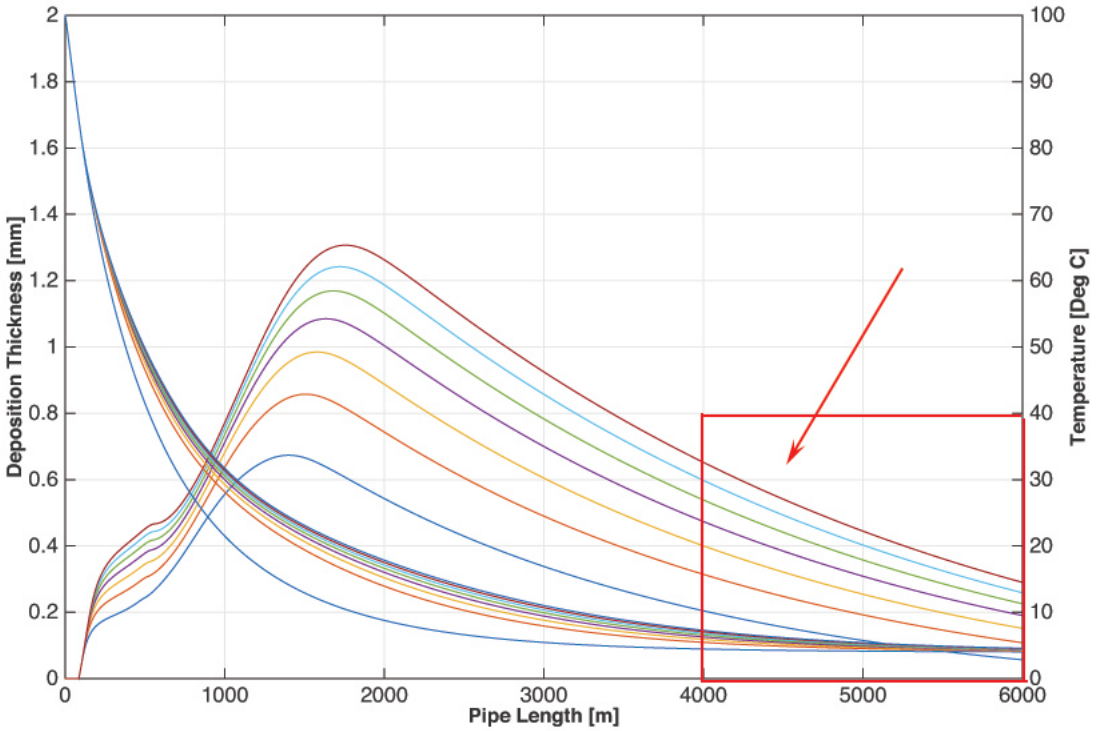


Figure 10.11: Subsea unit wax simulation. Each line represents 24 hours elapsed time. Both bulk temperature and wax layer is plotted.

10.6.2 Stuck Pig

The CWDU enables high pigging frequency in the unit itself and a lower one for the rest of the pipeline. This lowers the risk of a pig getting stuck outside the unit. On the other hand it increases the risk of a pig getting stuck in the unit. It could be possible to add an extra unit. In the event of a failure, the other unit can be used. The initial investments may be high, but the reliability will increase.

Another option is to have a pipeline that bypasses the unit. If a pig gets stuck in the unit then the flow can run through the bypass line. This means the fluid is not cooled to the same extent as in the deposition unit. A higher frequency on the entire pipeline pigging has to be started and kept until the problem is fixed.

10.6.3 Unable to Deploy Big

An option if a pig is unable to deploy is, as mentioned in the previous subsection, having another unit. Another option is to change the cooling fluid's mass flow rate to get a more even deposition. This also means that there will be more deposition outside the unit, but will give more time for troubleshooting and fixing the problem.

10.6.4 Shut-Down

In the case of a shut-down the remaining fluid in the unit will cool down. E.g. fluid at the inlet, which is at 100°C, will cool down to ambient temperature. For the simulation values presented in this chapter. The deposition for a one metre long stagnant fluid segment near the inlet is calculated by using the equation below:

$$WaxMass = Volume_{segment} \cdot 3wt\% \cdot \rho_{wax} \tag{10.2}$$

Here it is assumed that there is 3 wt% that can precipitate when the minimum temperature is about 4°C, and a wax density of 909 kg/m^3. This yields a value of only 1.9 kg of wax that will be spread over the one metre long, 0.3 m diameter pipe segment. This means that in the case of a shut down there will be a maximum addition of about 2.25 mm of wax thickness.

10.6.5 Higher Wax content

The fluid simulated in this thesis has a wax content of 4.5 wt%. This value may be a lot higher for a different fluid. The time it takes a pipe to travel from inlet to outlet is calculated using Equation 10.1. For a 6 km long pipeline it takes about 1 hour and 25 minutes. For a fluid containing a high amount of wax there might be a need for an additional pig so that the minimum cleaning cycle time is halved.

10.7 Pressure Drop

The diameter decreases while wax is deposited, this slightly alters the pressure drop. Studying the pressure drop equations gives a better understanding of what happens.

Writing Equation 6.7 again:

$$\Delta P = f_D \cdot \frac{L}{D} \cdot \frac{\rho u^2}{2} \tag{10.3}$$

And the mass flow rate correlation:

$$\dot{m} = \rho v A = \rho v \cdot \frac{\pi D^2}{4} \tag{10.4}$$

With some algebraic manipulation of Equation 10.4, solving for velocity, v, and combining this with Equation 10.3 yields:

$$\Delta P = f_D \cdot \frac{L}{D^5} \cdot \frac{16\dot{m}^2}{2\rho^2\pi^2 g} \tag{10.5}$$

As can be seen from Equation 10.5 the diameter is to the power of five. The mass flow rate is assumed to be constant. This means that a reduction in diameter due to wax deposition will have a big impact on the pressure drop. If the diameter decreases there will be a higher pressure drop. However, this is in fact not the case for these simulations. The pressure drop is lower when there is more wax deposited and when it is assumed that the wax layer does not alter the surface roughness. The reason for this is the changing temperature related to the wax deposition. Looking at Figure 10.5's temperature profile one can clearly see a temperature increase for a given section in the pipe as wax is deposited. This alters the fluid property values and in turn alters the frictional losses. E.g. when the temperature increases the viscosity decreases, which can be seen in Figure 10.7.

Additionally the deposited wax surface roughness also impacts the pressure drop. In the simulation it is chosen to be set to a constant surface roughness of ϵ =0.015 mm. A different roughness can be added to the wax. Pressure drop caused by things such as bends and welds are not a part of the calculations.

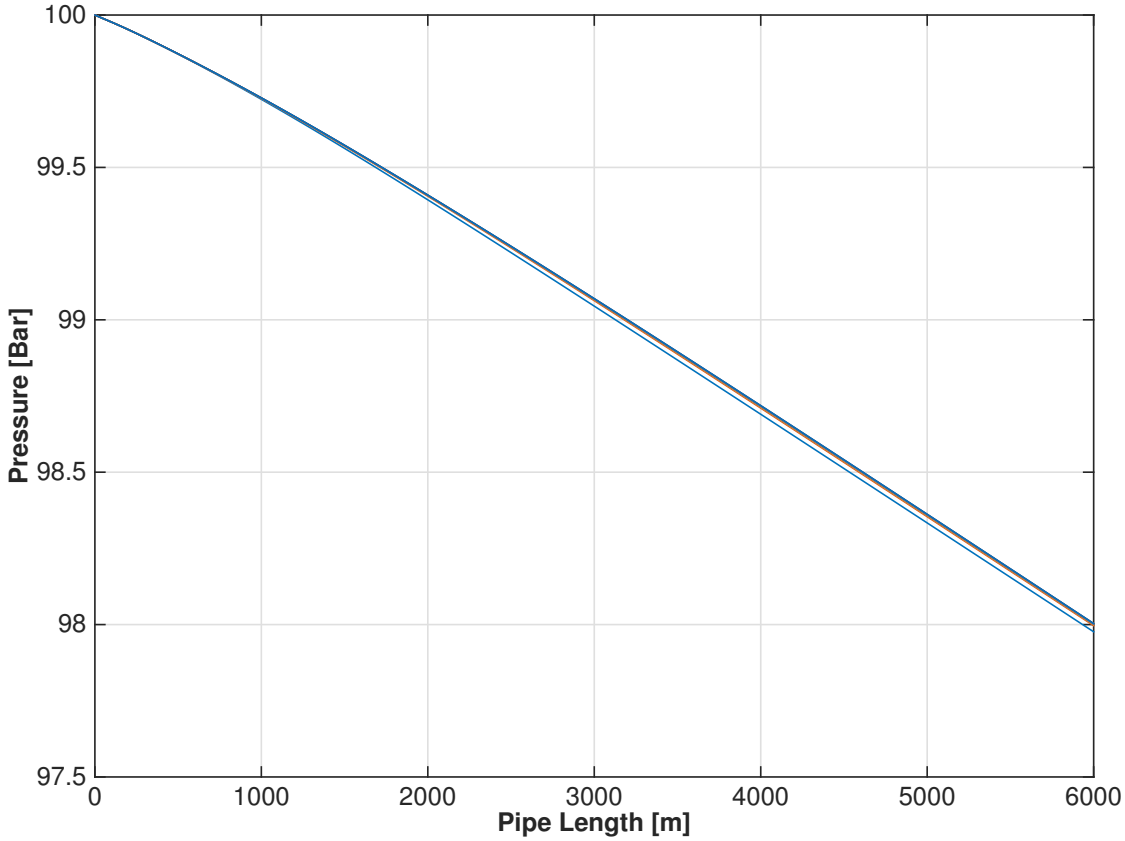


Figure 10.12: Pressure drop for wax deposition from 0 to 7 days

The bottom most line in Fiugre 10.12 represents the pressure drop with no wax deposition, each next line is an additional 24 hours of run time. With a total of 8 lines, i.e. the simulation is for a total elapsed time of 7 days. The figure below is the same plot, but zoomed in to more clearly show each line.

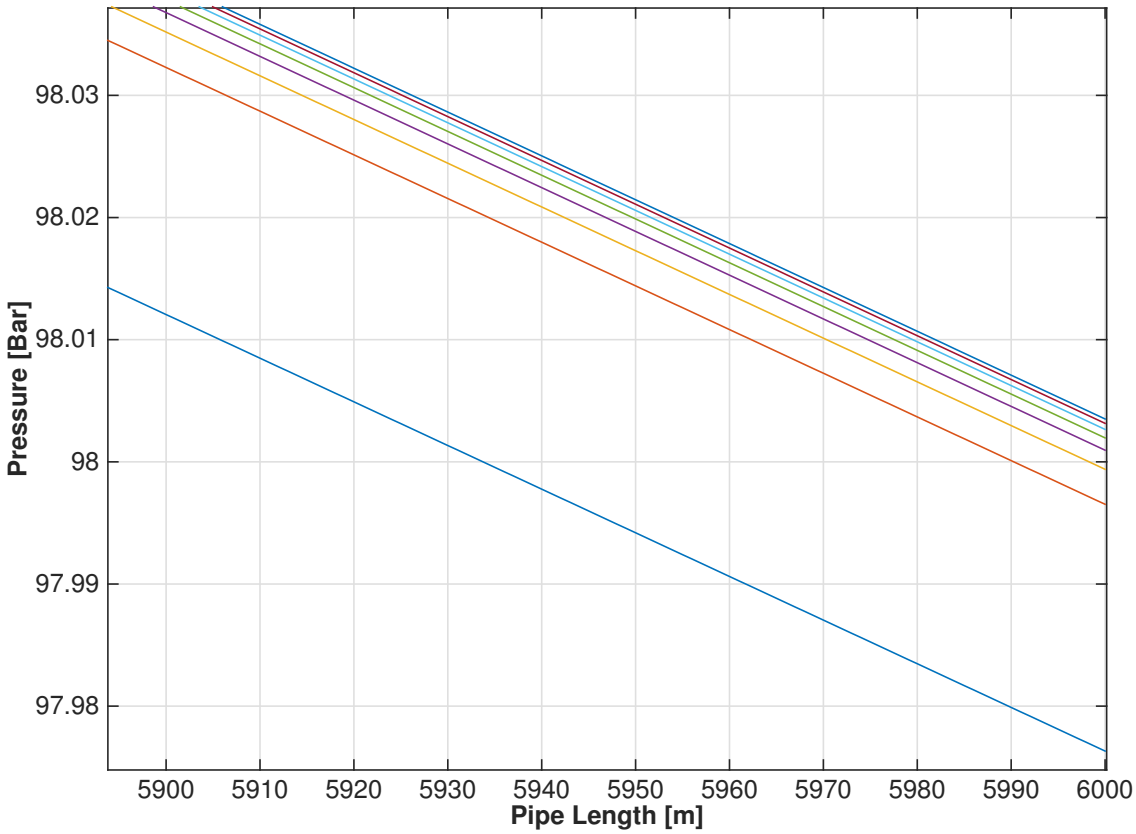


Figure 10.13: Pressure drop for wax deposition from 0 to 7 days zoomed in

As previously stated it is also possible to alter the surface roughness. The thickness of the wax is added to the existing surface roughness. This simulation has been done and the values are shown in the two figures below. The first top most line is the pressure drop when there is no wax present. Each next line (downwards) is an additional 24 hours without wax removal. The inclusion of wax thickness in the surface roughness has clearly impacted the pressure drop. It is uncertain whether simply adding the wax thickness to the surface roughness is accurate. However, this does not impact the deposition part of the simulations.

The code added to MATLAB is simply:
$epsi = 0.0015 * 10^{-3} + Depo\_thick(i, j);$

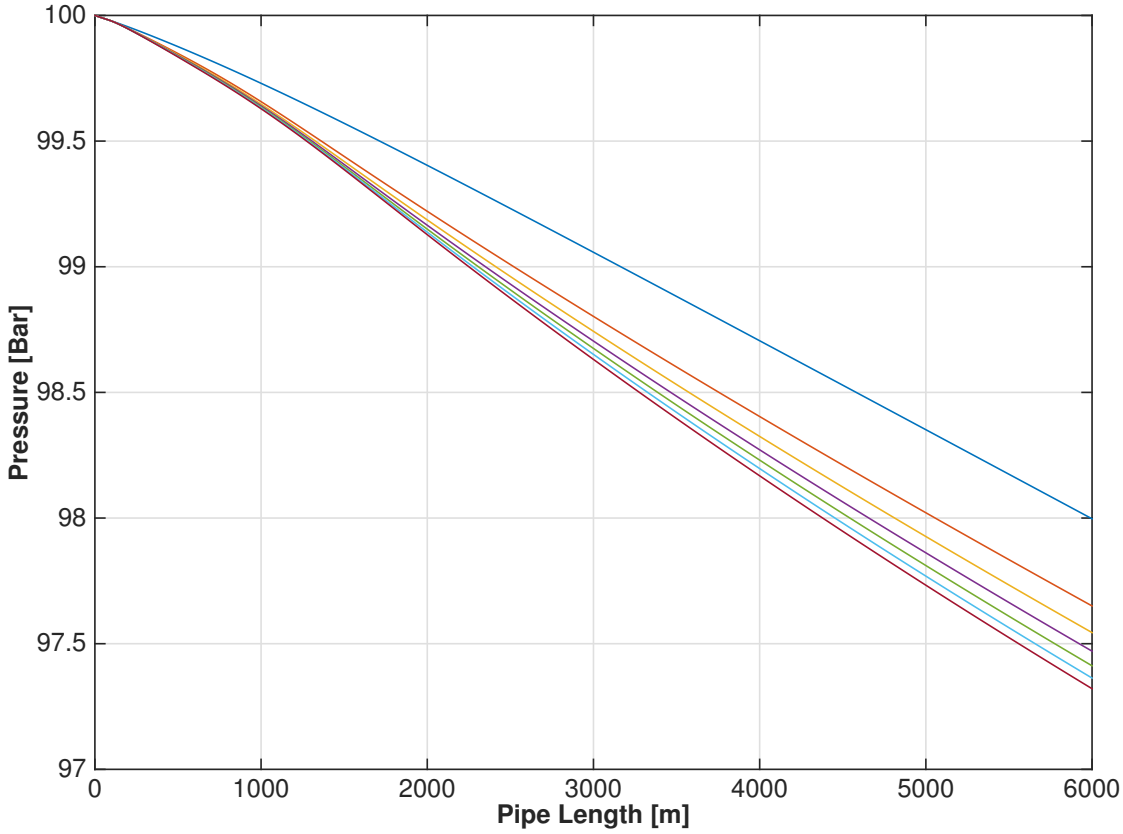


Figure 10.14: Pressure drop for wax deposition from 0 to 7 days with varying roughness

Below is the same figure as above, but zoomed in to better see each line.

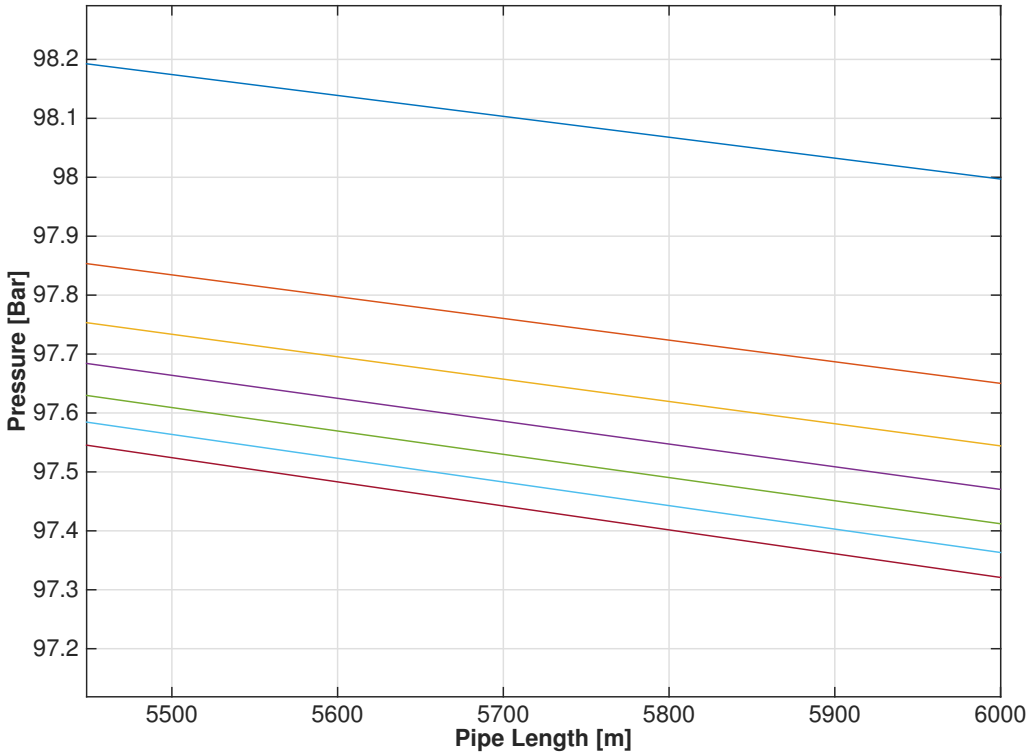


Figure 10.15: Pressure drop for wax deposition from 0 to 7 days with varying roughness zoomed in

10.8 Assumptions and Shortcomings

Most of the assumptions and shortcomings are presented in Section 7.3. Additionally some shortcomings are presented in this section, which are more specific for the simulations done in this chapter.

There are several assumptions which is done for the fluid property values such as wax density and wax conductivity. All of the values can be seen in the MATLAB scripts in appendix A.The wax density is assumed to be homogeneous. This means that the wax density is same for all of the wax deposited. This will most likely not be the case, as different wax components precipitate at different temperatures.

If the pipe length is e.g. 3 metre long and the pipe segments are divided into 2 meter lengths. Then the total number of pipe segments will be 1.5. The for loop in the MATLAB code treats this number as 1. This will only have an practical impact on short pipelines, and can be easily be avoided by choosing segment length so that the total number of pipe segments an integer.

11. Discussion and Conclusion

A subsea controlled wax deposition unit with a pigging loop is a possible solution to enable cold flow transportation of waxy crude oil. Cold flow transportation is when oil can be transporter at ambient temperature. A MATLAB program has been developed to calculate an axial temperature profile, and a wax deposition simulation for the unit. A lot of the fluid property values have been generated by NeqSim, which is a dynamic process simulator similar to the PVTsim.

The pressure drop for the unit is very low. When the surface roughness is kept constant the pressure drop decreases with more wax deposition. A lower diameter due to the wax should increase the pressure drop. However, the temperature profile changes when wax is deposited, due to the insulating effects wax has. If the wax thickness layer is added to the surface roughness, then the pressure drop decreases as wax is deposited.

The data gathered from experiments conducted by Statoil, which is discussed in Section 9.1, is compared to the four deposition models presented in Chapter 6. A tuning factor was added to the simulation to get a better correlation with the experimental values. Lack of data only made it possible to tune for a single value for each experiment. The Matzain and Heat Analogy models both gave the best recreation of the experimental values. The two models both include shear stripping. The Matzain model was chosen to be used for the controlled wax deposition unit simulation.

The unit proposal in Alternative II, see Section 4.1.2, is the best choice, because it utilises the pipe space the best. Pigging with a wax thickness layer between 2-4 mm is possible [38]. However, it is of great importance that most, if not all, of the wax precipitates inside the unit. By the 7 day simulations, presented in Chapter 10, it is possible to see that there is still some wax that can be deposited after the passing of the 6 km long unit. On the other hand, looking at the temperature profiles it is evident that the outlet temperature does not change much near the 5 km to 6 km pipe length position. If there is no temperature gradient then there will not be wax precipitation. From the simulation results one can see that a pigging frequency of 7 days for a length of 5 to 6 km, yields low deposition outside the unit. A lower frequent cleaning reduces the chance of a pig getting stuck and the wear on the pig. It is possible to have a shorter pipeline length, but then there might be a need for a higher pigging frequency in order to avoid big wax build up outside the unit. E.g. if a 4 km pipe length is chosen, then the pigging frequency should be around 2-3 days. It is important to note that the unit

is a loop and that a pipeline length of e.g. 6 km will mean a unit length of about 3 km.

Had the wax been spread out more evenly in the pipe then the pigging frequency could have been lower. The problem with spreading the deposition by manipulating the temperature is that the unit will have to be longer to get the same exit temperature.

Insulating the part after the unit will slow down the further cooling of the fluid, which is slightly above ambient temperature when exiting the subsea unit. This will allow the little wax that may deposit outside the unit to be more evenly spread out. The small amount of wax deposited outside the unit will be cleaned when the entire pipeline is pigged, which happens at a lower frequency than the regular unit pigging. If refrigeration is added to the last part of the unit, then the flow will go below the ambient temperature and in theory not precipitate any wax outside the unit. It is possible to add this to the model. It is highly unlikely that this will be cost efficient compared to how much the added cost will be by sending a pig through the entire pipeline at a more frequent time interval.

There is still a lot of work that needs to be done before the controlled wax deposition unit can be implemented and put to use in a field. However, the simulation model created while working on this thesis gives a good indication as to what the wax deposition profile may look like, and it paves the way for further development of a cold flow solution.

12. Further Work

- More work can be done on the MATLAB program: Adding an annulus temperature profile and optimising the code to enable quicker computation.
- Studying what effects a different cooling fluid (now only look at seawater) will have on the cooling, could be interesting.
- Comparisons to numerical simulations in programs such as OLGA can be done and studied in order to compare it to the model developed in MATLAB.
- Many of the assumptions made in Section 7.3 should be more closely looked at to see the potential effects the specific assumptions will have on the model. E.g. non-Newtonian versus Newtonian fluid, laminar sublayer versus not having a laminar sublayer etc.
- Further tuning of the Matzain and Heat Analogy models can be done, and shear stripping can be added to The Singh et al. and RRR models.
- Ageing effects are not included in the models and can be more closely studied.
- It is possible to alter the code to enable a simulation where the pig does not remove all of the wax. This means that the deposition and cleaning cycle values will be dependent on the previous simulation values.
- Fluid property value accuracy given by NeqSim can be further improved.
- Further development of the proposed unit discussed in Chapter 4 can be conducted.
- More experimental data should be collected in order to further improve the tuning of the simulations, and further verify the results.
- Updating the model to include natural convection and compare the model to existing waxy oil transport can be done. This may give a better understanding of the model and a better insight to possible improvements that can be done.
- An experiment where e.g. a 4 km long pipe is simulated can be done by recreating the point model experiments: Experiments can be done on a small pipe that represents a pipe segment in the controlled wax deposition unit(CWDU). Data should be collected for a given temperature that corresponds to a position in the CWDU. Changing the temperature will give a correspondents to another position

in the CWDU. Putting all of the experiment values together will correspond to all the pipe segments in the CWDU. This can give a great understanding as to how the wax deposition profile looks.

References

[1] E. O. Ajayi, "Modelling of controlled wax deposition and loosening in oil and gas production systems," *Norwegian University of Science and Technology, NTNU*, 2013.

[2] A. Hunt, "Uncertainties remain in predicting paraffin deposition," *Oil and Gas Journal*, vol. 94, no. 31, pp. 96–103, 1996.

[3] A. Hammami, C. H. Phelps, T. Monger-McClure, and T. Little, "Asphaltene precipitation from live oils: An experimental investigation of onset conditions and reversibility," *Energy & Fuels*, vol. 14, no. 1, pp. 14–18, 2000.

[4] F. H. Fadnes, "Natural hydrate inhibiting components in crude oils," *Fluid Phase Equilibria*, vol. 117, no. 1, pp. 186–192, 1996.

[5] R. Larsen, A. Lund, V. Andersson, K. Hjarbo, *et al.*, "Conversion of water to hydrate particles," in *SPE Annual Technical Conference and Exhibition*, Society of Petroleum Engineers, 2001.

[6] A. Singh, H. S. Lee, P. Singh, C. Sarica, *et al.*, "Flow assurance: Validation of wax deposition models using field data from a subsea pipeline," in *Offshore Technology Conference*, Offshore Technology Conference, 2011.

[7] K. S. Pedersen and H. P. Rønningsen, "Influence of wax inhibitors on wax appearance temperature, pour point, and viscosity of waxy crude oils," *Energy & fuels*, vol. 17, no. 2, pp. 321–328, 2003.

[8] Q. Miao, J. Liang, and B. Jiang, "Optimizing the pigging frequency through prediction of wax deposition of pipeline," in *2006 International Pipeline Conference*, pp. 729–739, American Society of Mechanical Engineers, 2006.

[9] J. Chen, J. Zhang, and H. Li, "Determining the wax content of crude oils by using differential scanning calorimetry," *Thermochimica acta*, vol. 410, no. 1, pp. 23–26, 2004.

[10] K. Paso, H. Kallevik, and J. Sjoblom, "Measurement of wax appearance temperature using near-infrared (nir) scattering," *Energy & Fuels*, vol. 23, no. 10, pp. 4988–4994, 2009.

[11] C. O'Malley, "Congo: Ds-1607 prevents wax deposition in deep water oil flow lines," *Schlumberger Case Study*, 2010.

[12] Rigzone, "How does pipeline pigging work?," *http://www.rigzone.com/training*.

[13] J. Southgate, *Wax removal using pipeline pigs*. PhD thesis, Durham University, 2004.

[14] "Subsea 7 internal document."

[15] Y. Guo, W. Li, L. Zhu, and H. Liu, "An excellent non-wax-stick coating prepared by chemical conversion treatment," *Materials Letters*, vol. 72, pp. 125–127, 2012.

[16] R. Hoffman and L. Amundsen, "Method for wax removal and measurement of wax thickness," 2009. WO Patent App. PCT/NO2008/000,371.

[17] F. P. Incropera, *Principles of heat and mass transfer*. Hoboken, NJ: John Wiley, 2013.

[18] A. Hamouda, J. Ravneøy, *et al.*, "Prediction of wax deposition in pipelines and field experience on the influence of wax on drag-reducer performance," in *Offshore Technology Conference*, Offshore Technology Conference, 1992.

[19] R. Hoffmann and L. Amundsen, "Single-phase wax deposition experiments," *Energy & Fuels*, vol. 24, no. 2, pp. 1069–1080, 2009.

[20] C. Sarica and M. Volk, "Tulsa university paraffin deposition projects," tech. rep., The University of Tulsa (US), 2004.

[21] X. Chen, T. Butler, M. Volk, J. Brill, *et al.*, "Techniques for measuring wax thickness during single and multiphase flow," in *SPE Annual Technical Conference and Exhibition*, Society of Petroleum Engineers, 1997.

[22] J. Creek, H. J. Lund, J. P. Brill, and M. Volk, "Wax deposition in single phase flow," *Fluid Phase Equilibria*, vol. 158, pp. 801–811, 1999.

[23] P. Singh, R. Venkatesan, H. S. Fogler, and N. Nagarajan, "Morphological evolution of thick wax deposits during aging," *AIChE journal*, vol. 47, no. 1, pp. 6–18, 2001.

[24] K. Rosvold, "Wax deposition models," *Norwegian University of Science and Technology, NTNU*, 2008.

[25] Z. Huang, S. Zheng, and H. S. Fogler, *Wax Deposition: Experimental Characterizations, Theoretical Modeling, and Field Practices*. CRC Press, 2015.

[26] R. Bagatin, C. Carniani, S. Correra, M. Margarone, C. Busto, *et al.*, "Wax modeling: there is need for alternatives," in *SPE Russian Oil and Gas Technical Conference and Exhibition*, Society of Petroleum Engineers, 2008.

[27] L. Hovden, C. Labes-Carrier, A. Rydahl, H. Ronningsen, and Z. Xu, "Pipeline wax deposition models and model for removal of wax by pigging: Comparison between

model predictions and operational experience.," in *The American Chemical Society*, vol. 225, 2003.

[28] O. Rygg, A. Rydahl, and H. Ronningsen, "Wax deposition in offshore pipeline systems," in *BHR GROUP CONFERENCE SERIES PUBLICATION*, vol. 31, pp. 193–206, MECHANICAL ENGINEERING PUBLICATIONS LIMITED, 1998.

[29] L. Hovden, C. Labes-Carrier, A. Rydahl, H. Ronningsen, and Z. Xu, "Pipeline wax deposition models and model for removal of wax by pigging: Comparison between model predictions and operational experience.," in *ABSTRACTS OF PAPERS OF THE AMERICAN CHEMICAL SOCIETY*, vol. 225, pp. U936–U936, AMER CHEMICAL SOC 1155 16TH ST, NW, WASHINGTON, DC 20036 USA, 2003.

[30] T. Brown, V. Niesen, D. Erickson, *et al.*, "Measurement and prediction of the kinetics of paraffin deposition," in *SPE Annual Technical Conference and Exhibition*, Society of Petroleum Engineers, 1993.

[31] A. Matzain, *Multiphase flow paraffin deposition modeling*. PhD thesis, University of Tulsa, 1999.

[32] E. Solbraa, "Neqsim," *NeqSim homepage - folk.ntnu.no/solbraa/neqsim/NeqSim*.

[33] K. S. Pedersen, P. L. Christensen, and J. A. Shaikh, *Phase behavior of petroleum reservoir fluids*. CRC Press, 2014.

[34] "Image source: http://stinger.no/services,"

[35] "Image source: http://www.dowcorning.com/content/oilgas/antifoams.aspx,"

[36] "Image source: http://www.sjvgeology.org/geology/anticline.gif,"

[37] T. Galta, "Bypass pigging of subsea pipelines suffering wax deposition," 2014.

[38] W. Wang, Q. Huang, S. Li, C. Wang, and X. Wang, "Identifying optimal pigging frequency for oil pipelines subject to non-uniform wax deposition distribution," in *2014 10th International Pipeline Conference*, pp. V004T08A004–V004T08A004, American Society of Mechanical Engineers, 2014.

A. MATLAB Scripts

Several of the MATLAB scripts written for the wax deposition model program is given in this Appendix.

A.1 Point Model Dialogue Box

```
%point_dialog.m

%*'This .m file is to be used in conjunction with inputsdlg.m which ...
    is a file
%copy written by Takeshi Ikuma 2009-2015 and Luke Reisner  2010
%(note the S in inputSdlg, i.e. not inputdlg)
%source: http://www.mathworks.com/matlabcentral/fileexchange/25862- ...
    inputsdlg--enhanced-input-dialog-box
%This document is created by Akul Vis

clear; close all;
point_model=1; %used in wax_calc..m file to determine if ...
    point_model values are to be used or not
Title = 'Akul''s Wax Deposition Program';

%%%% SETTING DIALOG OPTIONS
% Options.WindowStyle = 'modal';
Options.Resize = 'on';
Options.CancelButton = 'on';
Options.Interpreter = 'tex';
Options.ButtonNames = {'OK','Cancel'}; %<- default names, included ...
    here just for illustration
Option.Dim = 4; % Horizontal dimension in fields

Prompt = {};
Formats = {};
DefAns = struct([]);

Prompt(1,:) = {['This Program is Creaded by Akul Viswanathan and ...
    should only be used or made public with the concent of the ...
    creator. Creator''s e-mail address: akul1291@outlook.com. NB! ...
    Use period "." as decimal mark'],[],[]};
Formats(1,1).type = 'text';
Formats(1,1).size = [-1 0];
```

```
28 Formats(1,1).span = [1 1]; % item is 1 field x 4 fields %changed ...
       from [1 2]
29
30 Prompt(2,:) = {' Choose a Wax Model:','model',[]};
31 Formats(2,1).type = 'list';
32 Formats(2,1).format = 'text';
33 Formats(2,1).style = 'radiobutton';
34 Formats(2,1).items = {'Singh et. al.' 'RRR' 'Heat Analogy' ...
       'Matzain' }; %can use ; for new line and '' for empty space have ...
       to have space between 'test' 'test2'
35 Formats(2,1).labelloc= 'topleft'; %shifts input cell down one
36 Formats(end,1).unitsloc='rightmiddle'; %unit location
37 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
38 DefAns(1).model = 'RRR'; %first DefAns needs to have ()
39
40 %%%
41 Prompt(end+1,:) = {' Use constant fluid properties or ...
       NeqSim','neqsim',[]};
42 Formats(end+1,1).type = 'list';
43 Formats(end,1).format = 'text';
44 Formats(end,1).style = 'radiobutton';
45 Formats(end,1).items = {'No NeqSim' 'NeqSim'}; %can use ; for new ...
       line and '' for empty space have to have space between 'test' ...
       'test2'
46 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
47 Formats(end,1).unitsloc='rightmiddle'; %unit location
48 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
49 DefAns(1).neqsim = 'NeqSim'; %first DefAns needs to have ()
50
51 Prompt(end+1,:) = {' Length of pipesegment : ','dz','[m]'};
52 Formats(end+1,1).type = 'edit';
53 Formats(end,1).format = 'text';
54 Formats(end,1).size = 333; % automatically assign the height
55 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
56 Formats(end,1).unitsloc='rightmiddle'; %unit location
57 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
58 DefAns(1).dz = '0.5'; %first DefAns needs to have ()
59
60 Prompt(end+1,:) = {' Length of fluid segment: ','l_dz','[m]'};
61 Formats(end+1,1).type = 'edit';
62 Formats(end,1).format = 'text';
63 Formats(end,1).size = 333; % automatically assign the height
64 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
65 Formats(end,1).unitsloc='rightmiddle'; %unit location
66 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
67 DefAns(1).l_dz = '50'; %first DefAns needs to have ()
68
69 Prompt(end+1,:) = {' Number of fluid segments: ...
       ','total_num_FLUID_seg','[ - ]'};
70 Formats(end+1,1).type = 'edit';
71 Formats(end,1).format = 'text';
72 Formats(end,1).size = 333; % automatically assign the height
```

```
73 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
74 Formats(end,1).unitsloc='rightmiddle'; %unit location
75 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
76 DefAns(1).total_num_FLUID_seg = '15700'; %first DefAns needs to ...
       have ()
77
78 Prompt(end+1,:) = {' Pipeline length: ','total_length','[m]'};
79 Formats(end+1,1).type = 'edit';
80 Formats(end,1).format = 'text';
81 Formats(end,1).size = 333; % automatically assign the height
82 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
83 Formats(end,1).unitsloc='rightmiddle'; %unit location
84 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
85 DefAns(1).total_length = '0.5'; %first DefAns needs to have ()
86 %%%
87
88 %%%%
89 Prompt(end+1,:) = {'','Table',[]};
90 Formats(end+1,1).type = 'table';
91 Formats(end,1).format = {'char' 'logical'}; % table (= table in ...
       main dialog) / window (= table in separate dialog)
92 Formats(end,1).items = {'Chose plot(s) to display' ''};
93 Formats(end,1).size = [255 73];
94 Formats(end,2).span = [1 1];  % item is 2 field x 1 fields
95 DefAns.Table = {'Deposition Thickness' true
96                 'Temperature Profile' false
97                 'Pressure Drop' false
98                 '' false};
99
100 Prompt(end+1,:) = {' Hot product mass flow rate: ','m_h','[kg/s]'};
101 Formats(end+1,1).type = 'edit';
102 Formats(end,1).format = 'text';
103 Formats(end,1).size = 333; % automatically assign the height
104 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
105 Formats(end,1).unitsloc='rightmiddle'; %unit location
106 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
107 DefAns(1).m_h = '5.678'; %first DefAns needs to have ()
108
109 Prompt(end+1,:) = {' Cold seawater annulus mass flow rate: ...
       ','m_c','[kg/s]'};
110 Formats(end+1,1).type = 'edit';
111 Formats(end,1).format = 'text';
112 Formats(end,1).size = 333; % automatically assign the height
113 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
114 Formats(end,1).unitsloc='rightmiddle'; %unit location
115 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
116 DefAns(1).m_c = '4.167'; %first DefAns needs to have ()
117
118 Prompt(end+1,:) = {' Hot product inlet temperature: ' ...
       ,'T_hin','[C]'}; %convert to [K] afterwards
119 Formats(end+1,1).type = 'edit';
120 Formats(end,1).format = 'text';
```

```
121 Formats(end,1).size = 333; % automatically assign the height
122 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
123 Formats(end,1).unitsloc='rightmiddle'; %unit location
124 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
125 DefAns.T_hin = '20'; %first DefAns needs to have ()
126
127 Prompt(end+1,:) = {' Cold seawater annulus inlet temperature, i.e. ...
        ambient seawater:' ,'T_cin','[C]'}; %convert to [K] afterwards
128 Formats(end+1,1).type = 'edit';
129 Formats(end,1).format = 'text';
130 Formats(end,1).size = 333; % automatically assign the height
131 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
132 Formats(end,1).unitsloc='rightmiddle'; %unit location
133 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
134 DefAns(1).T_cin = '10'; %first DefAns needs to have ()
135
136
137 Prompt(end+1,:) = {' Inner flow diameter: ','D_h','[m]'};
138 Formats(end+1,1).type = 'edit';
139 Formats(end,1).format = 'text';
140 Formats(end,1).size = 333; % automatically assign the height
141 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
142 Formats(end,1).unitsloc='rightmiddle'; %unit location
143 % Formats(end,1).span = [2 1];  % item is 2 field x 1 fields
144 DefAns(1).D_h = '0.0525'; %first DefAns needs to have ()
145
146 Prompt(end+1,:) = {' Outer annulus flow diameter, i.e. without wall ...
        thickness included: ','D_cout','[m]'};
147 Formats(end+1,1).type = 'edit';
148 Formats(end,1).format = 'text';
149 Formats(end,1).size = 333; % automatically assign the height
150 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
151 Formats(end,1).unitsloc='rightmiddle'; %unit location
152 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
153 DefAns(1).D_cout = '0.1023'; %first DefAns needs to have ()
154
155 Prompt(end+1,:) = {' WAT: ','T_wat','[C]'};
156 Formats(end+1,1).type = 'edit';
157 Formats(end,1).format = 'text';
158 Formats(end,1).size = 333; % automatically assign the height
159 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
160 Formats(end,1).unitsloc='rightmiddle'; %unit location
161 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
162 DefAns(1).T_wat = '46.75'; %first DefAns needs to have ()
163
164 Prompt(end+1,:) = {' Hot fluid pipe wall surface roughness: ...
        ','epsi','[mm]'};
165 Formats(end+1,1).type = 'edit';
166 Formats(end,1).format = 'text';
167 Formats(end,1).size = 333; % automatically assign the height
168 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
169 Formats(end,1).unitsloc='rightmiddle'; %unit location
```

```
% Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
DefAns(1).epsi = '0.015'; %first DefAns needs to have ()

Prompt(end+1,:) = {' Hot fluid pressure inlet: ','P_h_inlet','[Bar]'};
Formats(end+1,1).type = 'edit';
Formats(end,1).format = 'text';
Formats(end,1).size = 333; % automatically assign the height
Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
Formats(end,1).unitsloc='rightmiddle'; %unit location
% Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
DefAns(1).P_h_inlet = '1.0132'; %first DefAns needs to have ()

Prompt(end+1,:) = {'Click for more information','',''};
Formats(end+1,1).type = 'button';
Formats(end,1).size = 200;
Formats(end,1).callback = @(¬,¬,handles,k)msgbox(sprintf('This ...
    Program is created by Akul Viswanathan as part of a Master ...
    Thesis',get(handles(k),'String')),'modal');



[Answer,Cancelled] = inputsdlg(Prompt,Title,Formats,DefAns,Options);
%DISPLAYS input values:
Answer

%Check if program is cancelled or not
if Cancelled==1
    disp('Program execution was cancelled!');
    break;
end

%%%
%isequal checks if values are equal, 1-> equal 0-> NOT equal
%check which radio button is chosen
singh= isequal(Answer.model,'Singh et. al.');
rrr= isequal(Answer.model,'RRR');
heatAnalogy=isequal(Answer.model,'Heat Analogy');
matzain= isequal(Answer.model,'Matzain');
%
no_neq= isequal(Answer.neqsim,'No NeqSim');
yes_neq= isequal(Answer.neqsim,'NeqSim');
%%%

%convert Answeres to numbers and assign a variable name to be used ...
    in wax
%program
dz = str2num(Answer.dz);
l_dz = str2num(Answer.l_dz);
total_num_FLUID_seg= str2num(Answer.total_num_FLUID_seg);
total_length= str2num(Answer.total_length);
%
```

```
m_h= str2num(Answer.m_h); %str2double converts array to number, ...
    here str2num is used because only single value
m_c= str2num(Answer.m_c);
T_hin = str2num(Answer.T_hin)+273.15; %273.15 to convert from [C] ...
    to [K]
T_cin = str2num(Answer.T_cin)+273.15; %273.15 to convert from [C] ...
    to [K]
% del_T = str2num(Answer.del_T);
D_h = str2num(Answer.D_h);
D_cout= str2num(Answer.D_cout);
T_wat= str2num(Answer.T_wat)+273.15; %273.15 to convert from [C] to [K]
epsi= str2num(Answer.epsi)*10^-3; % 10^-3 to convert from [m] to [mm]
P_h_inlet= str2num(Answer.P_h_inlet)*10^5; %10^5 to convert from ...
    [Bar] to [Pa]
%%%
   %Call Wax model.m program. with or without NeqSim
    if no_neq==1
        wax_calc__ediproject_heat_transfer_NO_neqsim__
    elseif yes_neq==1
        % CALL PROGRAM WITH NEQSIM
        NEQSIM_wax_calc__ediproject_heat_transfer;
    end

    %Call plot.m files if checkbox is selected
     if cell2mat((Answer.Table(1,2)))==1 %cell2mat is used because ...
        the table output value given in a cell array.
         %*'callplot.m
         disp('plot1');
     end

    if cell2mat((Answer.Table(2,2)))==1
        %*'callplot.m
        disp('plot2');
    end

    if cell2mat((Answer.Table(3,2)))==1
        %*'callplot.m
        disp('plot3');
    end

    if cell2mat((Answer.Table(4,2)))==1
        %*'callplot.m
        disp('plot4');
    end
%%%
```

A.2 Big Model Dialogue Box

```
%dialog.m

%*'This .m file is to be used in conjunction with inputsdlg.m which ...
    is a file
%copy written by Takeshi Ikuma 2009-2015 and Luke Reisner  2010
%(note the S in inputSdlg, i.e. not inputdlg)
%source: http://www.mathworks.com/matlabcentral/fileexchange/ ...
    25862-inputsdlg--enhanced-input-dialog-box
%This document is created by Akul Vis

clear; close all;

point_model=0; %used in wax_calc..m file to determine if ...
    point_model values are to be used or not
Title = 'Akul''s Wax Deposition Program';

%%%% SETTING DIALOG OPTIONS
% Options.WindowStyle = 'modal';
Options.Resize = 'on';
Options.CancelButton = 'on';
Options.Interpreter = 'tex';
Options.ButtonNames = {'OK','Cancel'}; %<- default names, included ...
    here just for illustration
Option.Dim = 4; % Horizontal dimension in fields

Prompt = {};
Formats = {};
DefAns = struct([]);

Prompt(1,:) = {['This Program is Creaded by Akul Viswanathan and ...
    should only be used or made public with the concent of the ...
    creator. Creator''s e-mail address: akul1291@outlook.com. NB! ...
    Use period "." as decimal mark'],[],[]};
Formats(1,1).type = 'text';
Formats(1,1).size = [-1 0];
Formats(1,1).span = [1 1]; % item is 1 field x 4 fields %changed ...
    from [1 2]

Prompt(2,:) = {' Choose a Wax Model:','model',[]};
Formats(2,1).type = 'list';
Formats(2,1).format = 'text';
Formats(2,1).style = 'radiobutton';
Formats(2,1).items = {'Singh et. al.' 'RRR' 'Heat Analogy' ...
    'Matzain'}; %can use ; for new line and '' for empty space have ...
    to have space between 'test' 'test2'
Formats(2,1).labelloc= 'topleft'; %shifts input cell down one
Formats(end,1).unitsloc='rightmiddle'; %unit location
% Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
DefAns(1).model = 'Matzain'; %first DefAns needs to have ()
```

```
40
41 %%%
42 Prompt(end+1,:) = {' Use constant fluid properties or ...
       NeqSim','neqsim',[]};
43 Formats(end+1,1).type = 'list';
44 Formats(end,1).format = 'text';
45 Formats(end,1).style = 'radiobutton';
46 Formats(end,1).items = {'No NeqSim' 'NeqSim'}; %can use ; for new ...
       line and '' for empty space have to have space between 'test' ...
       'test2'
47 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
48 Formats(end,1).unitsloc='rightmiddle'; %unit location
49 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
50 DefAns(1).neqsim = 'NeqSim'; %first DefAns needs to have ()
51
52 Prompt(end+1,:) = {' Length of pipesegment : ','dz','[m]'};
53 Formats(end+1,1).type = 'edit';
54 Formats(end,1).format = 'text';
55 Formats(end,1).size = 333; % automatically assign the height
56 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
57 Formats(end,1).unitsloc='rightmiddle'; %unit location
58 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
59 DefAns(1).dz = '1'; %first DefAns needs to have ()
60
61 Prompt(end+1,:) = {' Length of fluid segment: ','l_dz','[m]'};
62 Formats(end+1,1).type = 'edit';
63 Formats(end,1).format = 'text';
64 Formats(end,1).size = 333; % automatically assign the height
65 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
66 Formats(end,1).unitsloc='rightmiddle'; %unit location
67 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
68 DefAns(1).l_dz = '1'; %first DefAns needs to have ()
69
70 Prompt(end+1,:) = {' Number of fluid segments: ...
       ','total_num_FLUID_seg','[ - ]'};
71 Formats(end+1,1).type = 'edit';
72 Formats(end,1).format = 'text';
73 Formats(end,1).size = 333; % automatically assign the height
74 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
75 Formats(end,1).unitsloc='rightmiddle'; %unit location
76 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
77 DefAns(1).total_num_FLUID_seg = '2'; %first DefAns needs to have ()
78
79 Prompt(end+1,:) = {' Pipeline length: ','total_length','[m]'};
80 Formats(end+1,1).type = 'edit';
81 Formats(end,1).format = 'text';
82 Formats(end,1).size = 333; % automatically assign the height
83 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
84 Formats(end,1).unitsloc='rightmiddle'; %unit location
85 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
86 DefAns(1).total_length = '1500'; %first DefAns needs to have ()
87 %%%
```

```
88
89  %%%%
90  Prompt(end+1,:) = {'','Table',[]};
91  Formats(end+1,1).type = 'table';
92  Formats(end,1).format = {'char' 'logical'}; % table (= table in ...
        main dialog) / window (= table in separate dialog)
93  Formats(end,1).items = {'Chose plot(s) to display' ''};
94  Formats(end,1).size = [255 73];
95  Formats(end,2).span = [1 1];  % item is 2 field x 1 fields
96  DefAns.Table = {'Deposition Thickness' true
97                  'Temperature Profile' false
98                  'Pressure Drop' false
99                  '' false};
100
101 Prompt(end+1,:) = {' Hot product mass flow rate: ','m_h','[kg/s]'};
102 Formats(end+1,1).type = 'edit';
103 Formats(end,1).format = 'text';
104 Formats(end,1).size = 333; % automatically assign the height
105 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
106 Formats(end,1).unitsloc='rightmiddle'; %unit location
107 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
108 DefAns(1).m_h = '62'; %first DefAns needs to have ()
109
110 Prompt(end+1,:) = {' Cold seawater annulus mass flow rate: ...
        ','m_c','[kg/s]'};
111 Formats(end+1,1).type = 'edit';
112 Formats(end,1).format = 'text';
113 Formats(end,1).size = 333; % automatically assign the height
114 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
115 Formats(end,1).unitsloc='rightmiddle'; %unit location
116 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
117 DefAns(1).m_c = '230'; %first DefAns needs to have ()
118
119 Prompt(end+1,:) = {' Hot product inlet temperature: ' ...
        ,'T_hin','[C]'}; %convert to [K] afterwards
120 Formats(end+1,1).type = 'edit';
121 Formats(end,1).format = 'text';
122 Formats(end,1).size = 333; % automatically assign the height
123 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
124 Formats(end,1).unitsloc='rightmiddle'; %unit location
125 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
126 DefAns.T_hin = '100'; %first DefAns needs to have ()
127
128 Prompt(end+1,:) = {' Cold seawater annulus inlet temperature, i.e. ...
        ambient seawater:' ,'T_cin','[C]'}; %convert to [K] afterwards
129 Formats(end+1,1).type = 'edit';
130 Formats(end,1).format = 'text';
131 Formats(end,1).size = 333; % automatically assign the height
132 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
133 Formats(end,1).unitsloc='rightmiddle'; %unit location
134 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
135 DefAns(1).T_cin = '4'; %first DefAns needs to have ()
```

```
136
137
138 Prompt(end+1,:) = {' Inner flow diameter: ','D_h','[m]'};
139 Formats(end+1,1).type = 'edit';
140 Formats(end,1).format = 'text';
141 Formats(end,1).size = 333; % automatically assign the height
142 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
143 Formats(end,1).unitsloc='rightmiddle'; %unit location
144 % Formats(end,1).span = [2 1];  % item is 2 field x 1 fields
145 DefAns(1).D_h = '0.3'; %first DefAns needs to have ()
146
147 Prompt(end+1,:) = {' Outer annulus flow diameter, i.e. without wall ...
       thickness included: ','D_cout','[m]'};
148 Formats(end+1,1).type = 'edit';
149 Formats(end,1).format = 'text';
150 Formats(end,1).size = 333; % automatically assign the height
151 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
152 Formats(end,1).unitsloc='rightmiddle'; %unit location
153 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
154 DefAns(1).D_cout = '0.5'; %first DefAns needs to have ()
155
156 Prompt(end+1,:) = {' WAT: ','T_wat','[C]'};
157 Formats(end+1,1).type = 'edit';
158 Formats(end,1).format = 'text';
159 Formats(end,1).size = 333; % automatically assign the height
160 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
161 Formats(end,1).unitsloc='rightmiddle'; %unit location
162 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
163 DefAns(1).T_wat = '46.75'; %first DefAns needs to have ()
164
165 Prompt(end+1,:) = {' Hot fluid pipe wall surface roughness: ...
       ','epsi','[mm]'};
166 Formats(end+1,1).type = 'edit';
167 Formats(end,1).format = 'text';
168 Formats(end,1).size = 333; % automatically assign the height
169 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
170 Formats(end,1).unitsloc='rightmiddle'; %unit location
171 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
172 DefAns(1).epsi = '0.015'; %first DefAns needs to have ()
173
174 Prompt(end+1,:) = {' Hot fluid pressure inlet: ','P_h_inlet','[Bar]'};
175 Formats(end+1,1).type = 'edit';
176 Formats(end,1).format = 'text';
177 Formats(end,1).size = 333; % automatically assign the height
178 Formats(end,1).labelloc= 'topleft'; %shifts input cell down one
179 Formats(end,1).unitsloc='rightmiddle'; %unit location
180 % Formats(7,1).span = [2 1];  % item is 2 field x 1 fields
181 DefAns(1).P_h_inlet = '100'; %first DefAns needs to have ()
182
183 Prompt(end+1,:) = {'Click for more information','',''};
184 Formats(end+1,1).type = 'button';
185 Formats(end,1).size = 200;
```

```
186 Formats(end,1).callback = @(¬,¬,handles,k) ...
        msgbox(sprintf(Plot_thicknesscurve, ...
        get(handles(k),'String')),'modal');
187
188 [Answer,Cancelled] = inputsdlg(Prompt,Title,Formats,DefAns,Options);
189 %DISPLAYS input values:
190 Answer
191
192 %Check if program is cancelled or not
193 if Cancelled==1
194     disp('Program execution was cancelled!');
195     break;
196 end
197
198 %%%
199 %isequal checks if values are equal, 1-> equal 0-> NOT equal
200 %check which radio button is chosen
201 singh= isequal(Answer.model,'Singh et. al.');
202 rrr= isequal(Answer.model,'RRR');
203 heatAnalogy=isequal(Answer.model,'Heat Analogy');
204 matzain= isequal(Answer.model,'Matzain');
205 %
206 no_neq= isequal(Answer.neqsim,'No NeqSim');
207 yes_neq= isequal(Answer.neqsim,'NeqSim');
208 %%%
209
210 %convert Answeres to numbers and assign a variable name to be used ...
        in wax
211 %program
212 dz = str2num(Answer.dz);
213 l_dz = str2num(Answer.l_dz);
214 total_num_FLUID_seg= str2num(Answer.total_num_FLUID_seg);
215 total_length= str2num(Answer.total_length);
216 %
217 m_h= str2num(Answer.m_h); %str2double converts array to number, ...
        here str2num is used because only single value
218 m_c= str2num(Answer.m_c);
219 T_hin = str2num(Answer.T_hin)+273.15; %273.15 to convert from [C] ...
        to [K]
220 T_cin = str2num(Answer.T_cin)+273.15; %273.15 to convert from [C] ...
        to [K]
221 % del_T = str2num(Answer.del_T);
222 D_h = str2num(Answer.D_h);
223 D_cout= str2num(Answer.D_cout);
224 T_wat= str2num(Answer.T_wat)+273.15; %273.15 to convert from [C] to [K]
225 epsi= str2num(Answer.epsi)*10^-3; % 10^-3 to convert from [m] to [mm]
226 P_h_inlet= str2num(Answer.P_h_inlet)*10^5; %10^5 to convert from ...
        [Bar] to [Pa]
227 %%%
228    %Call Wax model.m program. with or without NeqSim
229     if no_neq==1
230         wax_calc__ediproject_heat_transfer_NO_neqsim__
```

```
231     elseif yes_neq==1
232         %CALL PROGRAM WITH NEQSIM
233         NEQSIM_wax_calc__ediproject_heat_transfer
234     end
235
236     %Call plot.m files if checkbox is selected
237      if cell2mat((Answer.Table(1,2)))==1 %cell2mat is used because ...
            the table output value given in a cell array.
238          %*'callplot.m
239          disp('plot1');
240      end
241
242     if cell2mat((Answer.Table(2,2)))==1
243         %*'callplot.m
244         disp('plot2');
245     end
246
247     if cell2mat((Answer.Table(3,2)))==1
248         %*'callplot.m
249         disp('plot3');
250     end
251
252     if cell2mat((Answer.Table(4,2)))==1
253         %*'callplot.m
254         disp('plot4');
255     end
256 %%%
```

A.3 Main Program Script

```
%NEQSIM_wax_calc__ediproject_heat_transfer.m

format LONG %more decimal places. depo thickness is small

%This function calculates the hot oil bulk- and wall temperature.
%NB! All temperatures are in KELVIN
%NB! start from i=1 is where seawater flows in and oil out ...
    (counter-current
%heat exchanger)

%function [T_h_bulk, T_h_wall, T_c_bulk, Q_tot_h, Length] = ...
    ediproject_heat_transfer_with_neqsim ...
    (m_h,m_c,T_hin,T_cin,del_T,dz,D_h,D_cout)

%waitbar -> shows code progression
waitbarcomp= waitbar(0,'Computing! Please wait...');

%NEQSIM
% This value is fluid composition dependent
max_dissolved_wax=4.5;
%Load regression values that are made from the NeqSim values
load('P_waxdepo_neqsim_eq.mat'); %this will be set as parameter ...
    P_waxdepo
P_rho= load('P_rho.txt');
P_vis= load('P_vis.txt');
P_cp= load('P_cp.txt');
P_k= load('P_k.txt');
P_VA= load('P_VA.txt');
%
%calc a value for a given temp example:
%%rho_h=polyval(P_rho,TEMP)
%

%value from Length calc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%
% m_h, Mass flowrate hot/oil [kg/s]
% m_c,  Mass flowrate cold/seawater [kg/s]
% T_hin, Temperature hot/oil in to heat exchanger (aka. T(1)) [K]
% T_cin, Temperature cold/seawater in to heat exchanger [K]
% del_T, Delta temperature above T_cin allowed (T_hout = T_cin + ...
    del_T) [K or C]
% dz, Step size i.e size of iteration step [m]
% D_h, inner hot/oil pipe diameter [m]
% D_cout, Outer wall diameter annulus flow, inside flow, i.e. ...
    without outer
```

```
%   wall thickness. [m]
%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%Default values
%  m_h = 62; %[kg/s]
%  m_c = 230; %[kg/s]
%  T_hin = 100+273.15; %[K]
%  T_cin = 4+273.15; %[K]
%  del_T = 5; %[K or C]
%  dz = 1; %[m]
%  D_h = 0.3; %[m]inner most diameter hot fluid pipe
%  D_cout= 0.5; %[m]


%%%%%%%
total_num_pip_seg=total_length/dz;
%Hinder total number of pipe segments being less than one
if total_num_pip_seg<1
    total_num_pip_seg=1;
end
%Hinder total number of fluid segments being less than one
if total_num_FLUID_seg<1
    total_num_FLUID_seg=1;
end
%set temperature profile of cooling seawater to constant. make matrix
T_c_bulk(1,1:total_num_FLUID_seg)= T_cin;
%%%%%%%

%Check that Fluid segment is equal to or longer than the pipe line ...
    segment
if l_dz<dz
    disp('Error l_dz<dz: Please make sure that the Fluid segment is ...
        equal to or larger than the pipe line segment!');
    break;
end

%counter
i=1;
j=1;


Q_tot_h=0;% [W] initial heat loss
R_win= D_h/2; % Inner wall hot/oil pipe radius
% T_hout= T_cin + del_T; % Hot/oil temp. out of exchanger

   %Use point model values or proposed big scale model values
    if point_model==1

        % Thermal conductivity values from subsea 7 "akul mathcad"
```

```
93          % Assume CRA liner, carbon steel, CRA liner then ...
               potentially wax layer
94          k_steel= 45; %[W/(m*K)]
95
96          %thickness
97          t_steel_point= 3.9*10^-3; %[m]
98
99          %Diameter inner steel wall annulus
100         D_inner_c = D_h + t_steel_point*2;
101
102         res_cra_h= 0;  %[K/W] Conduction resistance CRA oil ...
               side,(D_h+t_cra_in) is diameter with CRA layer
103         res_steel= log((D_inner_c)/(D_h))/(2*pi*k_steel*dz); %[K/W]
104         res_cra_c= 0; %[K/W]
105
106
107     elseif point_model==0
108
109         % Thermal conductivity values from subsea 7 "akul mathcad"
110         % Assume CRA liner, carbon steel, CRA liner then ...
               potentially wax layer
111         k_cra= 10; %[W/(m*K)]
112         k_steel= 45; %[W/(m*K)]
113
114         % Each layer thickness (rounded off values from subsea 7 ...
               doc "akul mathcad")
115         t_cra_h= 5*10^-3; %[m]
116         t_steel= 30*10^-3; %[m]
117         t_cra_c= 5*10^-3; %[m]
118         %t_wax= ; in i-loop
119
120         %(in order from center)
121         D_cra_h= D_h+2*t_cra_h;  %CRA_inner steel interface
122         D_steel= D_cra_h+2*t_steel; %Steel CRA_outer interface
123         D_cra_c= D_steel+2*t_cra_c; %CRA_outer seawater interface
124
125         %resistance
126         res_cra_h= log((D_cra_h)/(D_h))/(2*pi*k_cra*dz);  %[K/W] ...
               Conduction resistance CRA oil side,(D_h+t_cra_in) is ...
               diameter with CRA layer
127         res_steel= log((D_steel)/(D_cra_h))/(2*pi*k_steel*dz); %[K/W]
128         res_cra_c= log((D_cra_c)/(D_steel))/(2*pi*k_cra*dz); %[K/W]
129
130         D_inner_c=D_cra_c;
131     end
132
133
134
135 D_c_hyd= D_cout - D_inner_c; %[m] Hydraulic diameter of annulus
136
137
138 %Pipe is clean when first fluid segment flows through pipe. j=1 NO WAX
```

```
Depo_thick(1:total_num_pip_seg,1)=0;
Res_wax(1:total_num_pip_seg,1)= 0;%no wax resistance when pipe is wax free
D_wax(1:total_num_pip_seg,1)= D_h; %inner diameter without wax
k_wax= 0.2; %[W/(m*K)] (http://www.engineeringtoolbox.com/thermal-conductivity-d_429.html)
M_h_minus_depo(1,1)=m_h;%mass flow rate after deposition (i.e. NOT including deposited wax) Where first fluid segment and pipe segment has not deposit[kg/s]
Total_time_vec(j)=0; %used in time_seg_calc.m


%j is the fluid segment number
%set initial liquid section equal to pipe segment length dz


T_h_bulk(1,1:total_num_FLUID_seg)= T_hin; %[K] Temp. first cell, hot out and seawater in
%Pressure values used in Pressure_drop.m
P_h(1,1:total_num_FLUID_seg)=P_h_inlet;

for j=1:total_num_FLUID_seg

    T_c_bulk(1:total_num_pip_seg,j)= T_c_bulk(1,j); %[K] making cold flow array into matrix with same values
    %T_h_wall(1,j)=0; % This is at pipeline exit and is set to be unknown/skipped

    %Run wait bar
    waitbar(j/total_num_FLUID_seg,waitbarcomp);


%i=1 is Hot inlet. This calculation is done from
%Note this calculation is done with i=1 as hot INLET
%i is the pipe segment number
    for i=1:total_num_pip_seg % total_num_seg is the number of segments (total length)/dz found in lenght calc
        %pluss one to include the time it takes the fluid to exit the pipe
        %i=1 is the beginning of the pipe(left side of segment), i.e. 0m

        %Fluid properties

        %K_h(1,j)= 0.1; % [W/(m*K)] conductivity oil
        k_c= 0.563; % [W/(m*K)] conductivity seawater
        %Cp_h(1,j)= 2250; %[J/(kg*K)] Cp specific heat capacity for oil
        %Cp_h_wall(1,j)= 2250;
        Cp_c(1,j)= 3985; %[J/(kg*K)] Cp specific heat capacity for seawater
        %Rho_h(1,j)= 817; %[kg/m^3] Density for oil
        %Rho_h_wall(i,j)=817;
```

```
180         Rho_c(1,j)= 1027; %[kg/m^3] Density for seawater
181         %Vis_h(1,j)= 4*10^-4; %[Pa*s]Dynamic viscosity for oil, aka muy
182         %Vis_h_wall(1,j)=4*10^-4;
183         Vis_c(1,j)=1.88*10^-3; %[Pa*s]Dynamic viscosity for ...
                seawater aka muy
184
185         Rho_h(i,j)=polyval(P_rho,T_h_bulk(i,j));
186
187         Vis_h(i,j)=polyval(P_vis,T_h_bulk(i,j));
188
189         Cp_h(i,j)=polyval(P_cp,T_h_bulk(i,j));
190
191         K_h(i,j)=polyval(P_k,T_h_bulk(i,j));
192
193         M_h_minus_depo(i,j)=m_h; % this can be updated to change as ...
                there is deposition. Note will have to add stripped wax.
194         %"Annulus" of wax depo area * i*dz which is Length* wax density
195         % M_depo(i,j)= ((D_h^2/4)*pi - ...
                (D_wax^2/4)*pi)*(i*dz)*Rho_wax %"annulus"
196
197         %Reynolds Number
198         Re_h(i,j)=4*M_h_minus_depo(i,j)/(pi*D_wax(i,j)*Vis_h(i,j)); ...
                %Re=(4*m)/(pi*D*muy) Hot/oil
199         Re_c(i,j)=4*m_c*D_c_hyd/(pi*(D_cout^2-D_inner_c^2)* ...
                Vis_c(i,j)); %Re=(4*m*D_hyd)/(pi*(Dout^2-Din^2)*muy) ...
                cold/seawater annulus
200
201         %Prandtl Number
202         Pr_h(i,j)= (Cp_h(i,j)*Vis_h(i,j))/(K_h(i,j)); %[-] ...
                Pr=(Cp*muy)/k
203         Pr_c(i,j)= (Cp_c(i,j)*Vis_c(i,j))/(k_c); %[-] Pr=(Cp*muy)/k
204
205         %Convective heat transfer coefficient
206            %by combining and rearranging Dittus-Boelter equation ...
                  and Nussult Number eq.
207         H_h(i,j)= ...
                (0.023*Re_h(i,j)^0.8*Pr_h(i,j)^0.3)*K_h(i,j)/D_wax(i,j); ...
                %h=(Nu*k)/D_hyd ..&.. Nu=0.023*Re^0.8*Pr^0.3 cooling->0.3
208         H_c(i,j)= (0.023*Re_c(i,j)^0.8*Pr_c(i,j)^0.4)*k_c/D_c_hyd; ...
                %h=(Nu*k)/D_hyd ..&.. Nu=0.023*Re^0.8*Pr^0.4 heating->0.4
209
210         %Resistance Calculations from inside to out of tube:
211         %ref: Figure 3.8 and 3.34 in incropera heat and mass transfer
212         %%%?
213         Res_h(i,j)= 1/(pi*D_wax(i,j)*H_h(i,j)*dz); %[K/W] ...
                Convection resistance hot side in dz section
214         %
215         Res_c(i,j)= 1/(pi*D_inner_c*H_c(i,j)*dz); %[K/W] Convection ...
                resistance cold side
216         %%%
217
218
```

```
219         %TEMP and Total Resistance calc
220         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
221          Res_tot(i,j)= Res_h(i,j) + res_cra_h + res_steel + ...
                 res_cra_c + Res_c(i,j)+ Res_wax(i,j);
222
223
224          %"UA"=1/Res_tot where A is the area of 1 section
225
226
227                  %"Radial" heat loss in section/length dz
228                  Qr(i,j)= ...
                         1/(Res_tot(i,j))*(T_h_bulk(i,j)-T_c_bulk(i,j)) ; ...
                         %[W] Note! use outer area, possible to use ...
                         inner. "UA"=1/Res_tot
229
230                  %if the fluid segment is twice as big as the pipe ...
                         segment then
231                  %Qr will be double as big compared to if they were ...
                         equally
232                  %long.
233                  fluid_pipe_length_ratio=l_dz/dz;
234                  %"Radial" heat loss in section/length dz MULTIPLIED by
235                  %ratio yielding the total Qr for a fluid semgent
236                  Qr_fluid_seg(i,j)= ( ...
                         1/(Res_tot(i,j))*(T_h_bulk(i,j)-T_c_bulk(i,j)) ...
                         )*fluid_pipe_length_ratio; %[W] Note! use outer ...
                         area, possible to use inner. "UA"=1/Res_tot
237
238
239          T_h_wall(i,j)= T_h_bulk(i,j) - Qr(i,j)*Res_h(i,j); %[K] ...
                 calculate temperature at wall, NOTE. at current pipe ...
                 seg. Qr=q_h_axial=(1/Rconv)*(delT)
240
241          T_h_bulk(i+1,j)= T_h_bulk(i,j) - ...
                 Qr(i,j)/(M_h_minus_depo(i,j)*Cp_h(i,j)); %[K] calculate ...
                 next temperature a step upstream hot. ...
                 Qr=q_h_axial=m*cp*(delT)
242          %remove T_c_bulk beacause it is assumed this temp. profile is
243          %constant T_c_bulk
244
245          %wall fluid properties
246          Rho_h_wall(i,j)=polyval(P_rho,T_h_wall(i,j));
247          Vis_h_wall(i,j)=polyval(P_vis,T_h_wall(i,j));
248          Cp_h_wall(i,j)=polyval(P_cp,T_h_wall(i,j));
249
250          Re_h_wall(i,j)= ...
                 4*M_h_minus_depo(i,j)/(pi*D_wax(i,j)*Vis_h_wall(i,j)); ...
                 %Re=(4*m)/(pi*D*muy) Hot/oil
251          Pr_h_wall(i,j)= ...
                 (Cp_h_wall(i,j)*Vis_h_wall(i,j))/(K_h(i,j)); %[-] ...
                 Pr=(Cp*muy)/k
```

```
252           H_h_wall(i,j)= ...
                  (0.023*Re_h_wall(i,j)^0.8*Pr_h_wall(i,j)^0.3) * ...
                  K_h(i,j)/D_wax(i,j); %h=(Nu*k)/D_hyd ..&.. ...
                  Nu=0.023*Re^0.8*Pr^0.3 cooling->0.3
253           %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
254
255           pressure_drop
256           %call velocity_time_calc.m which calculates the time_seg
257         % time_seg_calc
258           CORRECTED_time_seg_calc
259
260
261           %This calls the equation for various temps to find ...
                  dissolved wax concentration
262           %remove possible regression oscillations (which might make the
263           %values go below zero)
264           %Not making a matrix for the dep_wax_.. values because this ...
                  is done
265           %for disolved
266           if ppval(P_waxdepo,T_h_bulk(i,j))*100<0
267               dep_wax_bulk=0;
268           else
269               dep_wax_bulk=ppval(P_waxdepo,T_h_bulk(i,j))*100;
270           end
271           if ppval(P_waxdepo,T_h_wall(i,j))*100<0
272               dep_wax_wall=0;
273           else
274               dep_wax_wall=ppval(P_waxdepo,T_h_wall(i,j))*100;
275           end
276
277
278           Dis_conc_bulk(i,j)= max_dissolved_wax - dep_wax_bulk;
279           Dis_conc_wall(i,j)= max_dissolved_wax - dep_wax_wall;
280
281   %        %remove "tail" created in NeqSim
282   %         if T_h_bulk(i,j)>T_wat
283   %             Dis_conc_bulk(i,j)=max_dissolved_wax;
284   %         end
285   %         if T_h_wall(i,j)>T_wat
286   %             Dis_conc_wall(i,j)=max_dissolved_wax;
287   %         end
288
289           %call diffusivity coefficient calculation, also values such ...
                  as Sc
290           %and Sh is calculated here
291            diffusivity_coefficient
292
293
294            %call one of the mass_transfer.m scripts
295
296           if singh==1
297               mass_transfer_singh
```

```
298         elseif rrr==1
299             mass_transfer_RRR
300         elseif heatAnalogy==1
301             mass_transfer_HeatAnalogy
302         elseif matzain
303             mass_transfer_matzain
304         else
305             disp(' Error: If statment. choosen radiobutton')
306         end
307
308
309
310         % Calculates depo thickness in a pipe segment
311         %Note the j+1 this is due to the fact that the thickness is ...
                assumed
312         %to be added after the liquid segment has passed i.e. it only
313         %effects the next liquid segment.
314
315         if 2*Depo_thick(i,j+1)≥D_h
316             %continuing calculations will give imaginary values!
317             disp('CODE STOPPED! The pipe is completely blocked!');
318             return;
319         end
320
321        %%%%%%%%%%%%%%%%%%
322         %multiply by two because wax depo on top and bottom
323         D_wax(i,j+1)= D_h - 2*Depo_thick(i,j+1); %diameter after ...
                depo. Depo_thick yields the total deposition. hence D_h ...
                is used and not D_wax(i,j) as this would "add" the ...
                previous deposit twice
324         Res_wax(i,j+1)=log((D_h)/(D_wax(i,j+1)))/(2*pi*k_wax*dz);  ...
                %[K/W] Conduction resistance wax
325        %%%%%%%%%%%%%%%%%%
326
327         K_h(i+1,j)= K_h(i,j); % This is done so it will be easy to ...
                vary Cp for each step
328
329         Cp_h(i+1,j)= Cp_h(i,j); % This is done so it will be easy ...
                to vary Cp for each step
330         Cp_c(i+1,j)= Cp_c(i,j); % This is done so it will be easy ...
                to vary Cp for each step
331
332         %Rho_h(i+1,j)= Rho_h(i,j); % This is done so it will be ...
                easy to vary  for each step
333         %Rho_h_wall(i+1,j)= Rho_h_wall(i,j);% This is done so it ...
                will be easy to vary for each step
334         Rho_c(i+1,j)= Rho_c(i,j); % This is done so it will be easy ...
                to vary for each step
335
336         %Vis_h(i+1,j)= Vis_h(i,j); % This is done so it will be ...
                easy to vary  for each step
```

```
337         Vis_c(i+1,j)= Vis_c(i,j); % This is done so it will be easy ...
               to vary for each step
338
339         %Total heat lost in the oil/hot fluid
340         Q_tot_h= Q_tot_h+Qr_fluid_seg(i,j); %[W] heat loss added ...
               for each section, which gives the total heat lost in the ...
               hot fluid
341
342
343
344
345     end
346 end
347
348
349 %total pressure drop
350 total_pressure_drop= P_h_inlet - P_h(i,j); %i=last pipe segment ...
       j=last fluid segment
351
352 %closes progression bar
353 close(waitbarcomp)
```

A.4 Time Segment Calculation

```
%CORRECTED_time_seg_calc.m

%calculate time for the liquid segment to pass a point on the pipeline
%segment. This can be done by calculating the velocity in each pipeline
%segment for each fluid segment.

% this segment length will change as wax is deposited, but will ... yield the
% same if the initial values are used or the new geometric values ... ("cancel
% out"). the length of the liquid segment increases when the area ... decreases
% but the velocity increases, the throughput time is the same. hence
% calculate with initial geometric and values as this is simpler
%set initial liquid segment length  L_dz=dz

A_h_cross_clean= pi*D_h^2/4; % A=pi*r^2

A_h_cross_wax(i,j)= pi*D_wax(i,j)^2/4;

%veloscity of h_bulk including narrowing pipe due to wax
vel_h(i,j)= m_h/(Rho_h(i,j)*A_h_cross_wax(i,j));

%t_seg is the time it takes a liquid segments to pass a point in the
%pipe segment i, from front to back
%NB! A_h_cross_wax is not used intentionally! because Time_seg will ... be the same
%as l_dz is proportionally increased with wax deposition.
Time_seg_to_next_FLUIDseg(i,j)= ... l_dz*Rho_h(i,j)*A_h_cross_clean/m_h; %t=s/v=
            %(s_clean_liquid*rho*A_clean_cross)/mass_flowrate
%Use dz as opposed to the above eq. This calcuates the time time from
%beginning of fluid segment to travel from enterence of the pipe ... segment to
%its exit.
Time_seg_to_next_PIPEseg(i,j) = dz*Rho_h(i,j)*A_h_cross_clean/m_h;

%Total time let product flow enter the pipe. Can think of this as ... the amount of time
%we let a valve be open and allow fluid through.
if i==total_num_pip_seg && j==total_num_FLUID_seg
    time_valve_open= ...
        Time_seg_to_next_FLUIDseg(1,1)*total_num_FLUID_seg;%[s]
    time_valve_open_HOURS=time_valve_open/ (60*60);
end

%This is moved outside wax.m for loop so that the cell is not ... erased each
%loop!
```

```
41 %Total_time_vec(1)=0;
42
43 if j==1
44     % time of fluid segment one, from inlet to outlet (once loop ...
          has gone through all i
45     %values) from front part of first fluid segment enters pipe to ...
          front
46     %part is at exits
47     Total_time_vec(j)= Total_time_vec(j) + ...
          Time_seg_to_next_PIPEseg(i,1);
48         %add the time it takes the first fluid segment, when its ...
              front is at
49         %the pipeline exit, to travel so that the end of the fluid ...
              segment
50         %is at the pipeline exit. The first Tota_time_vec(1) is the ...
              total
51         %time the first fluid segment is in the pipeline(from start ...
              to end)
52     if j==1 && i==total_num_pip_seg
53         Total_time_vec(j)= Total_time_vec(j) + ...
              Time_seg_to_next_FLUIDseg(i,j);
54     end
55 elseif j≠1 && i == total_num_pip_seg; %when j is NOT equal to 1 and ...
       i is equal to length+1
56     %add values in a vector for the last pipe segment, this can ...
          later be
57     %added. this calculates the
58     Total_time_vec(j)= Time_seg_to_next_FLUIDseg(i,j);
59 end
60
61 %This is calculated in wax.m
62 %total_time=sum(Total_time_vec);
```

A.5 Pressure Drop Calculation

```
% pressure_drop.m

% Can add epsi for wax as an input when this is to be calculated
%
 %function [P_h,Del_P_tot,head_l_tot]= ...
    final_pressure_drop_headloss(epsi)
% epsi, epsilon roughness height,  [m]
% P_h_inlet, [Pa] is the pressure at hot inlet
% Del_P_tot is total pressure loss [Pa]
% head_l_tot is total head loss [m]

%epsi=0.015*10^-3; %[m]
%P_h_inlet= 100*10^5; %[Pa]

P_h(1,1)=P_h_inlet;

    if i≥2
        g=9.81; %[m/s^2] gravitational acceleration
        %remove?!%Del_P_tot(1)=0; %set start value



        % %Calculate second cell pressure:
        % Fd(1)= 0.25*(log10(epsi/3.7*D_wax + ...
           5.74/Re_h(1)^0.9))^-2; %[-] Darcy friction factor. ...
           Swamee-jain equation
        % Head_l(1)= ( Fd(1)*dz/D_wax^5 * ...
           8*m_h^2/(Rho_h(1)^2*pi^2*g) ); %calc head loss for each ...
           pipe segment (same if parameters such as roh, epsi etc ...
           stay the same)
        %
        % P_h(2)= P_h(1)- (Rho_h(1)*g*Head_l(1)); % inlet pressure ...
           - one segment pressure drop
%             if Depo_thick(i,j)>0
%                 %epsi=
%             end

        Fd(i,j)= 0.25*(log10(epsi/(3.7*D_wax(i,j)) + ...
           5.74/(Re_h(i,j)^0.9)))^-2; %[-] Darcy friction factor. ...
           Swamee-jain equation
        %This equation is not used, rather go via the headloss ...
           equation,
        %P_h(v)= P_h_inlet - ( Fd(v)*dz/D_wax^5 * ...
           8*m_h^2/(Roh_h_fl(v)*pi^2) ); %[Pa] Pressure at each ...
           step, dz, down pipe. From hot inlet to hot outlet

        %head_l_tot= head_l_tot + ( Fd(v)*dz/D_wax^5 * ...
           8*m_h^2/(Roh_h_fl(v)^2*pi^2*g) ); %Total head loss from ...
           start to end
```

```
36
37          Head_l(i,j)= ( (Fd(i,j)*dz/(D_wax(i,j)^5)) * ...
                (8*m_h^2)/(Rho_h(i,j)^2*pi^2*g) ); %calc head loss for ...
                each pipe segment (same if parameters such as roh, epsi ...
                etc stay the same)
38
39          P_h(i,j)= P_h(i-1,j) -  (Rho_h(i,j)*g*Head_l(i,j)); %Array ...
                containing pressureloss for each segment
40
41         % Del_P_tot(j)= Del_P_tot(j) + ( Rho_h(i,j)*g*Head_l(i,j) ); ...
              % [Pa] delP= roh*g*headloss, add each pipe segment
42      end
```

A.6 Wax Mass Calculation

```
1  %wax_mass.m
2  format long
3  wax_volume(1)=0
4  for count_mass=1:total_num_pip_seg
5      %calculate the annulus volume aka the wax
6    wax_volume(count_mass+1)=  wax_volume(count_mass) + (pi*D_h^2/4- ...
         pi*D_wax(count_mass,+1)^2/4)*dz ;
7  end
8  Total_wax_volume=sum(wax_volume)
9  Total_wax_mass= Total_wax_volume * 0.909*1000 %[kg]
```

A.7 Diffusion Coefficient Calculation

```
%D_AB:diffusion.m

%Diffusion coefficient of wax
%Here molar volume is also found.
%Values found here are directly used in the various mathematical ...
    wax model
%.m files
%AVERAGE molar volume of n-paraffins [cm^3/mol]
%use these values if NeqSim is not to be used
if no_neq==1
    %values from experimental values one data point (does not ...
        change much)
    mw_w= 404; % %(total)wax molar Volume [g/mol]
    rho_w=0.909; % %wax density [kg/l=g/cm^3], NB! g/cm^3
    V_A(i,j)= mw_w/rho_w; %* [cm^3/mol] %M_A[g/mol] rho[kg/l=g/cm^3]
    %V_A=M_A/rho_w where w is the paraffins
elseif yes_neq==1  %if NeqSim regression values are to be used

    %file loaded in main .m file
    V_A(i,j)=polyval(P_VA,T_h_bulk(i,j));
end

D_AB(i,j)= ( 13.3*10^-12 * ...
    T_h_bulk(i,j)^1.47*((Vis_h(i,j)*10^3)^((10.2/V_A(i,j))-0.791))/ ...
    (V_A(i,j)^0.71) );

%Schmidt number:
Sc(i,j)= Vis_h(i,j)/(Rho_h(i,j)*D_AB(i,j));

%Sherwood number (dittus-boelter analogy correlation):
Sh(i,j)= 0.023*Re_h(i,j)^(0.8)*Sc(i,j)^0.3;
```

A.8 Wax Deposition Model -Singh

```
% mass_transfer_singh.m

%The Singh et. al model
%this script calculates mass transfer values needed for the wax ...
    deposition
%modelling

%assume Density for wax layer to be about (just for Singh model):
rho_w=0.909; %[g/cm^3]

%D_AB (diffusion coef.) and V_A (Molar Volume) is calculated/found in
%diffusion_coefficient.m

%Singh et. al. Model
%%%%%%%%%%%%%%%%%%%%
    %Mass flux
    %multiply C_ by Rho to get concentration in Kg/m^3
    J_A(i,j)= (Sh(i,j)/D_wax(i,j)) * ...
        D_AB(i,j)*(Dis_conc_bulk(i,j)*Rho_h(i,j) - ...
        Dis_conc_wall(i,j)*Rho_h(i,j)); %*'C(T_H_bulk

    %F_wax is the wax mass fraction of precipitation usually 50%-90%
    F_wax=1;


    %%%%%%Depo. thickness calc.
    %Wax resistance next liquid step
    % added depo_thick(i,j) in calc in order to add the previous
    %deposition *RHO is deposit not bulk in [g/cm^3] actually in ...
        [kg/m^3] but multiply by 1000
    Depo_thick(i,j+1) = Depo_thick(i,j) + 0.00055*J_A(i,j)*1/ ...
        (rho_w*1000*F_wax)*Time_seg_to_next_FLUIDseg(i,j);
%%%%%%%%%%%%%%%%%%%%
```

A.9 Wax Deposition Model -RRR

```
%mass_transfer_RRR.m

%The RRR model
%Note: Not summing all wax components rather taking the average

%D_AB (diffusion coef.) and V_A (Molar Volume) is calculated/found in
%diffusion_coefficient.m

%assume Density for wax layer to be about(just for Singh model):
rho_w=0.909; %[g/cm^3]

%RRR MODEL
%%%%%%%%%%%%%%%%%%%%
    %Molecular Diffusion
    %*' Here the calculations are not done for each wax component, ...
       rather wax
    %as a whole.
    %*' multiply by Rho to get concentration in [kg/m^3]
    %*' SIMPLYFIED: Rho at wall is different, but use Bulk value. ...
       Can change

    %Laminar sublayer/viscous sublayer thickness
    d_sublayer(i,j)=58*D_wax(i,j)*Re_h(i,j)^(-7/8);%[m]

    %*'F_wax is the wax mass fraction of precipitation usually 50%-90%
    F_wax=1;

    %Deposition thickness due to Molecular diffusion
    %Depo_thicness=
    %(D_wax*(del_consentration*rho)*S_wetted_frac/ ...
       (Δ_lamsub*density_wax))* Time
    %(wax density (above) is in [g/cm^3] multiply by 1000 to get ...
       [kg/m^3])
    Depo_thick(i,j+1)= Depo_thick(i,j) + (( D_AB(i,j)* ...
       (Dis_conc_bulk(i,j)*Rho_h(i,j)-Dis_conc_wall(i,j)*Rho_h(i,j)) ...
       *1 )/(d_sublayer(i,j)*rho_w*1000) )*1/F_wax ...
       *Time_seg_to_next_FLUIDseg(i,j);
%%%%%%%%%%%%%%%%%%%%
```

A.10 Wax Deposition Model -Matzain

```
%mass_transfer_matzain.m

%The Matzain model
%D_AB (diffusion coef.) and V_A (Molar Volume) is calculated/found in
%diffusion_coefficient.m

%MATZAIN MODEL
%%%%%%%%%%%%%%%%%%%%
    %Matzain constant, C1, C2 and C3
     mat_const1= 15;
     mat_const2= 0.055;
%     mat_const1= 15;
%     mat_const2= 0.1;
  %
%     mat_const1= 9;
%     mat_const2= 7.040;
  %
%     mat_const1= 6.6;
%     mat_const2= 5.060;
    mat_const3= 1.4;
    % mat_const3= 1;
    %Matzain Porosity (oil fraction in the wax layer)
    %*' Reynolds number with viscosity at wall temp!!
    Mat_c0(i,j)= (1- (Re_h_wall(i,j)^0.15 / 8)); %[-] N_RE,f=Re_h ...
        where D_wax is used
    %*' alternative eq.
    % Mat_c0(i,j)= 0.1;
    %Single phase flow
    %Vel_h found in time calc
    N_SR(i,j)= ...
        Rho_h(i,j)*vel_h(i,j)*Depo_thick(i,j)/Vis_h_wall(i,j);  ...
        %same as "regular" Re but with Diameter as Depo_thickness

    %Capital Greek letter pi: empirical relations
    pi_const1(i,j)= mat_const1 / (1 - Mat_c0(i,j)); %where ...
        mat_c0[-] is the porosity( volume fraction of oil in wax film)
    pi_const2(i,j)= mat_const2 * N_SR(i,j)^(mat_const3);

    %*' alternative eq.
    % pi_const2(i,j)= 1+ mat_const2 * N_SR(i,j)^(mat_const3);
    % Note! here Concentration is not multiplied by rho. as it is ...
        wanted in
    % [-]
    Depo_thick(i,j+1)= Depo_thick(i,j) +0.013*( ( ...
        ((pi_const1(i,j))/(1+pi_const2(i,j))) * ...
        D_AB(i,j)*(Dis_conc_bulk(i,j)- Dis_conc_wall(i,j)) ...
        *H_h_wall(i,j) )/(K_h(i,j)) )*Time_seg_to_next_FLUIDseg(i,j);
%%%%%%%%%%%%%%%%%%%%
```

A.11 Wax Deposition Model -Heat Analogy

```
%mass_transfer_HeatAnalogy.m

%Heat analogy model
%D_AB (diffusion coef.) and V_A (Molar Volume) is calculated/found in
%diffusion_coefficient.m

%lewisnumber
Le(i,j)= K_h(i,j)/( Rho_h(i,j)*Cp_h(i,j)*D_AB(i,j) );

%Mass transfer
K_massTransfer(i,j)= H_h_wall(i,j)/ ( Rho_h(i,j)*Cp_h(i,j)* ...
    (Le(i,j))^(2/3) );

 %Matzain constant, C1, C2 and C3
     mat_const1= 1; % diffusion enhancement is not needed for heat ...
         analogy as it is in the Matzain where C1 is 15
     mat_const2= 0.055;
%     mat_const1= 9;
%     mat_const2= 7.040;
%     mat_const1= 6.6;
%     mat_const2= 5.060;
     mat_const3= 1.4;

    %Matzain Porosity (oil fraction in the wax layer)
    %*' Reynolds number with viscosity at wall temp!!
    Mat_c0(i,j)= (1- (Re_h_wall(i,j)^0.15 / 8)); %[-] N_RE,f=Re_h ...
        where D_wax is used
    %*' alternative eq.
    % Mat_c0(i,j)= 0.1;
    %Single phase flow
    N_SR(i,j)= Rho_h(i,j)*vel_h(i,j)*Depo_thick(i,j)/Vis_h(i,j);  ...
        %same as "regular" Re but with Diameter as Depo_thickness

    %Capital Greek letter pi: empirical relations
    pi_const1(i,j)= mat_const1 / (1 - Mat_c0(i,j)); %where ...
        mat_c0[-] is the porosity( volume fraction of oil in wax film)
    pi_const2(i,j)= mat_const2 * N_SR(i,j)^(mat_const3);

%Deposition thickness
Depo_thick(i,j+1)= Depo_thick(i,j) + ...
    0.034*(pi_const1(i,j))/(1+pi_const2(i,j))*( K_massTransfer(i,j)* ...
    (Dis_conc_bulk(i,j)-Dis_conc_wall(i,j)) ...
    )*Time_seg_to_next_FLUIDseg(i,j);
```

A.12 Plot -Heat capacity

```
%plot_neqSim_cp.m

%see Neqsim_get_fluid_values.m file for calculations
hold on
grid on
% title('Regression of NeqSim Values');
plot(temp_fluid_MIN-273.15:stepsize_eq:temp_fluid_MAX-273.15,P_cp_eq);
xlabel('Temperature [Deg C]')
ylabel('Heat capacity [J/kg*K]')

plot(temp_fluid_MIN-273.15:1:temp_fluid_MAX-273.15,Cp_h, 'x');
xlabel('Temperature [Deg C]')
ylabel('Heat capacity [J/kg*K]')

legend('NeqSim Regression','NeqSim Values')
```

A.13 Plot -Conductivity

```
%plot_neqSim_fluid_conductivity.m

%see Neqsim_get_fluid_values.m file for calculations
hold on
grid on
% title('Regression of NeqSim Values');
plot(temp_fluid_MIN-273.15:stepsize_eq:temp_fluid_MAX-273.15,P_k_eq);
xlabel('Temperature [Deg C]')
ylabel(' Conductivity [J/kg*K]')

plot(temp_fluid_MIN-273.15:1:temp_fluid_MAX-273.15,K_h, 'x');
xlabel('Temperature [Deg C]')
ylabel('Conductivity [J/kg*K]')

legend('NeqSim Regression','NeqSim Values')
```

A.14 Plot -Density

```
%plot_neqSim_density.m

%see Neqsim_get_fluid_values.m file for calculations
hold on
grid on
% title('Regression of NeqSim Values');
plot(temp_fluid_MIN-273.15:stepsize_eq:temp_fluid_MAX-273.15,P_rho_eq);
xlabel('Temperature [Deg C]')
ylabel('Denisty [kg/m^3]')

plot(temp_fluid_MIN-273.15:1:temp_fluid_MAX-273.15,Rho_h, 'x');
xlabel('Temperature [Deg C]')
ylabel('Density [kg/m^3]')

density_ex_values=[836.9, 825.8, 817.6, 802.7, 788.0];
density_ex_temp= [5, 10, 20 , 40, 60];
plot(density_ex_temp,density_ex_values,'*-')

legend('NeqSim Regression','NeqSim Values','Experimental values')
```

A.15 Plot -Viscosity

```
%plot_neqSim_viscosity.m

%NB! Viscosity has 4 parameter values in its regression
%see Neqsim_get_fluid_values.m file for calculations
hold on
grid on
% title('Regression of NeqSim Values');
plot(temp_fluid_MIN-273.15:stepsize_eq:temp_fluid_MAX-273.15, ...
    P_vis_eq*10^3);
% xlabel('Temperature [Deg C]')
% ylabel('Viscosity [mPa*S]')
plot(temp_fluid_MIN-273.15:1:temp_fluid_MAX-273.15,Vis_h*10^3, 'x');

viscosity_ex_temp=[60, 56, 50, 45, 40, 35];
viscosity_ex_values = [ 1.26, 1.34, 1.48, 1.61, 1.76, 1.94 ];
plot(viscosity_ex_temp,viscosity_ex_values,'*-')

xlabel('Temperature [Deg C]')
ylabel('Viscosity [mPa*s]')
legend('NeqSim Regression','NeqSim Values','Experimental values')
```

A.16 Plot -Wax Thickness Vs. Time Point Model

```
%ThicknessVsTimePointModel.m

%dont want to plot for first liquid segment because
Total_Time_seg_to_nex_FLUIDseg(1,1)=0; %start at zero seconds
%calulates time elapsed when each fluid segment is in the pipe.
for thickVsTimeCounter=2:j+1 %total_num_FLUID_seg+1 <-- this will ...
    not work if pipe is blocked
     Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter)= ...
         Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter-1)+ ...
         Total_time_vec(1,thickVsTimeCounter-1);
end
%plot time vs thickness
plot( Total_Time_seg_to_nex_FLUIDseg(1:total_num_pip_seg,1:j+1)/ ...
    (60*60),Depo_thick(1:total_num_pip_seg,1:j+1)*10^3 )

xlab1=xlabel('Time [hr]');
ylab1=ylabel('Deposition Thickness [mm]');

set(xlab1,'FontSize',15,'fontweight','bold');
set(ylab1,'FontSize',15,'fontweight','bold');

% ylim([0 3])

grid on;
```

A.17 Plot -Multiple Point Model's Wax Thickness

```
%plot_several.m

%plot_several.
%Note! .mat files have to the same folder
%%
load('TEMPMatzain_1to50_Run_1.0.mat')
MAT_Total_Time_seg_to_nex_FLUIDseg(1,1)=0; %start at zero seconds
%calulates time elapsed when each fluid segment is in the pipe.
for thickVsTimeCounter=2:j+1 %total_num_FLUID_seg+1 <-- this will ...
    not work if pipe is blocked
     MAT_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter)= ...
         MAT_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter-1)+ ...
         Total_time_vec(1,thickVsTimeCounter-1);
end
data1_matzain1to10=Depo_thick;
%%
load('TEMPHeatAnalogy_1to50_Run_1.0.mat')
HEAT_Total_Time_seg_to_nex_FLUIDseg(1,1)=0; %start at zero seconds
%calulates time elapsed when each fluid segment is in the pipe.
for thickVsTimeCounter=2:j+1 %total_num_FLUID_seg+1 <-- this will ...
    not work if pipe is blocked
     HEAT_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter)= ...
         HEAT_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter-1)+ ...
         Total_time_vec(1,thickVsTimeCounter-1);
end
data2_heat1to50=Depo_thick;
%%
load('TEMPRRR_1to50_Run_1.0.mat')
RRR_Total_Time_seg_to_nex_FLUIDseg(1,1)=0; %start at zero seconds
%calulates time elapsed when each fluid segment is in the pipe.
for thickVsTimeCounter=2:j+1 %total_num_FLUID_seg+1 <-- this will ...
    not work if pipe is blocked
     RRR_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter)= ...
         RRR_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter-1)+ ...
         Total_time_vec(1,thickVsTimeCounter-1);
end
data3_RRR1to50=Depo_thick;
% %%
% load('TEMPSingh_1to50_Run_1.0.mat')
% SIN_Total_Time_seg_to_nex_FLUIDseg(1,1)=0; %start at zero seconds
% %calulates time elapsed when each fluid segment is in the pipe.
% for thickVsTimeCounter=2:j+1 %total_num_FLUID_seg+1 <-- this will ...
    not work if pipe is blocked
%     SIN_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter)= ...
    SIN_Total_Time_seg_to_nex_FLUIDseg(1,thickVsTimeCounter-1)+ ...
    Total_time_vec(1,thickVsTimeCounter-1);
% end
% data4_SIN1to50=Depo_thick;
%%
```

```
38
39 % From Plot: ThicknessVsTime plot
40 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
41 %Time vs thickness Point model
42
43 %plot time vs thickness
44 hold on
45 plot( ...
      MAT_Total_Time_seg_to_nex_FLUIDseg(1:total_num_pip_seg,1:j+1)/ ...
      (60*60),data1_matzain1to10(1:total_num_pip_seg,1:j+1)*10^3 )
46 plot( ...
      HEAT_Total_Time_seg_to_nex_FLUIDseg(1:total_num_pip_seg,1:j+1)/ ...
      (60*60),data2_heat1to50(1:total_num_pip_seg,1:j+1)*10^3 )
47 plot( ...
      RRR_Total_Time_seg_to_nex_FLUIDseg(1:total_num_pip_seg,1:j+1)/ ...
      (60*60),data3_RRR1to50(1:total_num_pip_seg,1:j+1)*10^3 )
48 %plot( ...
      SIN_Total_Time_seg_to_nex_FLUIDseg(1:total_num_pip_seg,1:j+1)/ ...
      (60*60),data3_SIN1to50(1:total_num_pip_seg,1:j+1)*10^3 )
49
50 xlab1=xlabel('Time [hr]');
51 ylab1=ylabel('Deposition Thickness [mm]');
52
53 set(xlab1,'FontSize',15,'fontweight','bold');
54 set(ylab1,'FontSize',15,'fontweight','bold');
55
56 legend('Matzain','Heat Analogy', 'RRR')
57
58 % ylim([0 3])
59
60 grid on;
```

A.18 Plot -Wax Molar Volume

```
%plot_neqsim_molarVolume_VA.m

%see Neqsim_get_fluid_values.m file for calculations
hold on
grid on
% title('Regression of NeqSim Values');
plot(temp_fluid_MIN-273.15:stepsize_eq:temp_fluid_MAX-273.15,P_VA_eq);
% xlabel('Temperature [Deg C]')
% ylabel('Molar Volume [cm^3/mol]')

plot(temp_fluid_MIN-273.15:1:temp_fluid_MAX-273.15,V_A, 'x');
xlabel('Temperature [Deg C]')
ylabel('Molar Volume [cm^3/mol]')

legend('NeqSim Regression','NeqSim Values')
```

A.19 Plot -Various Values

```
%Plot_thicknesscurve.m

 plot(1:total_num_pip_seg,Depo_thick(1:total_num_pip_seg, ...
     2:100:j+1)*10^3)  %2:j-1 j+1 because e.g. if there is one fluid ...
     segment then the thickness is added to the second j cell in ...
     Depo_thickness

%point model, added ,'o' because only a point
%plot(1:total_num_pip_seg,Depo_thick(1:total_num_pip_seg, ...
    2:100:j+1) *10^3,'o')%NBe.g.:10: STEP

xlab1=xlabel('Pipe Segment');
ylab1=ylabel('Deposition Thickness [mm]');

set(xlab1,'FontSize',15,'fontweight','bold');
set(ylab1,'FontSize',15,'fontweight','bold');

% plot(D_wax(1:total_num_pip_seg,2:j))

% %Temperature
% plot(1:total_num_pip_seg,T_h_bulk(1:total_num_pip_seg,2:j)-273.15)

% %Check wall and bulk temperature values
% hold on
% grid on
% plot(1:total_num_pip_seg,T_h_bulk(1:total_num_pip_seg,1:j)-273); ...
    %2:j-1
% plot(1:total_num_pip_seg,T_h_wall(1:total_num_pip_seg,1:j)-273); ...
    %2:j-1
```

```
24 %
25
26 %plot two axis
27 % hold on
28 % plot(1:total_num_pip_seg,T_h_bulk(1:total_num_pip_seg,2)-273); %2:j-1
29 % %TWO AXIS:
30 % plotyy(1:total_num_pip_seg,T_h_wall(1:total_num_pip_seg,j-1)-273, ...
       1:total_num_pip_seg,Depo_thick(1:total_num_pip_seg,j-1)); %2:j-1
31
32 %Pressure drop
33 % plot(1:total_num_pip_seg,P_h)
34
35 % plot(1:total_num_pip_seg,D_AB(1:total_num_pip_seg,2:j))
36
37 %Net Concentraition beween wall and bulk
38 % plot(1:total_num_pip_seg,(Dis_conc_bulk(1:total_num_pip_seg,2:j)- ...
       Dis_conc_wall(1:total_num_pip_seg,2:j)))
39 grid on
```

A.20 NeqSim -Initialise

```
%Akul InitNeqSim

count_dir=1;


tempClassPath = javaclasspath;
%if isempty(tempClassPath)

    %AKUL EDIT:
%     %Takes the txt file and makes a vector
%     ...
    directory_txt=importdata('Akul_DIRECTORY_NEQSIM.txt');%importdata ...
    takes the data and splits the /n(enter) into a new cell
%     %strjoin joins the cells and here, is added
%     cellarraywithpaths=({strjoin(directory_txt,', ')})

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    rootpath = '/Users/Akul/Documents/NTNU 2015 11 Master/neqsim/ext/';
    direc=dir([rootpath '*.jar']);%gives all the *.jar files in ...
       directory in rootpath. Can call the vector and output the ...
       name by writing vectordirname(counter).name counter e.g. 1

    size_direc=size(direc);%find number of *.jar files

    cellarraywithpaths=cell(1,size_direc(1)); %creates empty cells, ...
       e.i an array with a given size

    for count_dir=1:size_direc(1) %size_direc outputs a vector just ...
       want first value, as this is the number of files

        cellarraywithpaths{count_dir}=[strcat(rootpath, ...
           direc(count_dir).name)]; %stract joins to strings .name ...
           calls the name part of direc array
    end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    for i = 1:length(cellarraywithpaths)
        %cellarraywithpaths{end+1}=[rootpath a(i,:)];
        %javaaddpath(a(i,:));
    end
    javaclasspath(cellarraywithpaths);
%end

import thermo.*;
import thermo.system.*;
import PVTsimulation.simulation.*;
import thermodynamicOperations.*;
import processSimulation.measurementDevice.*;
import processSimulation.controllerDevice.*;
```

```
42 import processSimulation.processSystem.*;
43 import processSimulation.processEquipment.separator.*;
44 import processSimulation.processEquipment.*;
45 import processSimulation.processEquipment.mixer.*;
46 import processSimulation.processEquipment.stream.*;
47 import processSimulation.processEquipment.heatExchanger.*;
48 import processSimulation.processEquipment.compressor.*;
49 import processSimulation.processEquipment.valve.*;
50 import processSimulation.processEquipment.splitter.*;
51 import processSimulation.processEquipment.absorber.*;
52 import processSimulation.processEquipment.absorber.*;
53 import processSimulation.processEquipment.util.*;
54
55 if ¬exist('processOperations') % added Even 10/12/2011
56     global processOperations
57     processOperations = ProcessSystem;
58 end % end added
```

A.21 NeqSim- Get Fluid Data

```
%Neqsim_get_fluid_values.m

temp_fluid_MIN=253.15;
temp_fluid_MAX=373.15;
counter_fluid_prop=1;
%correction factor for viscosity
system1.getPhase('oil').getPhysicalProperties(). ...
    getViscosityModel().setTBPviscosityCorrection(  0.314);

for temp_fluid_prop= temp_fluid_MIN:temp_fluid_MAX

    system1.setTemperature(temp_fluid_prop);
    system1.setPressure(100.0); %Assume constant pressure
    system1.init(3);
    system1.initPhysicalProperties();

    % wtfracWax_plot(counter_fluid_prop) = ...
        system1.getWtFraction(system1.getPhaseNumberOfPhase('wax'));
    Vis_h(counter_fluid_prop)= ...
        system1.getPhase('oil').getPhysicalProperties().getViscosity();
    K_h(counter_fluid_prop) = system1.getPhase('oil'). ...
        getPhysicalProperties().getConductivity();
    Rho_h(counter_fluid_prop) = ...
        system1.getPhase('oil').getPhysicalProperties().getDensity();

    %Have to split into denominator and numerator
    Cp_h_denominator(counter_fluid_prop)= ...
        (system1.getPhase('oil').getNumberOfMolesInPhase()) * ...
        (system1.getPhase('oil').getMolarMass());
    Cp_h(counter_fluid_prop) = system1.getPhase('oil'). ...
        getCp()/Cp_h_denominator(counter_fluid_prop) ;%[J/kg*k]

    %if system1.hasPhase('wax')==1
        %Molar volume
        V_A(counter_fluid_prop)= ...
            system1.getPhase('wax').getMolarVolume()*10; %[cm^3/mol]
    %end

    counter_fluid_prop= counter_fluid_prop + 1;
end

%neqSim regression

%polyfit yields the second degree equation? regression,
%P_rho= ax^8+bx^7...+hx+i output values are a,b and c
%NOTE, viscosity has 4 parameters a,b,c and d to get more accurate
%regression
[P_rho] = polyfit(temp_fluid_MIN:1:temp_fluid_MAX,Rho_h,3);
[P_vis] = polyfit(temp_fluid_MIN:1:temp_fluid_MAX,Vis_h,4);
```

```
41 [P_cp] = polyfit(temp_fluid_MIN:1:temp_fluid_MAX,Cp_h,3);
42 [P_k] = polyfit(temp_fluid_MIN:1:temp_fluid_MAX,K_h,3);
43 [P_VA] = polyfit(temp_fluid_MIN:1:temp_fluid_MAX,V_A,3);
44
45 stepsize_eq=1;
46 %calculate multiple values P_rho= ax^8+bx^7...+hx+i for a step size ...
      of 0.1
47 P_rho_eq=polyval(P_rho,temp_fluid_MIN:stepsize_eq:temp_fluid_MAX);
48 P_vis_eq=polyval(P_vis,temp_fluid_MIN:stepsize_eq:temp_fluid_MAX);
49 P_cp_eq=polyval(P_cp,temp_fluid_MIN:stepsize_eq:temp_fluid_MAX);
50 P_k_eq=polyval(P_k,temp_fluid_MIN:stepsize_eq:temp_fluid_MAX);
51 P_VA_eq=polyval(P_VA,temp_fluid_MIN:stepsize_eq:temp_fluid_MAX);
```

A.22 NeqSim -Tune Wax Deposition

```
%Akul_ex_wax.m

%NB! density in g/cm^3
clear all
Akul_InitNeqSim
%NOTE! first run? then need to comment out "set values"
%set values from previous runs:
% newParams=[0.7126628737, 0.0002374445, -0.0383634293];
% newHfusparam= 0.3139609267;
% newTripeTparam= 1.2570707981;
%InitNeqSim
% processOperations.clearAll
system1 = SystemSrkEos(273,10);
%system1.addComponent('methane', 10.0);
%system1.addComponent('ethane', 10.0);
system1.addComponent('propane', 0.355);
system1.addComponent('i-butane', 0.448);
system1.addComponent('n-butane', 0.23664);
system1.addComponent('i-pentane', 2.031);
system1.addComponent('n-pentane', 3.457);
%Read excel file and split into num,txt and raw. num contains alle the
%numbers and raw txt and numbers
%%%%%%%%%%%%%%%%
 %Note: that the comma in excel must be , not .
    [num,txt,raw] = xlsread('statoil_fluid_composition.xlsx');
for count_neq=1:31
    %NB! last value is density the excel sheet value is in ...
       [kg/m^3], but
    %neqsim input has to be in [g/cm^3]
    %strcat: Add the name of component C6,C7 etc. , num(x,2) ...
       contains the
    %mol%,
    system1.addTBPfraction(strcat('C',num2str(5+count_neq)), ...
       num(count_neq,2), num(count_neq,3)/1000.0, ...
       num(count_neq,4)/1000); %mol/sec , molmass[kg/mol], ...
       density[g/cm^3]
end

%Characterise plussfraction
system1.setHeavyTBPfractionAsPlusFraction();
system1.getCharacterization().getLumpingModel(). ...
   setNumberOfLumpedComponents(6)
system1.getCharacterization().characterisePlusFraction();

%Set values, this should be commented at first run
% system1.getWaxModel().setWaxParameters(newParams);
% system1.getWaxModel().setParameterWaxHeatOfFusion(newHfusparam);
% system1.getWaxModel().setParameterWaxTriplePointTemperature ...
   (newTripeTparam);
```

```
%%%%%%%%%%%%%%%%%%%
system1.getWaxModel().addTBPWax();
system1.createDatabase(1);

system1.addSolidComplexPhase('wax');
system1.setMultiphaseWaxCheck(1);
system1.setMultiPhaseCheck(1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
system1.setTemperature(280.0);
system1.setPressure(5.0);
TPflash(system1,0);
%calcWAT(system1);
system1.getTemperature;
waxfraction=system1.getPhase('oil') ...
    .getWtFractionOfWaxFormingComponents();
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%wtfracWax = ...
    system1.getPhase('wax').getNumberOfMolesInPhase()*system1. ...
    getPhase('wax').getMolarMass()/ ...
    (system1.getMolarMass()*system1.getNumberOfMoles());
wtfracWax = ...
    system1.getWtFraction(system1.getPhaseNumberOfPhase('wax'));

temperatures = [ 305, 298.15, 293.15, 288.0, 285.5, 283.0, 273, ...
    270.5, 268, 263, 260.5, 253];
pressures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, ...
    1.0];
expdata = [ 0, 0.05, 0.21, 0.77, 1.21, 1.66 ,3.2, 3.52, 3.79, 4.22, ...
    4.36, 4.5 ;0, 0,0,0,0,0,0,0,0,0,0,0];
%temperatures = [ 288.0, 285.5, 283.0, 273];
%pressures = [1.0, 1.0, 1.0, 1.0];
%expdata = [ 0.77, 1.21, 1.66 ,3.2 ;0,0,0,0];

% temperatures = [ 288.0, 273];
% pressures = [1.0, 1.0];
% expdata = [ 0.77, 3.2 ;0,0];
% temperatures = [ 295, 283.0, 273, 270.5, 268];
% pressures = [ 5, 5, 5.0, 5.0, 5.0];
% expdata = [ 0.1, 1.66 ,3.2, 3.52, 3.79;0,0,0,0,0];
optim = WaxFractionSim(system1);
optim.setTemperaturesAndPressures(temperatures, pressures);
optim.setExperimentalData(expdata);
optim.getOptimizer().setNumberOfTuningParameters(5);
%Number of iterations:
optim.getOptimizer().setMaxNumberOfIterations(1);
%no iteration? comment next line:
optim.runTuning();
newParams = system1.getWaxModel().getWaxParameters();
newHfusparam = system1.getWaxModel().getParameterWaxHeatOfFusion();
newTripeTparam = ...
    system1.getWaxModel().getParameterWaxTriplePointTemperature();
```

A.23 NeqSim -Wax Deposition regression

```
%wax_dep_regression.m

%plot and regression of Dissolved wax in the hot fluid
%Using experimental values provided by statoil

%wax deposition matrix, Wax_dep(Temp,precipitated)
Wax_dep = dlmread('Wax_deposition_temp_vs_deposited_4,5%.txt');

%Change unit to kelvin by adding 273.15
Wax_dep_T=273.15+Wax_dep(:,1);

%NOTE! 4.5 minus because this valie gives the amount dissolved, and the
%value in the txt file is the amount precipitated
Wax_dep_dissolved=4.5-Wax_dep(:,2);

%polyfit yields the second degree equation? regression,
%P_dep= ax^8+bx^7...+hx+i output values are a,b and c
[P_dep] = polyfit(Wax_dep_T,Wax_dep_dissolved,3);

% plot(Wax_dep_T,Wax_dep_dissolved,'o');
% hold on;

%find the max and min temp. values, to use in plot
T_value_min= min(Wax_dep_T);
T_value_max= max(Wax_dep_T);
%findin max depo value
Wax_dep_dissolved_max=max(Wax_dep_dissolved);

%calculate multiple values P_dep= ax^8+bx^7...+hx+i for a step size ...
    of 0.1
P_depo_eq=polyval(P_dep,T_value_min:0.1:T_value_max);
% hold on
% %can visually see if the data points 'o' and equation line match also
% %*'possible to calculate R^2
% %%
% % title('Regression of Experimental Values');
% plot(T_value_min-273:0.1:T_value_max-273,P_depo_eq);
% xlabel('Temperature [Deg C]')
% ylabel('Dissolved Wax in Oil [%]')
%
% %%
% % title('Experimental Values');
% plot(Wax_dep(:,1),Wax_dep_dissolved(:),'*-');
% xlabel('Temperature [Deg C]');
% ylabel('Dissolved Wax in Oil [%]');
%
% legend('Regression Values','Experimental Values')
%%
%precipitated wax
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hold on
grid on
%can visually see if the data points 'o' and equation line match also
%*'possible to calculate R^2

% title('Regression of Experimental Values');
plot(T_value_min-273.15:0.1:T_value_max-273.15,4.5-P_depo_eq,'-.');
xlabel('Temperature [Deg C]')
ylabel('Precipitated Wax in Oil [%]')
%
% title('Experimental Values');
plot(Wax_dep(:,1),4.5-Wax_dep_dissolved(:),'*-');
xlabel('Temperature [Deg C]');
ylabel('Precipitated Wax in Oil [%]');

legend('Regression Values','Experimental Values')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.24 Neqsim -Plot Wax values

```
%Akul_plot_ex_wax.m

%NB! need to run ex_wax first!
count_neq_wax=0;
temp_neq_min=253;
temp_neq_max=453;
stepSize=0.1;
flagbreak=0;
%%%%%%%%%%%%%%%
for temp_neq= temp_neq_min:stepSize:temp_neq_max
    count_neq_wax= count_neq_wax+1;

    system1.setTemperature(temp_neq);
    system1.setPressure(5.0);
    TPflash(system1,0); %tp flash(system1, Display=1 or not=0!)
    system1.initPhysicalProperties();

        if system1.hasPhaseType('wax')==1
         wtfracWax_plot(count_neq_wax) = ...
             system1.getWtFraction(system1.getPhaseNumberOfPhase('wax'));
        else
        wtfracWax_plot(count_neq_wax) = 0;

        flagbreak=1
        %uncomment if you want to know the WAT
        % T_WAT=temp_neq
        break;
        end

        if flagbreak==1
            break;
        end

% To print to check values
% system1.setTemperature(273);
% system1.setPressure(5.0);
% TPflash(system1,0); %tp flash(system1, Display=1 or not=0!)
% wtfracWax_plot = ...
    system1.getWtFraction(system1.getPhaseNumberOfPhase('wax'))

end

    %completing rest of loop with values equal to zero.
for newLoop=temp_neq:stepSize:temp_neq_max - stepSize %have to ...
    subtract by a stepsize because counter +1 below will add one ...
    extra cell
    count_neq_wax= count_neq_wax + 1;
    wtfracWax_plot(count_neq_wax) = 0;
end
```

```
46          %parameter values that make up the regression equation
47           P_waxdepo=spline(temp_neq_min:stepSize:temp_neq_max, ...
                 wtfracWax_plot);
48           %This calls the equation for various temps
49           P_waxdepo_eq=ppval(P_waxdepo, ...
                 temp_neq_min:stepSize:temp_neq_max)*100;
50           %Save .mat file containing the regression values outputs ...
                 the P_waxdepo
51           % save('P_waxdepo_neqsim_eq.mat','P_waxdepo')
52
53  %%%%%%%%%%%
54  % %creating a regression of wax depo from neqsim
55  % [P_neq_wax] = ...
        polyfit(temp_neq_min:1:temp_neq_max,wtfracWax_plot*100,4);
56  % P_neq_wax_eq=polyval(P_neq_wax,temp_neq_min:0.1:temp_neq_max);
57  % hold on
58  % % title('Regression of NeqSim Values');
59  % plot(temp_neq_min-273:0.1:temp_neq_max-273,P_neq_wax_eq);
60  % xlabel('Temperature [Deg C]')
61  % ylabel('wax depo [m]')
62  % %%%%%%%%%%
63
64  hold on
65  %note -273->convert to celcius and *100 -> convert to "in per cent"
66  plot((temp_neq_min:stepSize:temp_neq_max)-273.15, ...
        ppval(P_waxdepo,temp_neq_min:stepSize:temp_neq_max)*100);
67
68  % plot(temp_neq_min-273:temp_neq_max-273, 4.5 - ...
        wtfracWax_plot(1:count_neq)*100)
69  %plot(temp_neq_min-273.15:stepSize:temp_neq-273.15, ...
        wtfracWax_plot(1:count_neq_wax)*100,'x')
70  plot((temp_neq_min:stepSize:temp_neq_max)-273.15, ...
        wtfracWax_plot(1:count_neq_wax)*100,'x')
71  % plot(temp_neq_min:temp_neq_max,  wtfracWax_plot(1:count_neq)*100)
72  legend('Regression of Experimental Values','Experimental Values', ...
        'NeqSim-Regression','NeqSim Wax Values')
```

A.25 Neqsim -Tune Viscosity values

```
%ex_wax_visc.m

%NB! density in g/cm^3
% clear all

Akul_InitNeqSim
processOperations.clearAll

system1 = SystemSrkEos(273,10);
%system1.addComponent('methane', 10.0);
%system1.addComponent('ethane', 10.0);
system1.addComponent('propane', 0.355);
system1.addComponent('i-butane', 0.448);
system1.addComponent('n-butane', 0.23664);
system1.addComponent('i-pentane', 2.031);
system1.addComponent('n-pentane', 3.457);

%Read excel file and split into num,txt and raw. num contains alle the
%numbers and raw txt and numbers
%%%%%%%%%%%%%%%%%
 %Note: that the comma in excel must be , not .
    [num,txt,raw] = xlsread('statoil_fluid_composition.xlsx');
for count_neq=1:31

    %NB! last value is density the excel sheet value is in ...
       [kg/m^3], but
    %NeqSim input has to be in [g/cm^3]
    %strcat: Add the name of component C6,C7 etc. , num(x,2) ...
       contains the
    %mol%,
    system1.addTBPfraction(strcat('C',num2str(5+count_neq)), ...
       num(count_neq,2), num(count_neq,3)/1000.0, ...
       num(count_neq,4)/1000); %mol/sec , molmass[kg/mol], ...
       density[g/cm^3]
end

%Characterise pluss fraction
system1.setHeavyTBPfractionAsPlusFraction();
system1.getCharacterization().getLumpingModel(). ...
   setNumberOfLumpedComponents(6)
system1.getCharacterization().characterisePlusFraction();

system1.getWaxModel().setWaxParameters(newParams);
system1.getWaxModel(). setParameterWaxHeatOfFusion(newHfusparam);
system1.getWaxModel(). ...
   setParameterWaxTriplePointTemperature(newTripeTparam);
%corrVisc = system1.getPhase('oil').getPhysicalProperties(). ...
   getViscosityModel().getTBPviscosityCorrection();
%%%%%%%%%%%%%%%%%%%
```

```
system1.getWaxModel().addTBPWax();
system1.createDatabase(1);
system1.addSolidComplexPhase('wax');
system1.setMultiphaseWaxCheck(1);
system1.setMultiPhaseCheck(1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
system1.getPhase('oil').getPhysicalProperties(). ...
    getViscosityModel().setTBPviscosityCorrection(0.3807540450);
system1.setTemperature(280.0);
system1.setPressure(5.0);
TPflash(system1,1);
system1.initPhysicalProperties();
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

temperatures = [60, 56, 50, 45, 40, 35]+273.15;
pressures = [5.0, 5.0, 5.0, 5.0, 5.0, 5.0];
expdata = [ 1.26, 1.34, 1.48, 1.61, 1.76, 1.94;0,0,0,0,0,0]*1e-3;

optim = ViscositySim(system1);
optim.setTemperaturesAndPressures(temperatures, pressures);
optim.setExperimentalData(expdata);
optim.getOptimizer().setMaxNumberOfIterations(10);
optim.runTuning();
%optim.runCalc();
hoilVisc  = optim.getOilViscosity();
viscCorrFact= system1.getPhase('oil').getPhysicalProperties(). ...
    getViscosityModel().getTBPviscosityCorrection();
```