Master Thesis

Optimization of well Start – up

By Jörg Michael Six

MONTAN UNIVERSITÄT University of Leoben
Dept. of Petroleum Production Engineering

Supervised by: Univ.-Prof.Dipl.-Ing.Dr.mont. Herbert Hofstätter, University of Leoben, Austria
Prof. Curtis H Whitson, NTNU Trondheim, Norway

Approval date: 10th June 2010

"Dedicated to my family"

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich diese Arbeit selbstständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfsmittel bedient habe.

AFFIDAVIT

I declare in lieu of oath, that I wrote this thesis and performed the associated research myself, using only literature cited in this volume.

Datum

Unterschrift

Table of Contents

List of Figures

Acknowledgement

I would like to express my sincerest thanks to my thesis supervisor Prof. Curtis Hays Whitson from the NTNU Department of Petroleum Technology and Applied Geophysics for the academic and personal guidance he provided. I have to admit that this thesis was the most challenging tasks to fulfil during my entire academic career. Especially in the beginning it was very troublesome and I really struggled to make any progress. During this period I really depended on the guidance of Prof. Whitson. And I have to say he did a great job. Not only that he helped me with his expertise as a Petroleum Engineer but also for keeping me on track by encouraging and motivating me to find solutions. Moreover he provided me an office place at his company PERA to experience the help of all the people working there. This leads me to my next thank you.

A big thank you to all people working at PERA for the warm welcome when I first came there and for the constant help, thesis related or not, I experienced during my stay. The friendly and yet focussed working environment made it a real pleasure to spent my time there.

Prof. Herbert Hofstätter, Head of the Department for Petroleum Production and Processing at the University of Leoben, has accompanied me for most of my time in Leoben as a Professor and Thesis supervisor. He always supported me wherever he could and also when it came to write my Master thesis in Norway he encouraged me to do it. Thank you for passing your long-time industry experience to me as a student and also for all the offered possibilities that are making my student years memorable.

I also want to say thank you to SPT Group Norway and Coats Engineering for granting me free access to their software.

Last but not least I want to thank my friend and colleague Michael Schietz on which thesis this paper is based on. Not only for bringing me into the game when it came to the point who is going to continue on his work but also for the provided help during my thesis.

Leoben, May 2010

Jörg Michael Six

A. Abstract

This Master of Science Thesis deals with the optimization of well start-up procedures. Coupling of a reservoir model with a well bore flow model is the best way to capture all the complex transient flow phenomena that are predominant during such an operation.

An important aspect when reading this thesis is to know that it is based on the Master of Science Thesis “Optimization of well start-up” by Michael Schietz. It is highly recommended to be familiar with the basic thesis to fully understand the changes and advancements described in this thesis.

At the beginning an overview is given why optimization of well start-up is necessary and which problems can be avoided by doing such. The employed software and special tools offered by the used software to deal with all the phenomena encountered during start-up is described in detail. To achieve an accurate simulation of well start-up a reservoir model, a well bore flow model and an integrated model to combine those two models is necessary. The used software is SENSOR 6k (reservoir simulator), OLGA 6 GUI (well bore simulator) and Pipe-It (software especially designed for integrating and optimizing multi-component processes). The used reservoir model as well as the used well bore flow model is described in detail. Moreover the changes to achieve better results than the basic model are pointed out.

The core of this thesis is the advancement to the basic model. Advancements refer to a more accurate simulation of shut in conditions and to a more complex model to simulate well start-up. The start-up simulation is now able to deal with counter current flow within the reservoir and well bore model as well as in the integrated model. Another achievement is that the model is now able to handle water in the system. It is shown that the advanced model results in better and more consistent simulations. To ensure that the new model is more advanced comparisons to the basic model are shown.

This project has been carried out in close collaboration with the company PERA AS and Curtis H Whitson as supervisor. The used software was kindly provided for free by Petrostreamz AS, SPT Group Norway AS and Coats Engineering Inc. All used sources where stated using the ISO 690 convention for citation, in order to give credit to the authors and make sure that continuative research can start from this papers’ final state of knowledge.

1. Introduction

At very first it has to be pointed out that this Master Thesis is based on the results of the Master Thesis "Optimization of well start-up" [1] by Michael Schietz. It is very important to be familiar with this mentioned Master Thesis to fully understand the problems and simulation strategies that are described in this integrated model.

Well start-up is a procedure that has to be conducted on a regular basis in the oil and gas industry. The initial condition of a well that has to be started up is shut-in. When a well was shut in it is quite clear that it has to be started up if it is supposed to go on production again. There are two main scenarios why a well can be in shut-in condition. The most obvious one is a new well. After the well was drilled and the completion is finished shut-in conditions are predominant. The other scenario is that a well on production has to be shut down. There are several possible reasons for a production shut down:

- maintenance of down-hole or surface facilities or a change in the completion;
  to ensure safe and efficient operations
- well tests;
  to estimate reservoir and well performance sequences of shut-ins and start-ups are required
- well is run as a swing producer;
  e.g. to fill up rest capacities in platform fluid handling systems
- reservoir pressure build-up to kick out accumulated fluids;
  needed in largely depleted or low permeable reservoirs to kick out produced fluids and/or liquid slugs in gas wells
- HSE issues;
  e.g. hurricane warnings

For all these described scenarios the well has to be shut-in. Shutting in a well is not a very difficult procedure to conduct. The difficult part is always bringing the well back on production. The reason is that during the start-up procedure very complex flow phenomena are predominant within the entire system, especially in the region near the perforations down-hole. State-of-the-art procedure for a well start-up is to do it manually. This is more or less a trial and error process because it is not known how down-hole conditions are influenced when changes to the system are introduced on surface; in this case opening the well head choke. Due to that it needs highly experienced people to ensure a successful and efficient well start-up.

Successful refers to an as fast as possible start-up procedure without damaging neither well nor surface facilities. When the tubing head pressure (THP) gets changed pressure waves are generated. Those travel down the tubing to finally reach the bottom of the well and introduce after a certain time delay also a change in bottom-hole pressure (BHP). Subsequent this change in BHP changes the inflow rates into the well. Depending on which kind of completion is used this can cause different problems; especially when fluid velocities, pressures or flow rates alter too fast. Gravel packs can be eroded or contaminated gravel can get mixed with clean gravel caused by high acceleration rates or permanent changes in fluid velocity. Another problem that can crop up in connection with the latter is onset of sand production due to disturbed near well bore formation stresses.

If those problems are encountered the well can be permanently damaged what could lead to impaired production. Also the necessity of a re-completion or sidetracking the well can be caused.

Bottom line all problems that slow down the well start-up or lead to any damage cost a lot of money. The goal is to reduce additional cost by minimizing the start-up time for a well, but without harming the well or surface facilities.

[1] Michael Schietz, Master Thesis "Optimization of well start-up", 2009

Not only damages caused by the start-up can be cost intensive. The procedure of a well start-up can last several hours or even days. Most of the fluids that are produced during the start-up procedure have to be torched or disposed. The reason is the contamination of the well stream with mud, debris or completion fluids from previous drilling or intervention activities. Those contaminations are not allowed to access export pipelines or tankers by all means. When talking about torching and disposing the well stream during the start-up also the penalties and CO_2 taxes, caused by flaring hydrocarbons, have to be considered. Moreover the amount of hydrocarbons produced during the start-up can be considered as lost production and corresponding loss of revenues.

To avoid, or at least reduce, all the inconveniences mentioned above is it quite meaningful to think about an optimization of well start-up. To capture all the flow phenomena that are predominant during the start-up a fully coupled reservoir / well-bore flow model is needed. With such a model is it possible to minimize start-up time without damaging the well or surface facilities.

As already mentioned in the very beginning this paper is the advancement to a Master Thesis. To show improvements that have been made comparisons to the base case are necessary. The term “current” refers always to the model developed by Michael Schietz. Models that have been developed in this thesis are always referred as “new”.

1.1 General modelling and optimization

The difficult task when simulating well start-up is that two different environments have to be covered: reservoir (porous media) and pipe (tubing). Because there is no software that is able to handle the simulation of flow through porous media and also in pipe is it necessary to develop a coupled model to combine a reservoir simulator with a pipe flow simulator.

The used software for this thesis is the reservoir simulator SENSOR 6k, the pipe flow simulator OLGA 6 GUI and Pipe-It to couple the two simulators.

In well start-up operations the only parameter that can be influenced directly is the change in choke position and the corresponding change in THP. This change is introduced in the OLGA model. OLGA then calculates how this change affects the inflow pressure into the tubing at bottom of the well. From this point on the SENSOR model calculates the caused changes in flow rates within the reservoir. The communication of the two models is handled in Pipe-It. The entire simulation is broken down into small time steps (TSs), eg. 15 seconds. For each TS Pipe-It with the help of OLGA and SENSOR optimizes the change in THP. Optimizing means to ensure the largest increase in surface flow-rate for each TS but without violating maximum change in pressure and/or flow rates.

Besides simulating well start-up this paper also shows an efficient way to simulate shut-in conditions within the coupled model. This is important because the initial state of a well before start-up is shut-in.

1.2 Advancements to the current model

The current model is only able to simulate flow scenarios where the entire well stream flows in upwards direction. It is not possible to simulate down-flow of fluids. However during start-up and shut-in it is very likely that some part of the well stream also flows downwards. This e.g. could be the case for a well where three phases, oil and gas and water, are predominant. This scenario can lead to simultaneous upwards (gas and oil) and downwards flow (water). This flow phenomenon can now be simulated with the new model. In order to achieve counter-current flow the current model for SENSOR as well as the current model for OLGA had to be changed. Also the Pipe-It model had to be adjusted.

Moreover allows the new model a much more accurate simulation of shut-in conditions within the integrated Pipe-It model. The current model was not able to capture phase segregation in the tubing, especially the development of a section with free gas at the very top of the tubing.

Due to changes that have been made in the PVT calculations the behaviour of the reservoir simulator and pipe flow simulator at the coupling point are much more consistent. This does mean that the results delivered by SENSOR and OLGA at the coupling point are equal. This was not the case for the current model caused by a mistake in the PVT calculations.

It can be stated that the new model draws a much more accurate picture of real physical behaviour during well start-up and shut-in. Well start-up including counter-current flow as well as shut-in can now be simulated within the integrated model.

2. Employed Software

2.1 OLGA 6 GUI

OLGA is a product of the Norwegian company SPT Group AS. It is claimed to be the market leading pipe flow simulator for engineering the flow of oil, water and gas in wells, pipelines and receiving facilities.

OLGA is used for networks of wells, flow lines and pipelines and process equipment, covering the production system from bottom hole into the production facilities. OLGA comes with a steady state pre-processor included which is intended for calculating initial values to the transient simulations, but which also is useful for traditional steady state parameter variations. However, the transient capabilities of OLGA dramatically increase the range of applicability compared with steady state simulators. [2]

The input system for OLGA consists of four input files.

- general input (*.genkey)
- the flow path geometry (*.geo)
- the fluid PVT data (*.tab)
- restart information (*.rsw)

Due to the fact that OLGA offers a very intuitive GUI, it is quite easy to get used to and start working with it. Its user friendliness is also confirmed by many papers dealing with multiphase flow problems solved in OLGA. The used program release for this paper is the OLGA 6.1.0.0112 of 2009.

2.1.1 Steady-state pre-processor [3]

The steady state pre-processor in OLGA computes a steady state solution for a pipeline or an entire flow network. Steady state pressures, temperatures, mass flows, liquid hold-ups and flow regimes are calculated along pipelines. The steady state pre-processor is primarily intended for generation of initial values for dynamic computations, but may also be used as a standalone steady state tool.

This tool is used in the current project. But with using this pre-processor some inconveniences crop up when trying to bring the integrated model to the next level. But first let’s have a look on the advantages this application offers:

- no user defined initial conditions needed
- get a consistent initial state as a basis for dynamic simulations
- perform screening studies

But there are also negative effects of the pre-processor. Especially in case of the described integrated model this thesis deals with. What problems occur and why they occur with the steady state pre-processor are described in detail in the following chapters. Here is only an overview given of what this application can handle and where its limits are. Tasks the pre-processor can not handle and which would affect this integrated model are:

- counter-current flow (e.g. positive gas flow and negative liquid flow)
- zero flow in the pipeline (shut-in)

[2] (OLGA User Manual 6.1 / Introduction / Background)
[3] (OLGA User Manual 6.1 / Input Description / Steady State Processor / Purpose)

- possible instabilities with pressure controlled inflow nodes or negative sources

If the steady state pre-processor can not be used the initial conditions have to be defined by the user. Solutions are now calculated by the dynamic OLGA. The solution computed by the steady state pre-processor and the solution obtained when simulating with the dynamic solver until a steady state is achieved may not be equal. This is mainly due to the two following reasons:

- For unstable systems (e.g. slugging cases) the solution found by the pre-processor may differ from the average value in the transient solution.
- The steady state pre-processor has some small residual errors that are removed by the transient simulation. In some sensitive cases this can cause a difference in pressure, temperature and hold-up profiles.

The dynamic OLGA usually is more robust than the steady state pre-processor.

When defining the initial conditions manually the steady state pre-processor must be turned off (STEADYSTATE=OFF). Pressure, temperature, gas volume fraction, mass flow, and water volume fraction in the liquid phase for all sections in each flow path must be defined by the user.

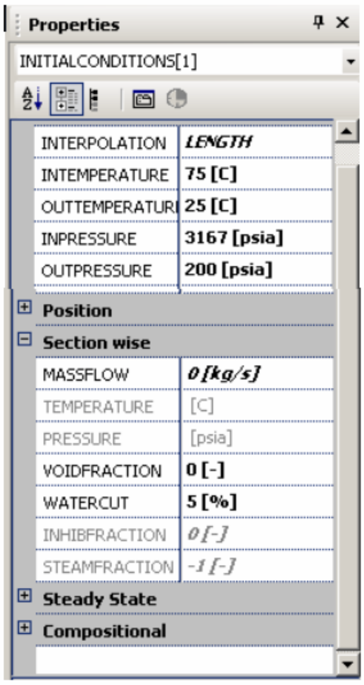


Figure 1: Example for initial conditions in OLGA

2.1.2 PVT File

The properties of the used fluid are specified in a text file in tabular form. OLGA also offers the possibility to define them in a keyword based form, but for this project the tabular form is used. The properties are given as functions of pressure and temperature. According to OLGA support[4] a fluid table should contain between 20-25 pressure points. Using less could lead to instabilities when executing the program. As an example the table for gas density is shown.

Density gas [kg/m³]

p [psia]	0 °C	15,55 °C	25 °C	50 °C	75 °C	100 °C
14,5	1,50	1,44	1,40	1,35	1,30	1,25
369,8	24,05	22,56	21,65	20,13	19,10	18,53
725,2	46,60	43,68	41,90	38,90	36,90	35,80
1000,0	70,53	64,86	61,42	55,71	51,84	49,58
1232,8	90,80	82,81	77,95	69,95	64,50	61,25
1740,5	135,00	121,94	114,00	101,00	92,10	86,70
2000,0	158,26	142,82	133,43	117,62	107,16	100,33
2248,1	180,50	162,77	152,00	133,50	121,55	113,35
2755,7	226,00	203,61	190,00	166,00	151,00	140,00
3000,0	267,00	242,12	227,00	199,50	181,50	168,50
3263,3	281,00	255,27	239,64	210,94	191,92	178,23
3771,0	308,00	280,63	264,00	233,00	212,00	197,00
4000,0	328,75	301,95	285,66	254,20	232,75	218,43
4278,6	308,00	280,63	264,00	233,00	212,00	197,00
4786,2	400,00	375,12	360,00	327,00	304,00	292,00
5000,0	426,74	405,92	393,27	365,11	344,85	333,06
5293,9	463,50	448,26	439,00	417,50	401,00	389,50
5801,5	527,00	521,40	518,00	508,00	498,00	487,00
6309,1	579,00	573,40	570,00	560,50	550,50	540,50
6816,8	631,00	625,40	622,00	613,00	603,00	594,00

Figure 2: Example fluid PVT table

[4] olgasupport@sptgroup.com

2.2 SENSOR 6k

SENSOR stands for "System for Efficient Numerical Simulation of Oil Recovery". It was developed by Dr. Keith H. Coats beginning in 1992 and is now a product of Coats Engineering, Inc.

Sensor is a generalized 3D numerical model used by engineers to optimize oil and gas recovery processes through simulation of compositional and black oil fluid flow in single porosity, dual porosity, and dual permeability petroleum reservoirs. Sensor reservoir modelling software provides unparalleled results in terms of speed, accuracy, stability, reliability, and ease of use. It runs in a fraction of the CPU time required by other reservoir simulators, for both black oil and compositional fluid PVT descriptions, enabling unprecedented levels of detail and accuracy in studies, and allows making much better and faster decisions. The Sensor reservoir simulator has been used in numerous field studies by consultants and by independent, national, and major integrated oil companies. Sensor is compiled for use on hardware running Windows 32- and 64-bit operating systems. A limited version can be downloaded, along with data sets, mapping and plotting tools, and documentation. Integrated interfaces and tools for static modelling, data preparation, job submission, pre/post processing and visualization, assisted history matching and optimization, and complete seismic-to-simulation workflows are offered by third parties.[5]

For this project the black oil PVT model was used exclusively. The black oil PVT includes oil in the gas phase, and therefore applies to gas condensate and volatile black oil problems. The black oil PVT table can be re-entered in Recurrent Data at various times to reflect changes in surface processing at the time of the re-entry. Black oil tables can be internally generated from input compositional descriptions.

Sensor includes Impes and Implicit formulations. Its three linear solvers are reduced bandwidth direct (D4), Orthomin preconditioned by Nested Factorization, and Orthomin preconditioned by ILU with red-black and residual constraint options.

Any grid type or combination of grid types can be used with Sensor.

- Seven point orthogonal Cartesian xyz grid
- R-theta-z cylindrical coordinate system
- Any grid, e.g. corner point, refined, unstructured, hybrid

Sensor needs getting used to in the beginning. The not existing graphical user interface (GUI) makes it a little bit difficult to understand how it works and how the input has to look like. But once used to that, Sensor is very easy to handle and changes to your input data can be done very quickly. The input data is written in text files (e.g. in TextPad [6]). To execute Sensor a batch file is needed. Sensor runs in the DOS command window and an output file is generated, also in the format of a text file.

For a graphical depiction of the results Coats Engineering offers Map2Excel and Plot2Excel. Those tools are offered for free. They have been used to investigate the results created with Sensor.

2.2.1 Map2Excel [7]

Map2Excel is a simple tool to create two-dimensional contour maps on aerial or vertical slices of a structured reservoir grid. Maps can display a variety of grid properties of Sensor input or calculated results. The map display includes a slice of the simulation grid with properties in colour-fill mode, wells, well names, faults, colour legend, IJK subscript, depth scale and title. User defined map units are available. SensorMap can also calculate and show properties

[5] (Coats Engineering Inc., 2009; http://www.coatsengineering.com/sensor_reservoir_simulator.htm)
[6] (Helios Software Solutions)
[7] (Coats Engineering Inc., 2009; http://www.coatsengineering.com/plot2excel.htm)

averaged across several blocks normal to the slice. Certain options are provided to control display of the main map details.

Map2Excel is actually an Excel spreadsheet with Visual Basic automation. It reads map data files created by SensorMap and creates drawings in MS Excel using the MS Draw functionality. With MS Draw you can also manually change the maps to edit title, delete or add new elements, etc.

SensorMap is a Fortran program that reads user input and the Sensor binary map file (fort.71). For each of the requested maps the program assembles xyz coordinates, grid block properties, fault and well positions, plus the display options. The output data are written in a tab-delimited text file with an easy to read format, for input to Map2Excel. This output file is referred to as the *map data file*.

2.2.2 Plot2Excel [8]

Plot2Excel is an Excel spreadsheet enhanced with Visual Basic for Applications to facilitate creating X-Y plots from large formatted data sets. It efficiently generates many plots with a user-controlled standard style. It reads data sets written by SensorPlot with a specified format in Tab-delimited text files. The program allows the user to select data tables using an advanced search filter. Afterwards, any data columns from the selected tables can be plotted on the same chart with user-controlled size, style and axis scale. Plot2Excel creates descriptions of each plot and allows saving them as a text file called *Plot-Log*. Later you can import a new data set plus the saved Plot-Log file. By pressing a button, all plots are created with the new data, using the same format which was previously defined. More than 100 plots can be instantly generated with the Plot-Log function.

SensorPlot is a post-processor program, which converts Sensor result files for graphical analysis with X-Y plotting application Plot2Excel. You can select data for plotting, compare multiple simulation runs, add observed data, make unit conversions and display results in MS Excel. A Sensor reservoir simulation run creates a binary plot file “fort.61” containing calculated data for wells, platforms, regions, superregions, total field and traced components.

[8] (Coats Engineering Inc., 2009; http://www.coatsengineering.com/map2excel.htm)

2.3 Pipe-It 1.0

Pipe-It is a product of the Norwegian company Petrostreamz AS. It is used to couple the reservoir simulator Sensor 6k and the pipe flow simulator OLGA.

This software has been specially designed for integrated asset optimization and exploitation in the petroleum industry. Its GUI enables the user to lay out his project in different organizational layers, which makes it easy to understand and organize complex projects integrating many different processes and sub-processes. The objective of such an effort is to optimize the operational mode of the considered production system in order to maximize the economic value generated, while integrating any number of processes and elements that are connected to it and may affect its performance. [9]

A Pipe-It project is basically designed by using four different elements.

- composites (brown rectangle)
- resources (blue rectangle)
- processes (green ellipse)
- connectors (black arrows)

Composites are used to make a project easier to understand. They group resources, processes, and sub-composites. For example this project uses a composite shut-in that contains the entire process of simulating shut-in conditions of the entire system, using SENSOR and OLGA. Resources can be any type of input, output or DOS-batch file. Those files are required by the processes to work properly. Processes tell Pipe-It what to do, e.g. execute Sensor with the given input file. Due to the way the processes and composites are arranged to each other, a parallel or serial execution can be achieved. In simple words, the design of the project tells Pipe-It the chronology of the defined processes. Connectors are used to connect the different resources, processes and composites.

A graphical depiction of a Pipe-It project is shown to make the system of organizational layers easier understandable.

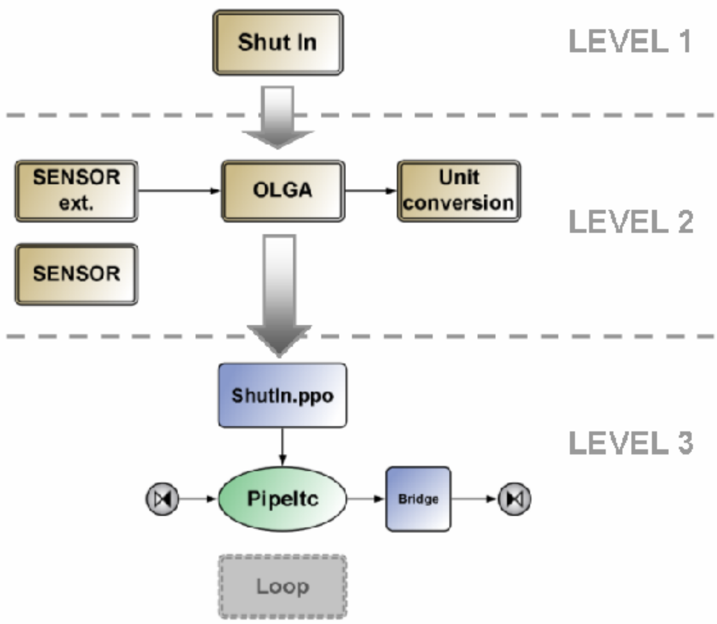


Figure 3: Graphical depiction of a Pipe-It project

[9] (Petrostreamz AS, 2008)

2.3.1 PipeIt Optimizer

Pipe-It is designed to perform optimization globally or locally at any process being part of a PipeIt project.

The optimizer works with four different variables.

- VAR – Variable. User or optimizer specified. May be written to file. Updated before any other variable and before model execution.
- AUX – Auxiliary. Either set by equation, read from file, or user-specified. If set by equation it may be written to file. Updated after VARs but before model execution.
- CON – Constraint. Unless a user specified constant, either read from file or set by equation after model execution.
- OBJ – Objective. Same as CON, except the optimizer will try to minimize or maximize it.

Modus operandi of the optimizer:

At first all the defined VARs are updated. They are either user or optimizer specified but never set by equation. If the VAR is linked to a file it will be written to that file during the update process. The next step is that all the AUXs are updated. AUXs can be specified by equation, a file, both or neither. If it is set by equation it will always use this value. If also a file is specified, the updated value will be written to that file. If it has a file but no equation the value will be read from that file. It also can be user specified when neither file nor equation is used.

After the VARs and AUXs are updated and written (if necessary) to their associated files the model is executed.

Now the other variables, CONs and OBJs, are updated. CONS can be specified by equation or file, but not by both. A user specified constraint would have neither.

At the end the calculated values for the CONs fed back to the optimizer. If all constraints are fulfilled the optimizer will stop. If any constraint is violated, the optimizer will execute another run with a new value for VAR.

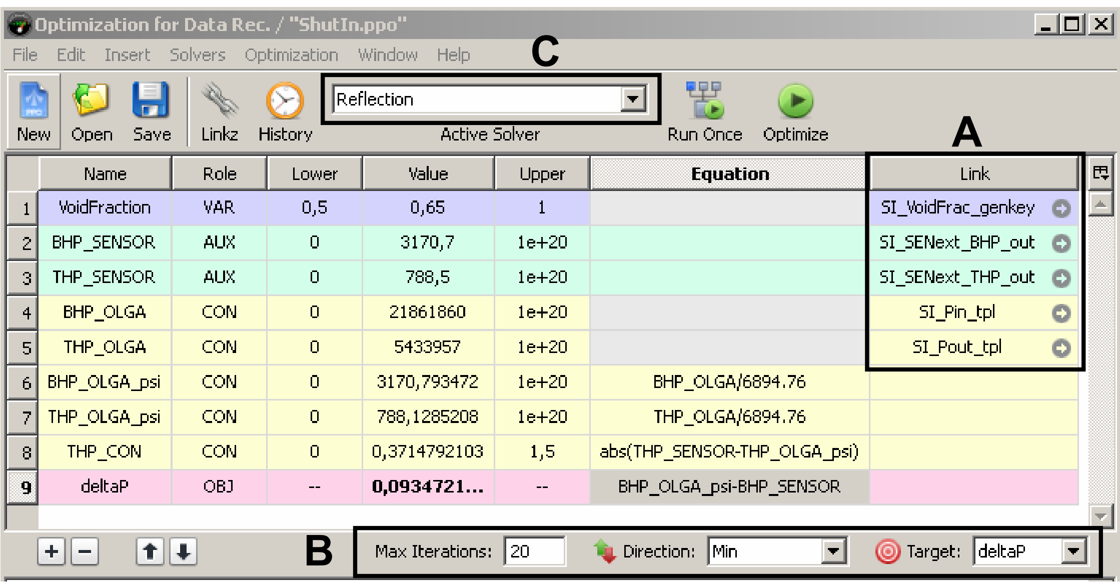


Figure 4: Example optimizer PipeIt

Above an example for a defined optimizer is shown. The different variables are quite easy to diversify because they are shown in different colours and moreover they are named under role. The interesting parts are highlighted with black rectangles. “A” shows the association of certain variables with different files via Links. VAR will be actually written to that file but AUXs and CONs are read. “B” shows the settings necessary in combination with a used objective. The number of maximum iterations in that example is set to 20. If the optimizer would not find a solution that fulfils all given constraints within 20 iterations, it would take the best fitting value and the project would proceed with that one. The defined target for the optimizer in this example is a pressure difference that should be minimized. Under “C” the optimization method can be selected. It can be selected between in-house solvers as well as 3rd party solvers from Frontline Systems Inc. In this project the Nelder-Mead-Simplex reflection solver is used. [10]

PipeItc

PipeItc.exe is a build in application that launches the optimizer automatically. The optimizer can be started from the optimizer window manually, but this is very inconvenient when multiple optimizers within a project are used. With the PipeItc application only the optimizer input file and the composite path within the optimization should run needs to be specified. Whenever this optimization is finished, the sub process is terminated and PipeIt proceeds with the next task in line.

2.3.2 Linkz and LinkzUtil

Linkz and LinkzUtil are build-in applications to locate certain values in data files and transfer them between files.

Linkz is responsible for finding values and defining their position within a file. To do that the position of the target value is cross-referenced with different strings in the data file. In combination with the order whether Linkz should start searching at top or bottom of the file the target value can be defined uniquely. An example how a certain value within a file is cross-referenced is shown below. Those defined links can be addressed within PipeIt, either by the optimizer or LinkzUtil.

LinkzUtil is an application that is run from a DOS-batch file. It actually transfers variables between different files. All variables that are not transferred within an optimizer are moved by LinkzUtil. Only the information on what variable to read and what value to overwrite is needed.

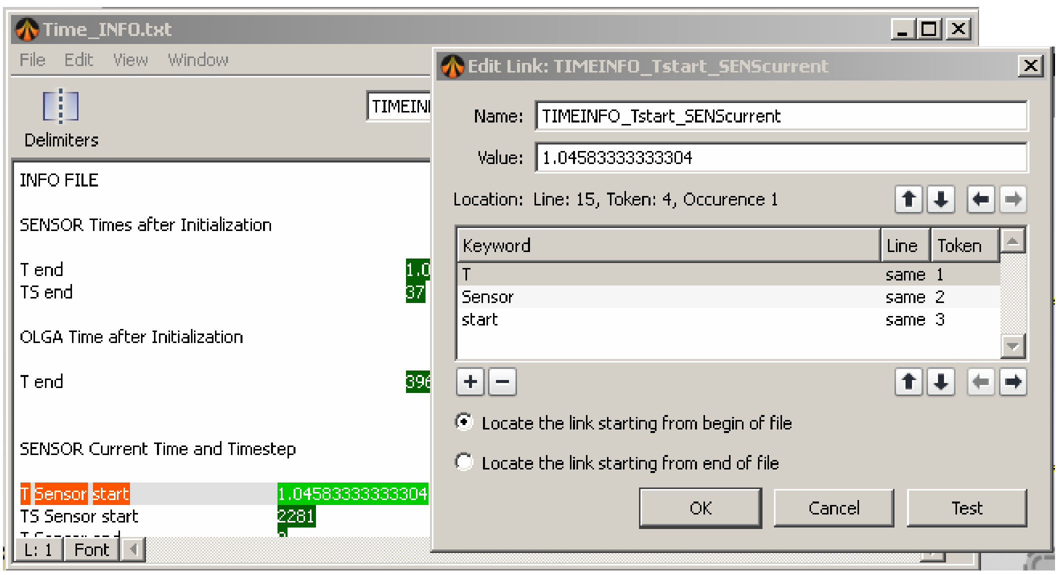


Figure 5: Example for PipeIt Linkz

[10] see Michael Schietz, Master Thesis “Optimization of well start-up”, p. 15-16

3. Initialization of the system

Initialization of the model has to be done because the initial conditions need to be known. In the case of simulating a well start-up the initial condition is shut-in. This imposes that before the simulation of well start-up can be conducted, the shut-in condition needs to be simulated in both models, OLGA and SENSOR. The current model uses SENSOR to simulate the reservoir up to the coupling point. OLGA takes care of the fluid behaviour in the tubing up to the surface, during shut-in as well as flowing conditions. However this system has been changed in order to get results closer to the real physical behaviour during shut-in. SENSOR is not only used to simulate the reservoir part in the model, it is extended to cover also the tubing section up to the surface. This entire model from the reservoir to surface does not replace the model that only covers the reservoir, it runs additionally. This does mean that now two SENSOR models and one OLGA model are used to initialize the system.

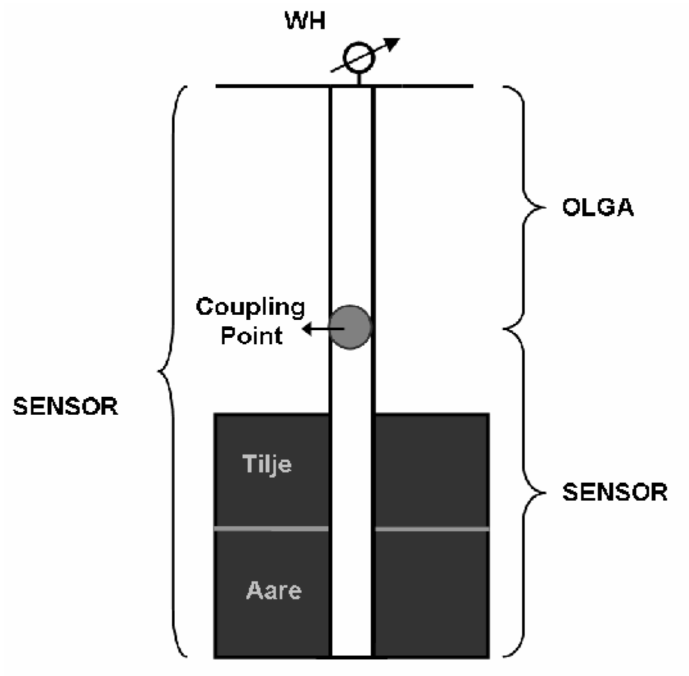


Figure 6: Different models for initialization

On the first sight it may seem that the simulation time will increase due to the additional model, actually it stays quite the same. Consequently there is an increase in the model accuracy without any negative effects on the simulation speed.

To make it better understandable a short review of the current modus operandi for the initialization shall be given. SENSOR is used to define the shut-in condition for the reservoir part up to the coupling point like shown in the figure above. Accordingly the shut-in bottom-hole pressure (BHP) is calculated by SENSOR. This BHP is than transferred to OLGA. OLGA with the help of PipeIt calculates the corresponding tubing head pressure (THP) for shut-in conditions. After the THP is found, the initialization is finished.

Another reason why initialization has to be done is caused by the way the integrated model works. The entire integrated model run is split-up in small time steps, e.g.15 seconds. This can be done because OLGA as well as SENSOR creating restart-files after every run. This restart files contain information about the current conditions in the system at the end of each simulated time step. These restart files can be read from the programs at the beginning of the next time step. Due to that is it possible to restart the model with the same conditions predominant at the end of the previous time step. This leads finally to a continuos simulation over the entire start-up, split up into many small time steps.

3.1 OLGA shut-in

In the current integrated model the initialization in OLGA is done with a mass-flow controlled node at bottom and a pressure controlled node at top. Caused by the fact that the model uses the steady-state pre-processor the initialization has to be started with a short period where fluid flows through the tubing to the surface, although the system is initialized for shut in conditions. This period is only one second. Subsequently the flow rate at the inflow node is set to zero. A PipeIt loop is constructed to find the corresponding shut in tubing-head pressure in OLGA that leads to the same pressure at the coupling point given by SENSOR. A graphical depiction of this modus operandi is shown in figure 7.

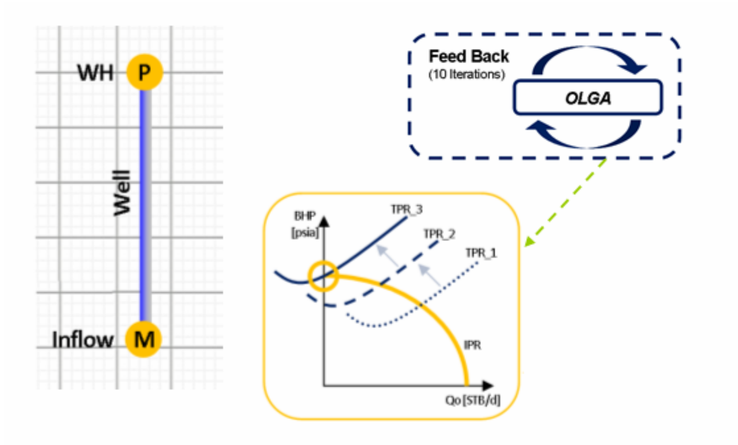


Figure 7: Current Initialization

By using a pressure node at top the system is not really shut-in with a boundary that doesn't allow any flow at all. The final condition in this case is a tubing filled totally with oil up to top. However this does not display the reality. The real shut-in condition would be that the tubing is filled with oil and in the top section would be some free gas, although an all oil reservoir is present. This free gas can come from predominant free gas in the tubing or somewhere in the tubing the pressure drops below the bubble point and gas will start getting out of solution. In the current model gas that accumulates in the top section of the tubing seeps through the well-head (WH) node. This effect caused by the pressure boundary was neglected, but in terms to get a more accurate simulation of shut-in conditions the model has to be changed. The chart below shows the very little flow rates through the top node during shut-in simulation.

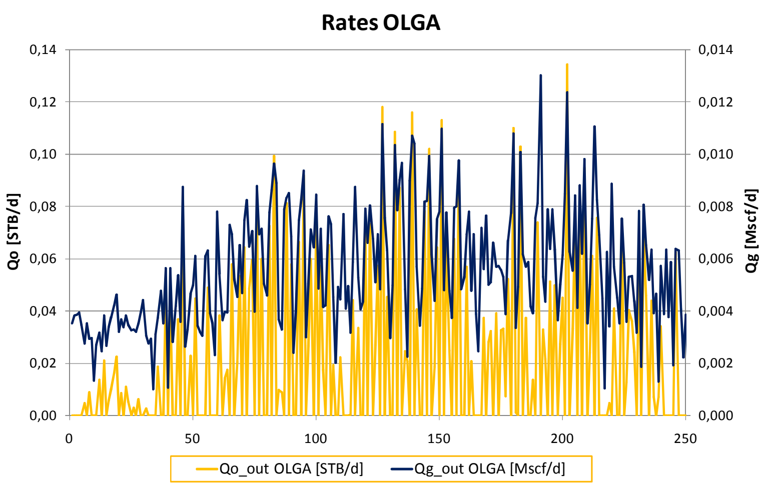


Figure 8: Flow rates through top node while shut in

This means that the tubing-head (TH) shut-in pressure in the current model is actually too low because it does not account for the gas at top and the corresponding change in the pressure gradient from oil to gas.

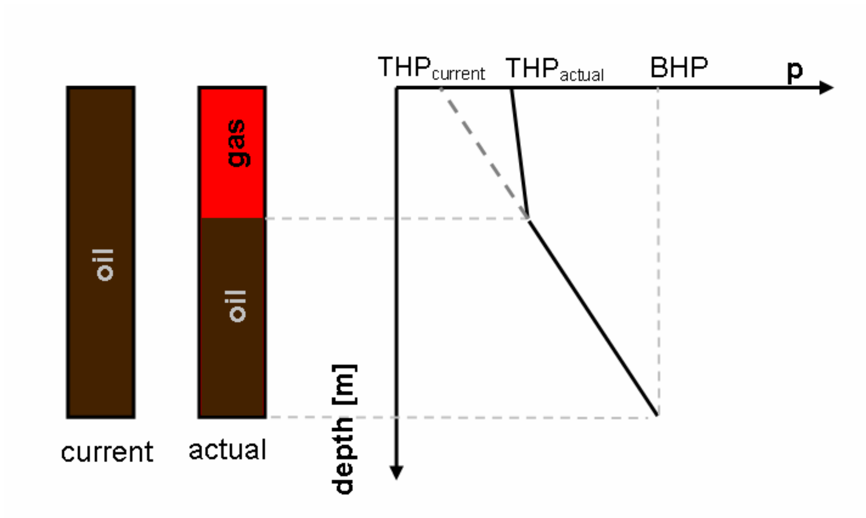


Figure 9: Change in shut-in THP

To avoid this error in the WH shut-in pressure the model in OLGA has to be changed. The changed model in OLGA now consists of a pressure controlled node at bottom and a closed node at the WH. The application of a pressure controlled node at bottom is possible for the simulation of shut-in conditions because no inflow into the system exists. It has even the advantage that negative flow rates at the bottom can be detected. These negative flow rates of oil occur caused by gravitational segregation displacing the oil in the upper section of the tubing. The latter could be used to re-evaluate the BHP in SENSOR by injecting that amount of fluid that is leaving the system from OLGA into the reservoir. However, this is not necessary because the used SENSOR model simulates the entire system up to the WH. This means that SENSOR already takes care of the fluid that flows from the tubing back into the formation and comes up with the correct bottom-hole and tubing-head pressure needed for OLGA.

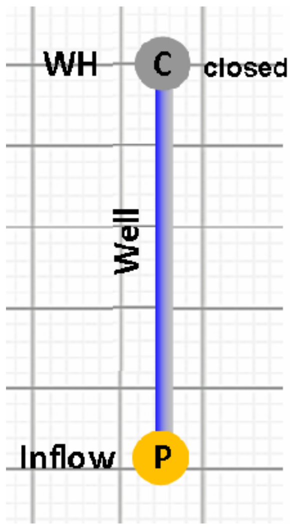


Figure 10: Schematic depiction of the new OLGA model

Now all necessary input parameters for the OLGA model are know. As initial conditions for the OLGA model the known pressures at bottom and top are used. The initial void fraction (free gas in the tubing) has to be determined by trial and error to match the well-head pressure at the end of the simulation to the initially given. This is done in PipeIt with the described initialization loop in chapter 3.4. Immediately after the well gets shut-in the tubing is filled with oil and with a certain amount of free gas distributed over the entire tubing length. Then gravitational segregation begins. This means that all the free gas in the tubing, and additional gas that gets out of solution when the pressure drops below the bubble point pressure, will start to rise and displace the oil in the most upper section of the tubing. This implies that as much fluid as necessary to keep the given BHP will leave the system through the bottom node. To ensure that the OLGA model works properly several tests were conducted. The following chart shows the negative flow rates through the bottom-node for a shut in period of 20 hours. However the chart only shows the first hour during shut in. After this time the majority of fluid that flows back has already left the system and the flow rate approaches zero.

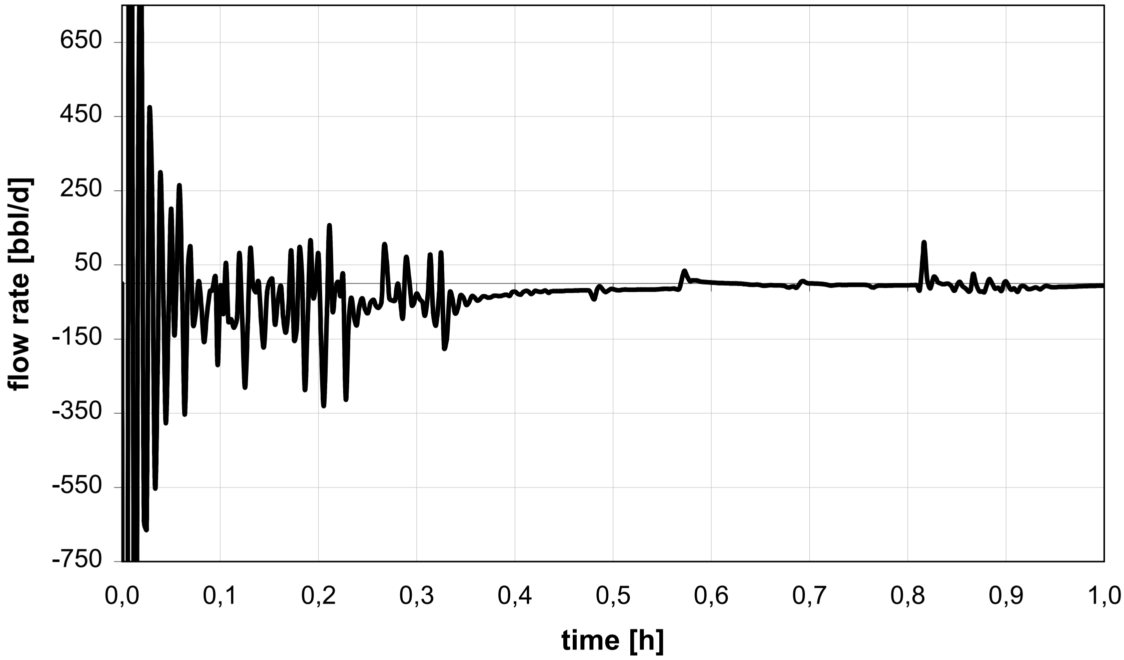


Figure 11: Negative flow rates during shut-in

Due to numerical disturbances the calculated flow rates show a quite erratic behaviour. But it can be seen that most of the flow-rate values are negative. To show that the major part of the flow is negative also the accumulated oil volume flow through the bottom node is shown.

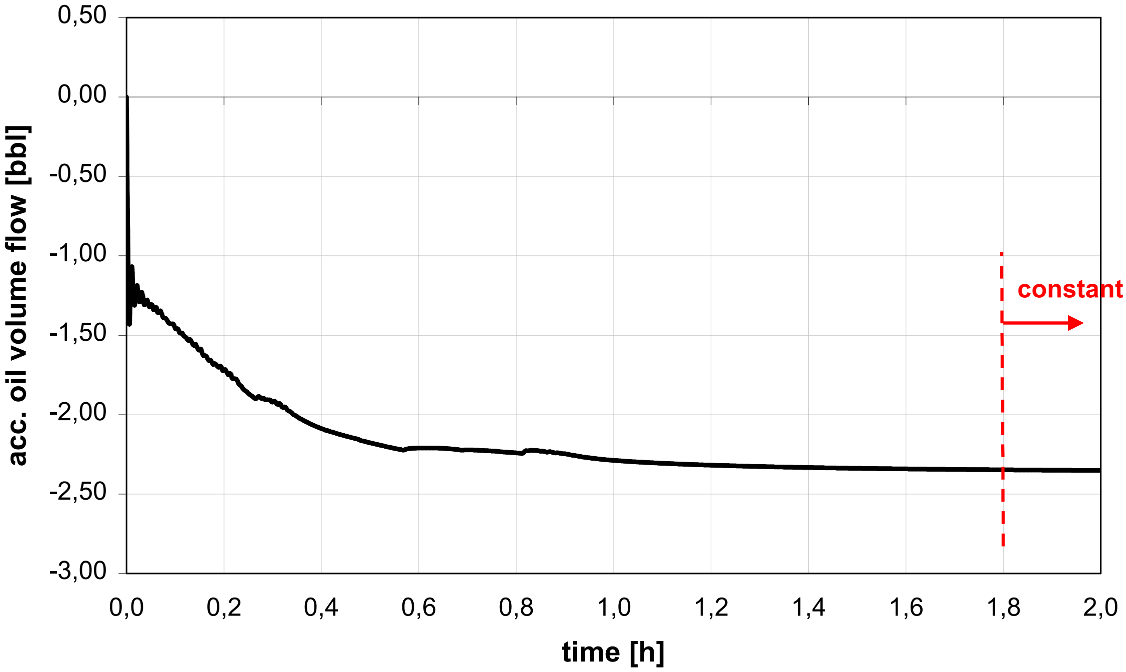


Figure 12: Accumulated oil volume flow during shut-in

This chart shows that approximately 2.3 bbl of oil leave the system during the simulated shut-in period of 20 hours, although only 2 hours are shown in the chart. After these two hours the value for the accumulated oil volume flow stays constant.

The same investigation was examined with a water fraction of 5 % in the tubing at the beginning of the shut-in period. Normally it should not cause any troubles when including a certain water fraction in the tubing. However this investigation has been conducted because OLGA had some difficulties to handle different water problems associated with flow scenarios (cf. chapter 4.1.2). To make sure that the system also works with water a test case was set up with an initial water fraction of 5%.

What should happen is that the entire water leaves the system during the shut-in period and the system ends up with 0% water fraction, some gas at the top and the rest filled with oil. The results of the run are shown in the following. Due to a highly erratic behaviour of the oil and water flow rates figure 13 only shows accumulated oil and water volume flows.

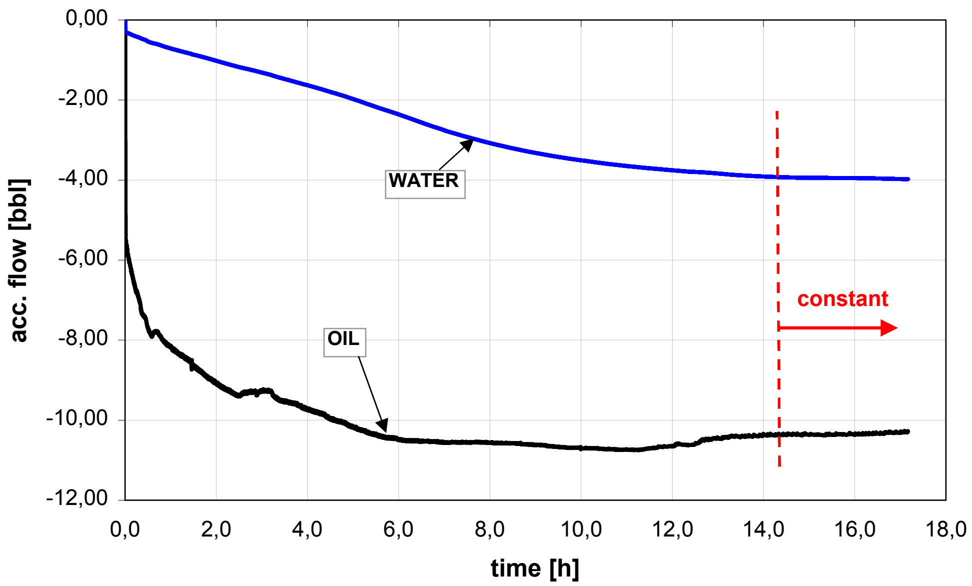


Figure 13: Negative flow rates during shut-in including water

It can be seen that both oil and water leave the system. Compared to the system with zero water-fraction it takes much longer that the system equilibrates, 14 hours compared to 2 hours. To ensure that this amount of water is the entire water that was initially in the system oil, gas and water fractions at the end of the shut-in period are shown. This behaviour of leaving water is now possible because a pressure driven node at the bottom is used. If a mass-flow controlled node would be used, the water would not leave the system. Instead it would accumulate at the bottom of the tubing. This would be the coupling point in the integrated model. Another reason that this happens is the usage of the steady-state pre-processor (SSPP) in the current model. The SSPP generates initial conditions automatically. The downside is that it does not account for counter-current flow, and accordingly OLGA does not account for negative flow rates. The behaviour of the current model would lead to the highly unrealistic end condition shown in figure 14. It is only a product of the integrated model and that it does not account for down flowing water.

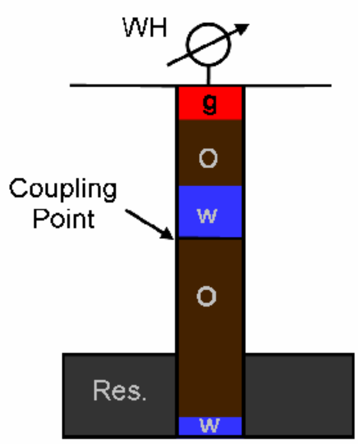


Figure 14: End condition for the current model after shut-in

The changed model is now able to solve this problem. In the chart below full lines show the fractions at the end of shut-in period, dashed lines at the beginning. The free-gas fraction is not shown at t=0 because it is set to zero.

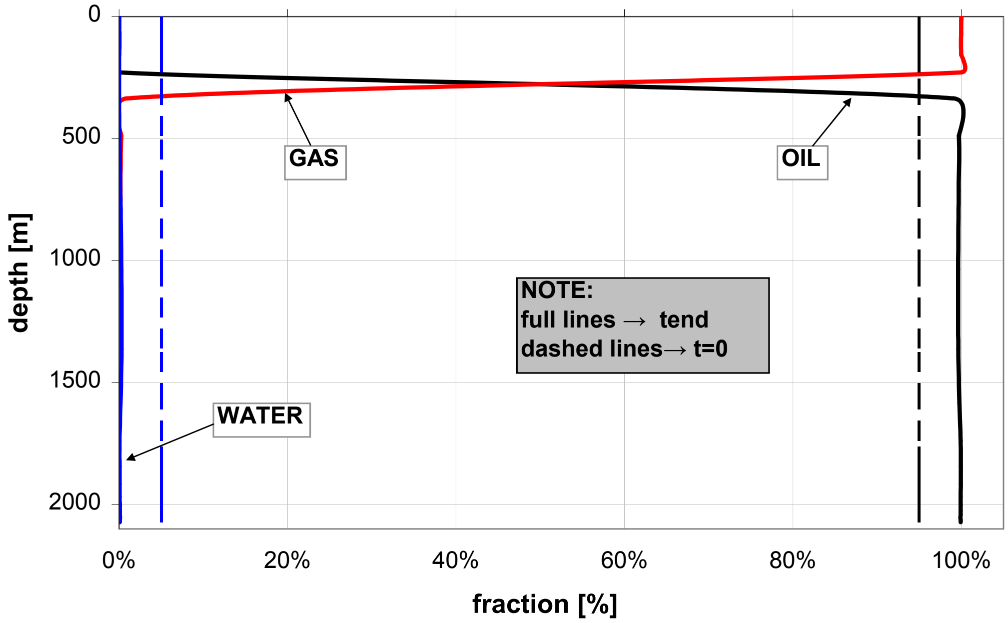


Figure 15: Oil, gas & water fractions for shut-in

All the above shown investigations of the model are necessary to prove that the model works properly in different situations.

It is now possible to simulate shut-in conditions very close to the real physical behaviour of a well. For the latter is it important that during the shut-in period fluids can leave the system at bottom. Due to the changed model and usage of the dynamic solver of OLGA instead of the SSPP this is now possible.

The input file for the OLGA model used for shut-in is shown in the appendix.

3.2 SENSOR shut-in

Like already mentioned in the beginning of this chapter two SENSOR models are in use for the initialization of the integrated model. Compared to the current integrated model, that only uses one SENSOR model, it seems quite obvious that the run-time for two models will be twice as much as the run time for one model. In fact this is true but the difference in total run-time for the integrated model stays insignificant because it only takes a few seconds to run a model in SENSOR and the models for shut-in only run once. A schematic depiction of the now used models is shown in figure 16.

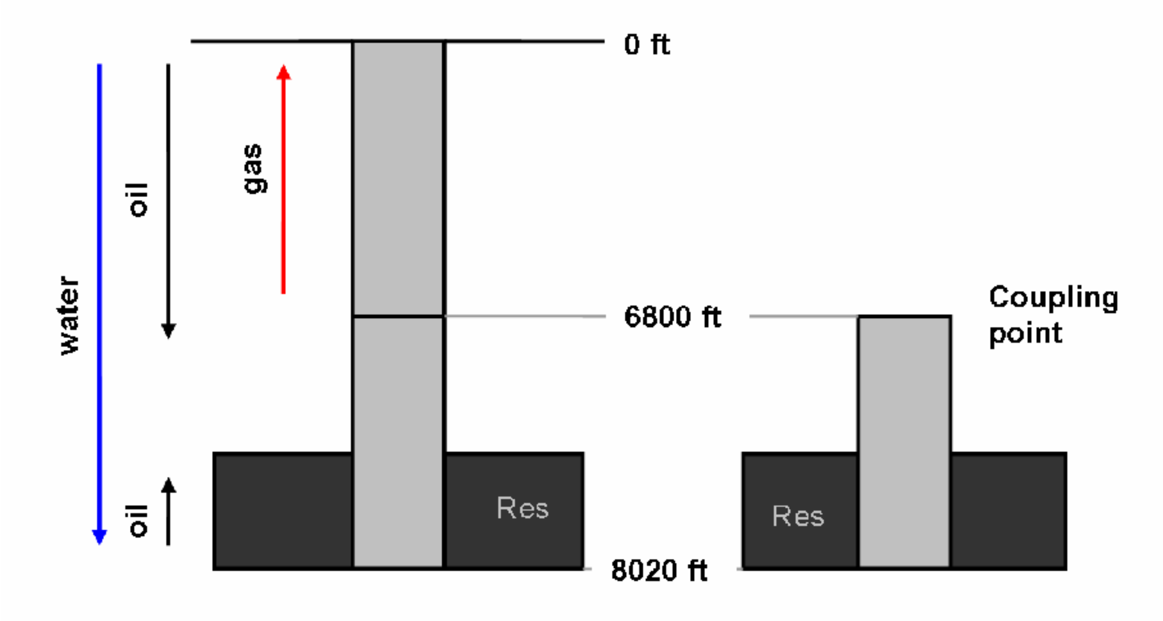


Figure 16: Schematic depiction of SENSOR models for initialization

But what is the reason that two models are used now? Like already very short mentioned in chapter 3.1 it is necessary to capture all flow phenomena to get an accurate BHP. If the SENSOR model would only reach up to the coupling point it would have no idea what is going on above that point. But it is important that the tubing part is also taken into consideration during shut-in of the system.

The most likely phenomenon that will occur is that oil and/or water from the tubing flows back into the reservoir part of the model. Simultaneously gas will rise in the tubing. This gas can be free gas or gas that gets out of solution. Down-flow has been already discussed extensively in the OLGA section of shut-in. A SENSOR model only up to the coupling point does not consider down-flow from the tubing. It would be necessary to take the negative flow rates from the OLGA model and inject those into the reservoir to see how it affects the BHP. This imposes that the entire initialization process has to be split up into small time-steps. This is no problem to do. But the run time for the model would literally explode. If a shut-in of 10 hours is simulated and the model is broken down into 15 second time-steps 2400 TSs would be necessary to achieve a result. This huge amount of more effort stands in no relation to the increase in accuracy the integrated model would experience.

Using instead a SENSOR model up to the surface during initialization has the advantage that all flow phenomena during shut-in are captured and SENSOR itself takes care of probable down flowing fluids. By running this model, which only takes a couple of seconds, highly accurate results combined with very short run time of the model can be achieved. Although this little amount of fluid that flows back probably will have no or only a very small impact on the BHP it has to be captured in terms to get an accurate model.

Now the tubing up to the surface is grided and simulated in SENSOR too. Like the small part above the reservoir to the coupling point that is already grided in SENSOR. For more information about the existing reservoir model in SENSOR please refer to the Master thesis “Optimization of well start-up”. [11] In terms to keep it simple the tubing up to the surface has been grided in majority in 50 ft long sections.

SENSOR is a reservoir simulator and not constructed to simulate flow through pipes like it would be needed when extending the SENSOR model to the surface. But the flow behaviour in the tubing section from SENSOR can be matched to the flow behaviour in OLGA. In SENSOR it is not possible to define tubing or any other pipes. The only chance is to imitate the properties of tubing by defining the region that represents the tubing in a proper way. This means that the porosity is set to one (Φ=1) and the permeability has to be defined accordingly. By defining the permeability it is important that the overall dynamic pressure drop in the tubing for a certain flow rate matches the overall dynamic pressure drop for the same flow rate in OLGA. If this requirement is fulfilled the SENSOR model up to the surface can be used with clean conscience for the initialization process. Especially because the flow rates during the shut-in period usually are very small and consequently they will only lead to small dynamic pressure losses.

The problem that has to be solved here is that SENSOR uses Darcy’s law to calculate the pressure drop caused by flowing fluids. The nature of Darcy’s law leads to a linear behaviour of the pressure drop in relation to the flow rate. Darcy’s law is a simple proportional relationship between a constant flow through porous media, the viscosity (μ) of the fluid, the permeability (k) and the pressure drop over a given distance. In its most simple form it is shown in figure 17.

$$Q = -\frac{kA}{\mu}\frac{(Pb - Pa)}{l} \quad \text{........................Eq. 3.1}$$

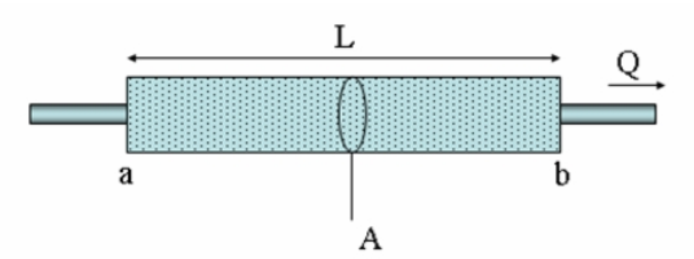


Figure 17: Darcy’s Law

It is obvious that Darcy’s law describes a linear relationship between permeability and flow rate. However OLGA uses more complex relationships to calculate the dynamic pressure loss caused by fluid flowing through the tubing. The exact way how OLGA calculates flow-rates and pressures is a secret of SPT Group. Like already mentioned permeability in SENSOR is the only parameter that can be influenced to end up with a similar behaviour like OLGA. All the other parameters, cross-section of the pipe, viscosity of the fluid and length of the pipe, are constants.

In order to imitate the behaviour of OLGA, several runs of SENSOR with different permeability values over a flow-rate range of 0 – 5000 STB/d were conducted. For comparison also OLGA has been run over the same flow-rate range. The next chart shows the dynamic pressure losses for SENSOR and OLGA. Only the dynamic pressure losses are of importance because the hydrostatic pressure loss in the tubing should be the same in both programs.

[11] Michael Schietz, Master Thesis “Optimization of well start-up”, 2009

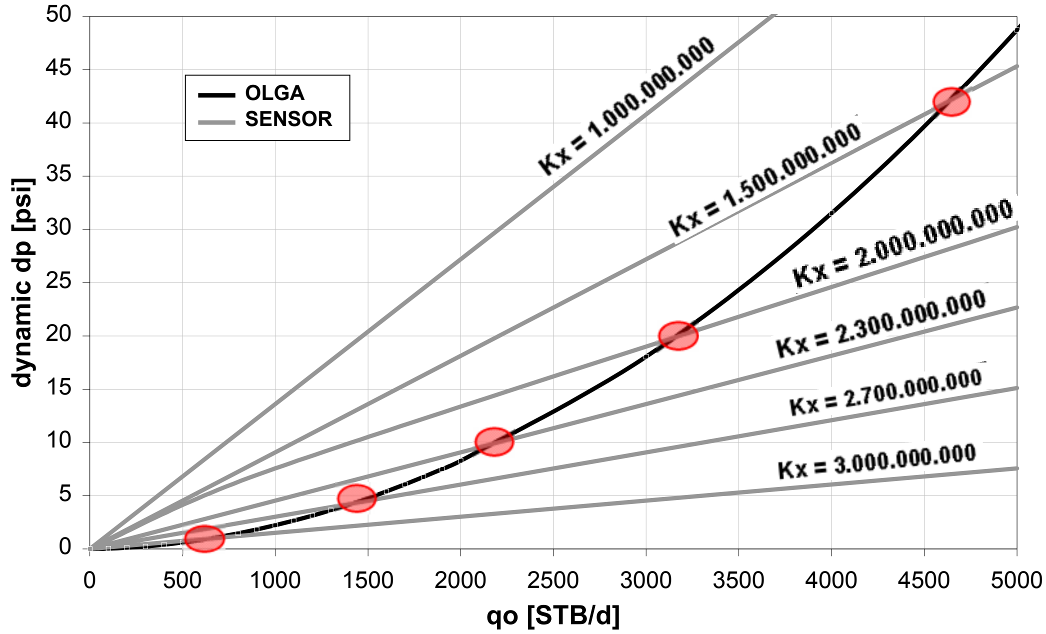


Figure 18: Dynamic pressure loss OLGA / SENSOR

As expected shows SENSOR a linear behaviour over increasing flow rates, OLGA doesn't. The red circles mark intersections of SENSOR and OLGA. At these points SENSOR shows, with the given combination of flow-rate and permeability, the same behaviour like OLGA. Transferring this intersection points to a chart showing permeability over flow-rate leads to an equation which can be used to calculate k-values for all flow-rates.

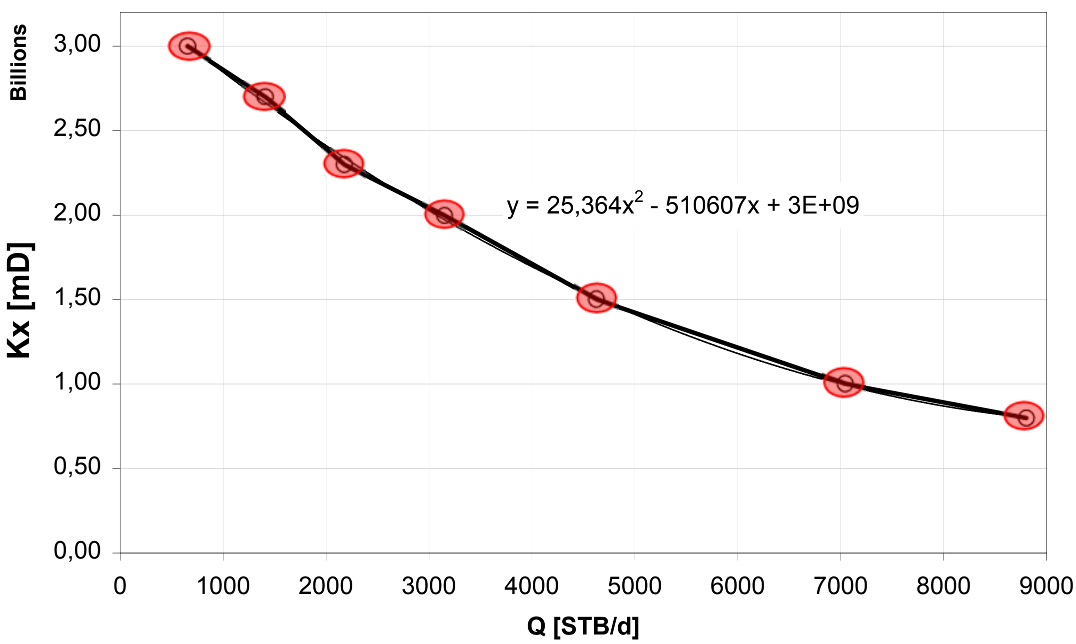


Figure 19: k – fitting SENSOR

Most of the k-values are in the range of billions. Especially for smaller flow-rates the k-values are very high. By using this equation for every time-step in the coupled model the corresponding permeability in SENSOR could be updated. But due to the fact that SENSOR gets problems if the permeability is set to a value higher than $2*10^9$ mD, the final decision is to set the permeability value a little below that limit. It is set to $1.90*10^9$ mD.

3.3 Comparison OLGA – SENSOR shut-in

In theory the tubing head pressure calculated by OLGA and the tubing head pressure calculated by the overall SENSOR model should be the same, although SENSOR and OLGA are using different mathematical models. But for shut-in condition, where the pressure difference between WH and bottom-hole (BH) is only driven by hydrostatics, this shouldn't play any role. The only parameters that have an impact on the pressure distribution in the tubing are the mixture of fluids currently predominant in the tubing and the corresponding pressure gradients and of course the depth. This leads to the PVT calculations done for SENSOR. This comparison also has been done to control whether the calculations have been done properly or not. In OLGA the possibility is given to use different thermal calculation models. Depending on what model is used the WH pressure in OLGA differs. The different models available for thermal calculations are:

- OFF: No temperature calculations at all, the initial temperature has to be specified.
- WALL: Heat flux through the pipe wall layers is calculated by the code with user-defined thermal conductivities, specific heat capacities and densities for each wall layer.
- FASTWALL: Similar as wall but heat storage is neglected in the well.
- ADIABATIC: No heat transfer to the surroundings [12]

All four methods of OLGA are taken into consideration and compared to the result that is delivered by SENSOR.

Figure 20 shows the pressure profile for all different methods in OLGA as well as the SENSOR profile.

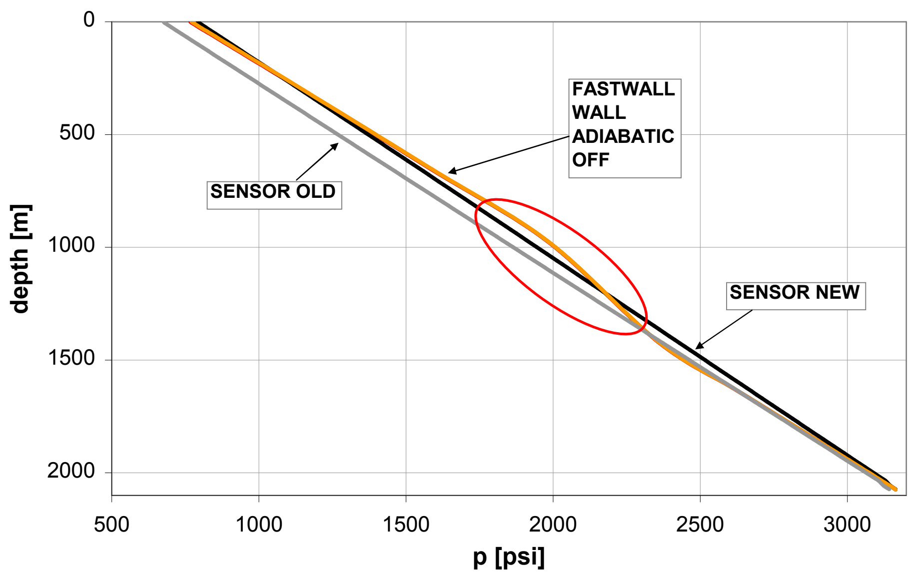


Figure 20: Comparison OLGA to SENSOR total depth

The difference in behaviour between SENSOR and OLGA pressure profiles is as assumed not very significant. Only at approximately half the depth, highlighted with the red ellipse, OLGA shows a little bit a curvy shape, the rest is quite linear. This is most likely caused by the coarser grid that is used in the OLGA model in this section. By showing the total length of the tubing no difference between each thermal calculation method in OLGA can be seen.

At the beginning of this study the difference in the tubing-head pressure between OLGA and SENSOR was quite significant, even in a not feasible range. Shown in the graph as "SENSOR

[12] (OLGA User Manual 6.1 / Input Description (Thermal Computations)

Old". This large difference was caused by a minor mistake in the PVT calculations, which are necessary because OLGA and SENSOR use different PVT parameters to define a fluid. The error itself was that a standard gas density of 0.44 kg/m³ was used instead of 1.44 kg/m³. After eliminating this error the results became much closer. The PVT calculations are not explained in detail anymore; please refer to "Optimization of well start-up". [13] But calculations and results used for SENSOR are shown in the appendix.

Having a closer look on the top section of the tubing reveals differences between the thermal computational models in OLGA, and also the difference to the result from SENSOR gets more obvious.

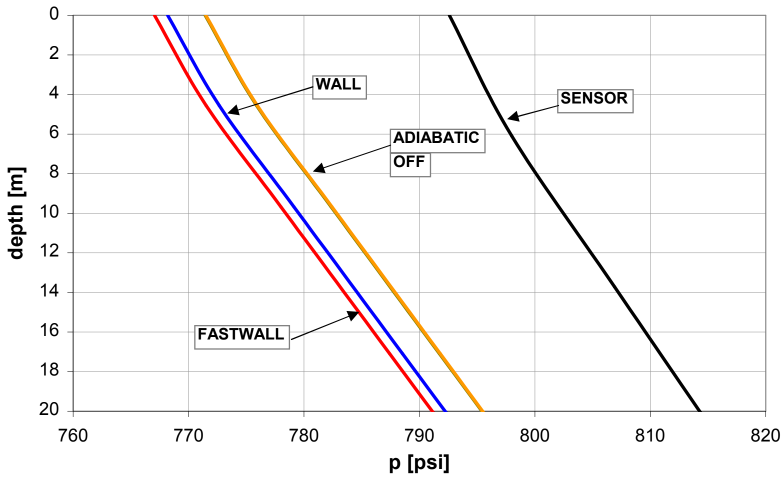


Figure 21: Comparison OLGA to SENSOR top section

The overall behaviour of the four OLGA models is quite the same as in SENSOR. But they get shifted to the left in the top section. In general every OLGA model ends up with a lower WH pressure than the SENSOR model. The closest match to the SENSOR model is delivered by using "Adiabatic" or "Off" as thermal computation method. Comparing the pressure delivered by SENSOR with the pressures delivered by OLGA will give a clue about what model should be used for the simulations. Figure 22 shows the differences in psi and in percent.

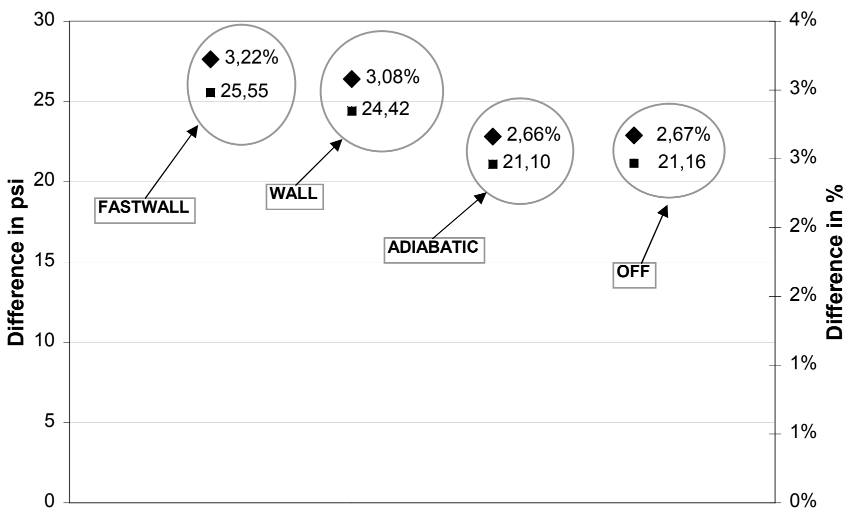


Figure 22: Difference between OLGA and SENSOR model

[13] Michael Schietz, Master Thesis "Optimization of well start-up"

With “Adiabatic” and “Off” the difference is below 3%. This not significant deviation from the SENSOR result can be found in the necessity of PVT calculations. Because different parameters are used also different interpolations within the programs itself need to be done. For the other two methods the difference is slightly above 3 %.

Because the difference between the thermal computational models is very small, each of the methods can be used for the integrated model. Based on the delivered accuracy and the corresponding run-time the decision was made to use for this simulation “Fastwall”. Both accuracy as well as run-time are in a fair enough range for this simulation.

3.4 PipeIt shut-in

The used PipeIt assembly for the initialization is quite a simple one. Basically it contains the execution of SENSOR and a trial and error loop to find the proper initial void fraction for OLGA. At first the SENSOR model up to surface (in Pipe-It named SENSOR extended) runs. After that Pipe-It executes the OLGA model several times with different void fractions to match the THP in OLGA to that one given from SENSOR. The composite "Unitconversion" actually is needed because of an inconvenience in Pipe-It. It is not possible in LinkzUtil to transfer values and simultaneously do for example a unit conversion. Due to that it is a necessity to run a PipeItc application to do the unit conversion and write the values into an info file. A schematic depiction is shown on the right. Let's have a closer look on this Pipe-It model and its assembling. The first part shown in detail is the run of the extended SENSOR model. After that run an accurate description of the shut-in conditions exists. The BHP and the THP are known. These pressures are transferred than to OLGA with the help of LinkzUtil. To show it more figuratively the schematic of the composite SENSOR extended is shown in figure 24.

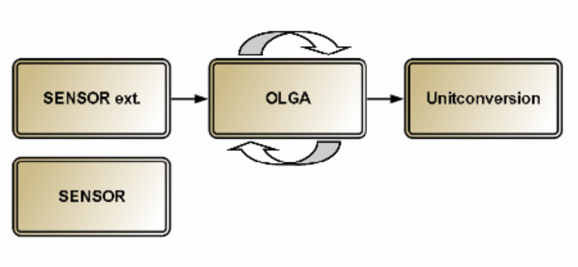


Figure 23: Schematic depiction PipeIt - initialization

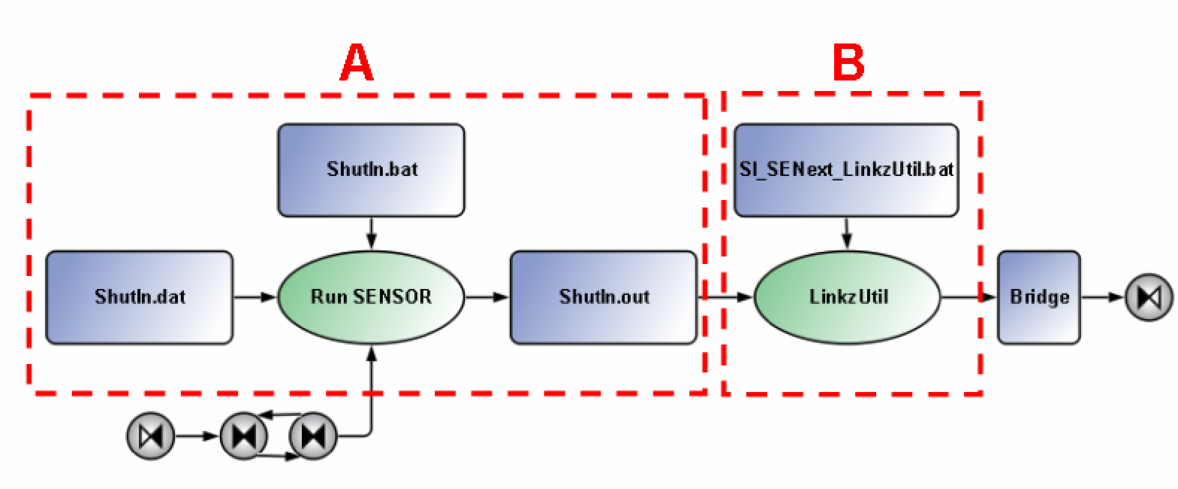


Figure 24: Pipe-It Shut-in – SENSOR

The part of the model shown above marked with "A" is just the automatic execution of the SENSOR model that reaches up to surface. The needed values for the OLGA model are taken from the output file by using PipeIt Linkz. "B" shows the execution of LinkzUtil. Within this application the BHP and THP from the SENSOR output file are transferred to the OLGA input file.

One level above (figure 23) it can be seen that the SENSOR model up to the coupling point runs also. This is necessary to get the first restart file for the start-up simulation afterwards. When a restart file is used to define the current conditions in a model, the griding between the model that has been used to create the restart file and that one that uses the restart file has to be identical. The Pipe-It model of this run is not shown here because it looks basically the same like "A" in figure 24.

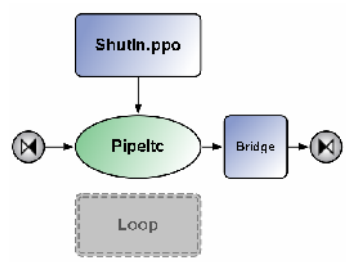


Figure 25: Trial and error OLGA

When the SENSOR model run is finished PipeIt proceeds to the composite "OLGA". This contains the trial and error runs for OLGA. This actually means that an optimizer is used to find the best fitting initial void fraction to meet the THP by SENSOR given. To launch the trial and error process a PipeItc application is used. The actual execution of OLGA happens in the composite “Loop”. Basically it looks the same like “Sensor extended” but two output files are used. The .tpl file contains the pressure values and the .out file contains the simulation time.

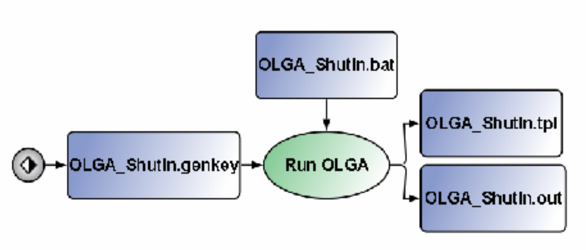
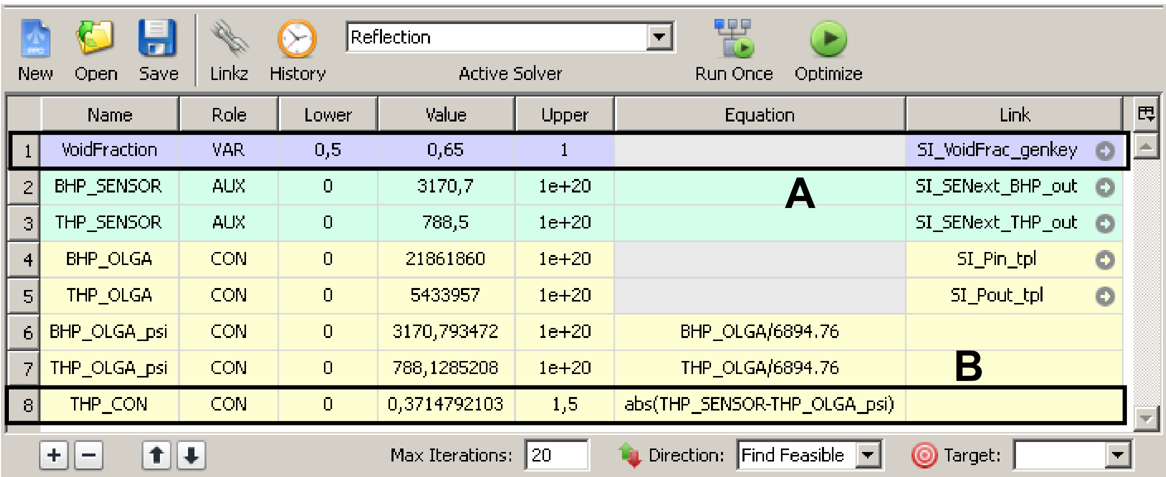


Figure 26: Execution of OLGA - Initialization

The most interesting part here is actually how the optimizer is defined. What is the goal that has to be reached? The BHP and the THP are known from the SENSOR model. These two values are used for the initial conditions in the OLGA model. But it is unknown how much gas is predominant in the tubing. The optimizer now runs several times with different initial void fractions and comes up with a void fraction that leads to a THP in OLGA with a difference of ±1 psia to that one in SENSOR. Below the used optimizer to find the accurate void fraction is shown.



	Name	Role	Lower	Value	Upper	Equation	Link
1	VoidFraction	VAR	0,5	0,65	1		SI_VoidFrac_genkey
2	BHP_SENSOR	AUX	0	3170,7	1e+20		SI_SENext_BHP_out
3	THP_SENSOR	AUX	0	788,5	1e+20		SI_SENext_THP_out
4	BHP_OLGA	CON	0	21861860	1e+20		SI_Pin_tpl
5	THP_OLGA	CON	0	5433957	1e+20		SI_Pout_tpl
6	BHP_OLGA_psi	CON	0	3170,793472	1e+20	BHP_OLGA/6894.76	
7	THP_OLGA_psi	CON	0	788,1285208	1e+20	THP_OLGA/6894.76	
8	THP_CON	CON	0	0,3714792103	1,5	abs(THP_SENSOR-THP_OLGA_psi)	

Figure 27: Optimizer for Initialization

The two most important lines in this optimizer are marked with black boxes. Box A shows the variable that is used, in this case void fraction. This is the value the optimizer changes and writes into the input file of OLGA. Box B shows the constraint that tells the optimizer when to stop. Under equation the absolute difference between the THP of OLGA and SENSOR is determined. This value is constrained to be between 0 and 1. Since the absolute difference is used the THP of OLGA can vary ±1 psia to that one of SENSOR. The maximum number of iterations is set to 20, but usually the optimizer finds a solution within the first five iterations.

3.5 Final conditions after shut-in

In the following the conditions after a 10 h period of shut-in are shown. These conditions are results of the new integrated model. All the results discussed before in the chapter shut-in refer to investigations of stand alone models either in SENSOR or OLGA.

The final THP is 809.4 psia and the final BHP is 3150.9 psia. The gas-oil contact (GOC) is approximately at 40 m depth. This can be seen in the chart. This proves that the model works correct because we end up with some free gas in the most upper section of the tubing. The rest is entirely filled with oil.

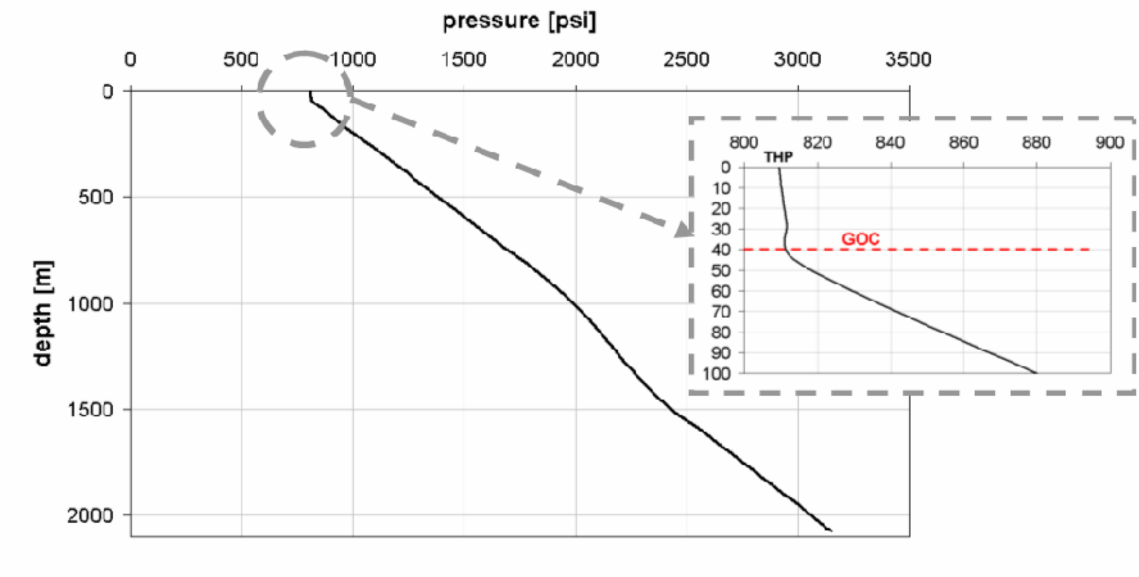


Figure 28: Final pressure versus depth for shut-in

What also needs to be controlled is, whether or not fluids have actually left the system through the bottom node. It has to be done to see whether the model works properly or not. This can be easily proved by having a look on the accumulated oil volume flow through the bottom node.

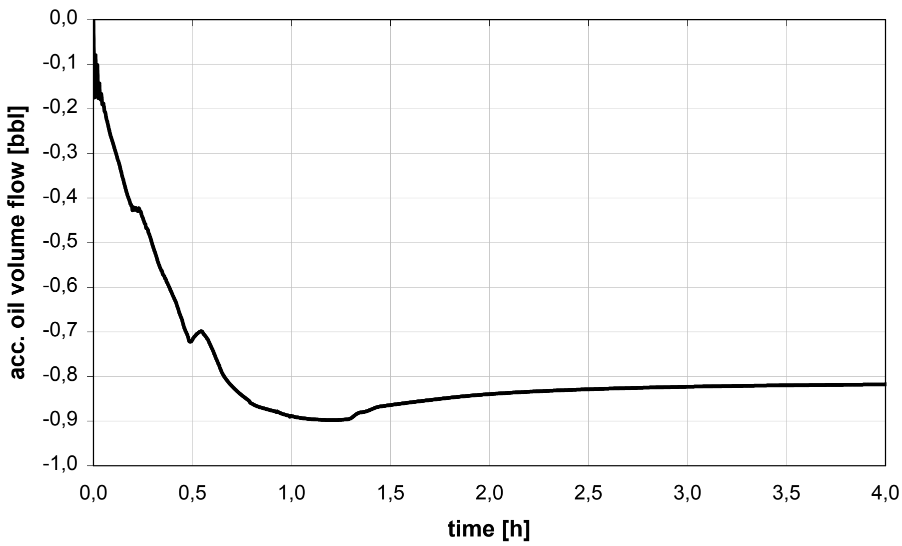


Figure 29: Accumulated flow through bottom node

In total approximately 0.8 bbl of oil leave the system during the shut-in period.

The results of this shut-in simulation show a realistic behaviour. But it is not possible to begin the start-up simulation with these conditions. This is caused by the used model for simulating start-up, a mass flow controlled bottom node and a pressure controlled top node. What would happen is that in the beginning of the start-up simulation the gas in the top section of the tubing would leave the system. Because of the used pressure controlled node at top the WHP would stay constant and the BHP would decrease. But in reality the WHP would decrease and the BHP would stay constant. This is a result of the now entirely oil filled tubing. To achieve a realistic simulation of this behaviour an intermediate step is needed. This OLGA model in between shut-in and start-up simulation is responsible to release the gas at top to end up with realistic conditions for the oil filled tubing. The OLGA model used for that step consists of a pressure controlled bottom node, to keep the BHP constant, and a mass flow controlled top node (all flow rates set to zero). Although the flow rates are set to zero the gas and probably some oil will leave the system through the top. This system is simulated for one hour. After this time all the gas has left the system. The now predominant conditions in the system are useable to begin the start-up simulation.

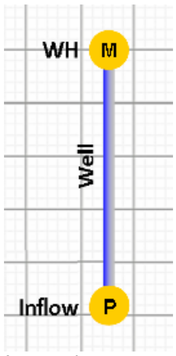


Figure 30: OLGA gas release

The BHP now is 3150.9 psia and the THP is 755.4 psia.

It is quite obvious that during this process gas will leave the system at top. But it is not sure whether oil will leave the system at top or bottom. It is a necessity to keep track of these flow rates to fully understand what happens in the system during the different processes.

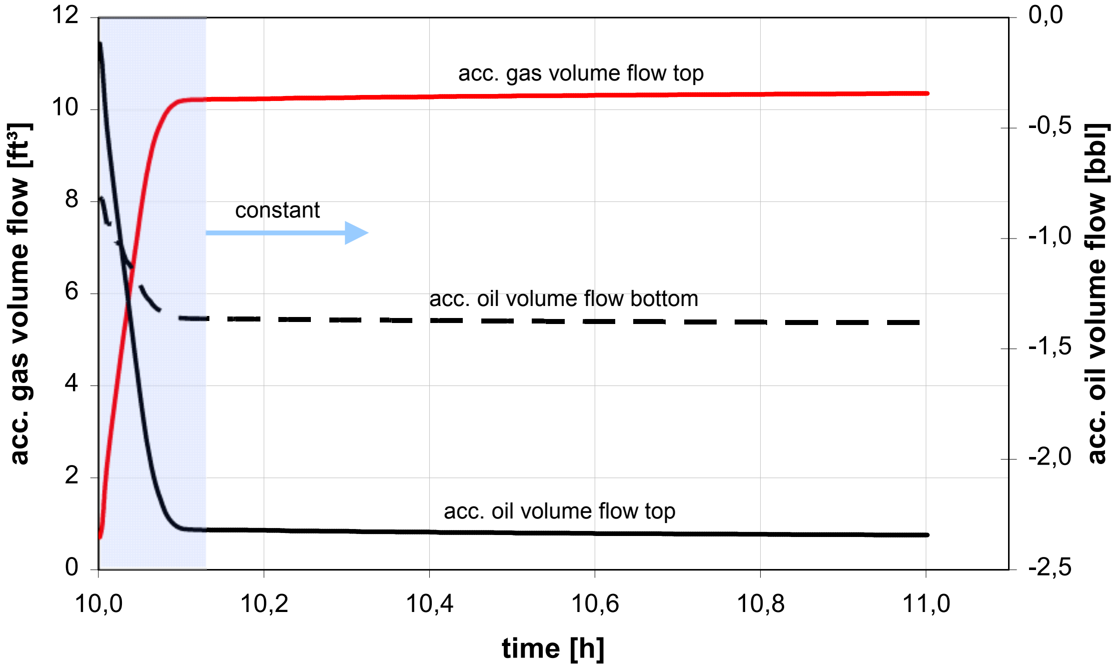


Figure 31: Accumulated low rates during gas release

The chart shows that approximately after 0.1 hours the accumulated flow rates stay constant. This does mean that the flow rates of oil and gas are zero or very little so that they can be neglected.

The model to release the gas runs automatically in PipeIt after the simulation of shut-in condition is finished. This PipeIt composite consist of a single execution of OLGA and a PipeItc application for unit conversion and data transfer.

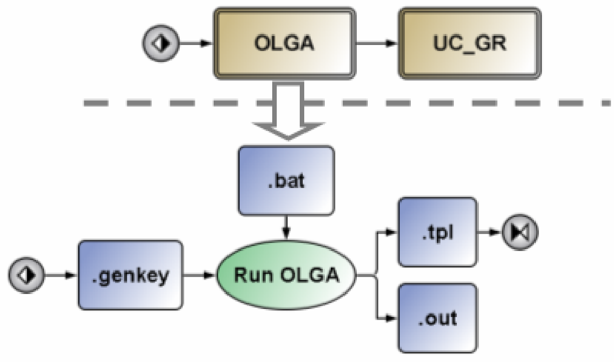


Figure 32: Depiction gas release composite

After this the initialization of the system is finished and a good starting point for the start-up simulation is obtained

4. Optimization of well start-up

In order to get a more accurate model the problem of counter current flow through the coupling point has to be solved. To achieve that the OLGA model as well as the SENSOR model currently used need to be changed. Moreover the integrated model in Pipe-It needs some additions in order to handle this flow phenomenon. These changes are described in the following.

4.1 Counter-current flow

4.1.1 SENSOR

The problem that has to be solved here is that OLGA delivers only one pressure for the coupling point. But it is necessary to produce and inject different fluids, e.g. water and/or oil, simultaneously. This is physically not possible. If only one pressure exists at one point either production or injection of fluids occurs. It is known that production and injection, or at least backflow, at the same time can occur in reality. For example for low production rates of oil it could happen that water flows back down, e.g. accumulated on the inside of the tubing wall. Another case when counter-current flow through the coupling point would be necessary is the simulation of shut-in conditions. Due to gravitational segregation oil, gas and water will flow within the tubing. Gas will flow upwards to accumulate in the top section. Water will flow downwards to accumulate at the bottom of the tubing. And oil will flow downwards in the upper section of the tubing where it is replaced by gas and maybe upwards in the lower section of the tubing where it gets replaced by water. To make simultaneous production and injection possible, the producer and the injectors had to be spatially separated. This means that the producer and injector are not located in the same depth. Due to that it is possible to define separate producing and injecting pressures. The issue here was to define an appropriate maximum injection pressure. As a default value 20 000 psi is given in SENSOR. If this value would be used every rate would be injected back into the reservoir by all means. But this is not the real physical behaviour. A feasible size of the injection pressure has to be found. It is not exactly known what really happens with the fluid that flows back down, maybe it gets produced again, maybe it gets injected into the reservoir or it accumulates at bottom of the well. To account for this unknown behaviour the pressure relation between the reservoir, the producer and the injector should be as close as possible to reality. Figure 33 shows a schematic sketch of the SENSOR model.

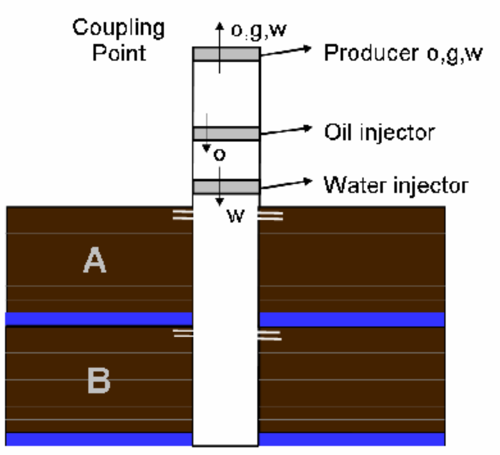


Figure 33: Schematic sketch of the SENSOR model

OLGA gives one pressure at the coupling point. This pressure is taken as the limiting producing pressure for the producer at top of the SENSOR model. This does mean that the SENSOR model tries to produce at the highest rate without dropping below the defined pressure. To get a feasible injection pressure for the oil and water injector the water/oil gradient is used. The maximum injection pressure for each phase is then defined by the producing pressure plus the additional hydrostatic pressure caused by the depth difference. To get this additional pressure the depth difference between producer and injector is multiplied with the associated pressure gradient.

$$\Delta P = \Delta depth[m] * gradient[psi/m] \text{ ....Eq. 4.1}$$

As an example the following equation describes the maximum allowed injection pressure for water ($P_{Iw\ max}$) where P_p is the producing pressure and ΔD_{Iw} is the difference in depth between the producer and water injector.

$$P_{Iw\max} = P_P + \Delta D_{Iw} * watergradi\ ent \text{ ....Eq. 4.2}$$

The schematic is shown in figure 34.

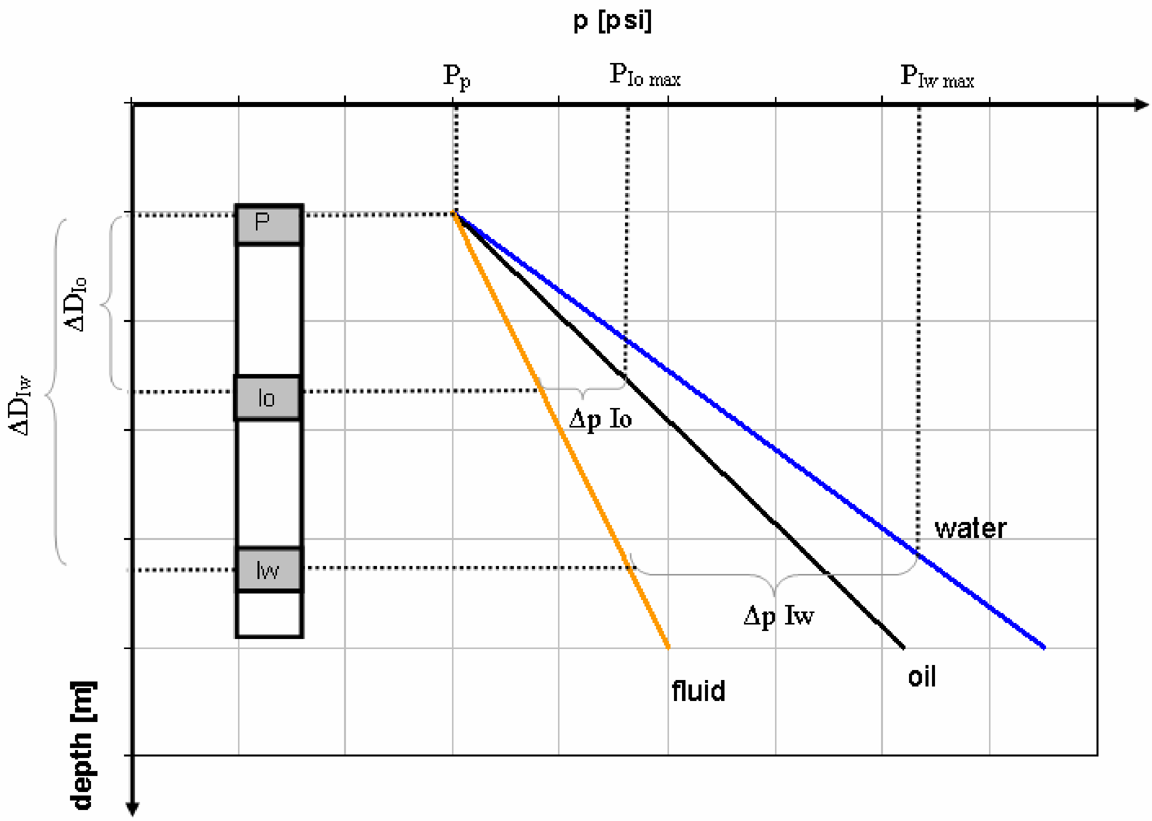


Figure 34: Maximum injection pressures

What is the reason the maximum injection pressure is defined that way? When it is known that back flow occurs it has to be ensured that the fluid actually flows downwards in the SENSOR model. This can only be ensured by maximizing the injection pressure by the method explained above. The easiest way to explain this approach is by a simple model consisting of a pipe filled with water within a tank filled with oil. A schematic depiction is shown in figure 35.

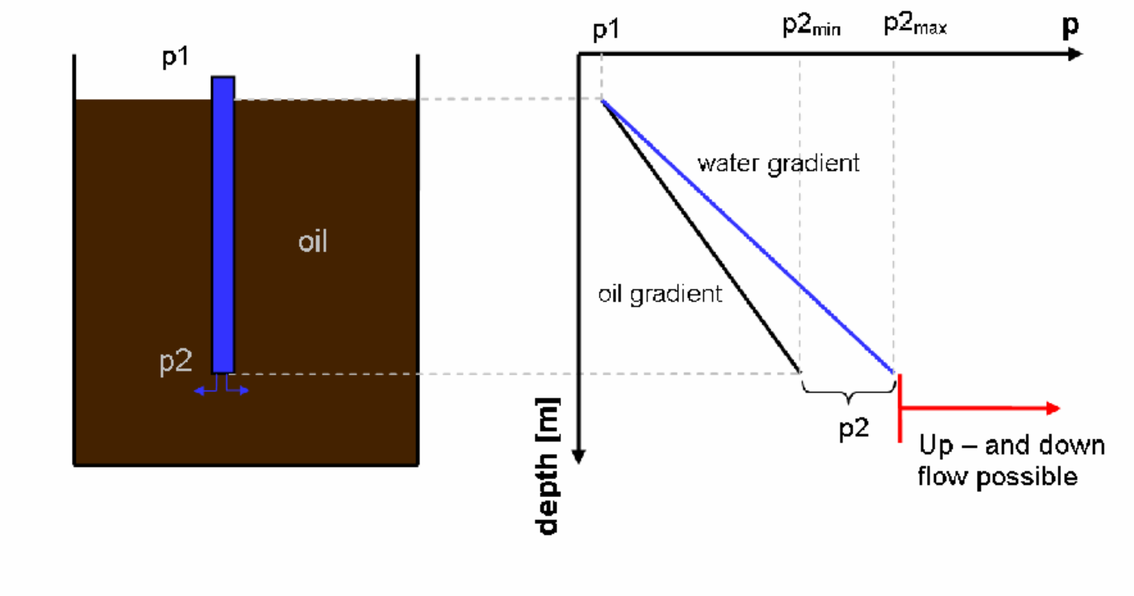


Figure 35: Definition of maximum injection pressure

The task here is to define a maximum for p2 to make sure that the water from the pipe actually flows into the tank and not upwards in the pipe. This figure shows in a very simple way the reason for the boundary set for the maximum injection pressure. As long as the pressure p2 stays within the range shown in the chart, only down flow of water is possible. But if p2 would be above this range, shown by the red arrow, the water theoretically would have the possibility to flow upwards. And this must be prevented by all means. In the actual model p1 would be the well flowing pressure and p2 would be the injection pressure.

To find the best way to realize this principal in SENSOR a few different attempts were necessary. The first question that had to be answered was how to grid the well bore and how to separate the producer and injectors spatially. Basically there are two possibilities. The first possibility is that the producer is separated from the injector in z-direction only. This does mean that the injector and the producer are both modelled as circles (A). The other possibility is that the well bore is grided in x-direction also; this imposes that the well bore consists of three rings. The two outer rings are used as a producer and only the most inner ring is used as an injector. This separation in x-direction is done additionally to the difference in depth (B).

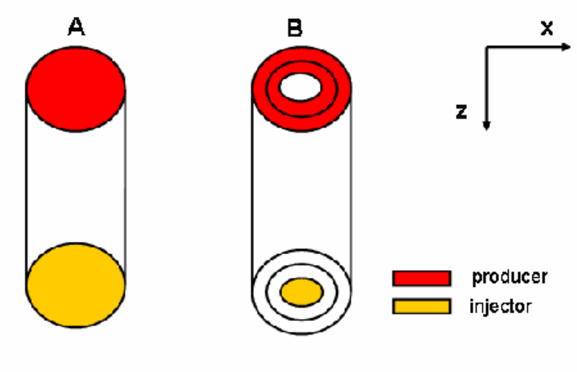


Figure 36: SENSOR model for injection and production simultaneously

Both types have been investigated. Due to the fact that in model type B the well-bore is grided with more cells consequently the result of the simulation is slightly more accurate than with model A. But this difference in accuracy is not significant.

The following chart shows the difference in results for using either model A or model B. The important input values are: Injection rate of water is 20 [bbl/d], production rate for oil is 80 [bbl/d], maximum injection pressure is 3188 [psi] and minimum pressure at the producer is 2845 [psi]. The chart does not show the output values for the produced oil and the injected water. The reason is that these two values stay constant at their input value over the simulation period of one day.

The full lines in figure 37 represent the results for model type A and the circles show the results for model type B.

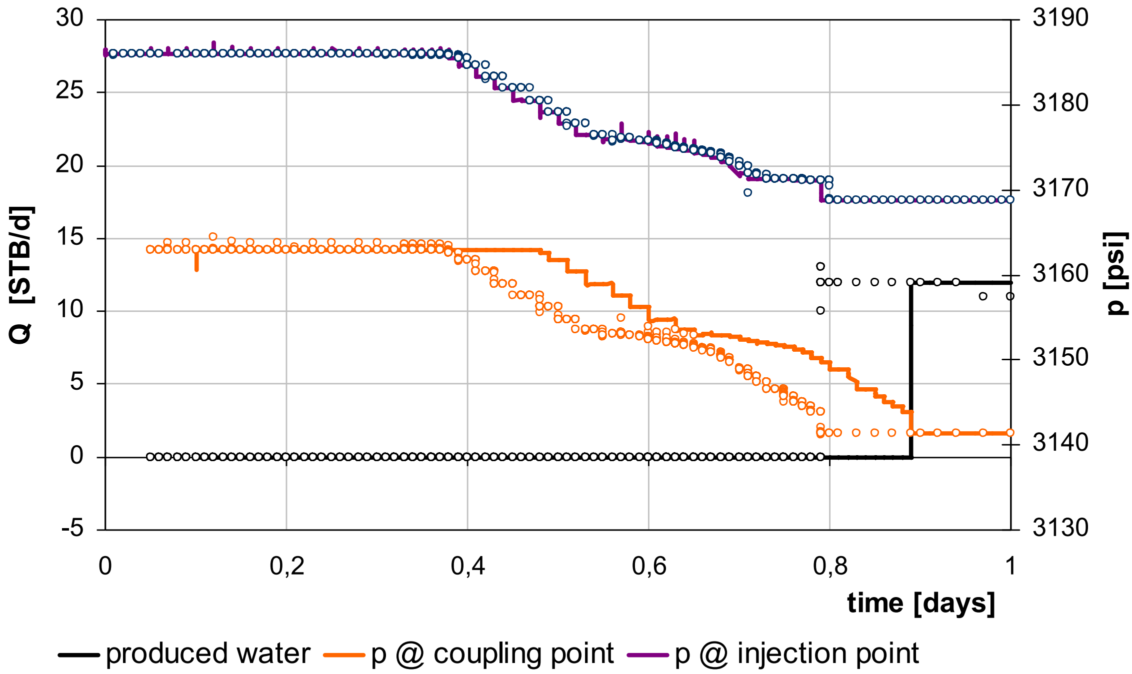


Figure 37: Comparison of model A and model B in accuracy

What can be recognized is that there is hardly any difference in the result for the pressures at the injection point between model A and model B. If attention is put on the pressure at the coupling point (=point of production) a shift of the line representing model B to the left can be seen. This shift in the production pressure also influences the starting point of water production. Just like the coupling point pressure line also that one for the water production is shifted to the left for model B. As a conclusion it can be stated that there are differences in the results for the differently grided models, but they are not significant. Having only a look on the accuracy of the models the first choice would be model B. But if also the needed CPU time to run the simulation is taken into consideration the first choice is model A. This is caused by the fact that through the additional grid cells and the corresponding increase in grid cell boundaries the needed CPU time increases three times compared to that one needed for model A. In this case the small inaccuracies can be neglected with clean conscience to get the advantage of less simulation time with model A.

Another issue that has to be investigated is the way the injector influences the producer. In SENSOR a limiting pressure or a limiting flow rate can be defined. In this case the producer is limited by pressure, this does mean that SENSOR tries to maximize the production without falling below the given production pressure. In case of the injector a maximum allowed injection pressure is given and additionally the desired injection rate is set to 50 STB/d. The result for a simulated time of one day is shown in figure 38

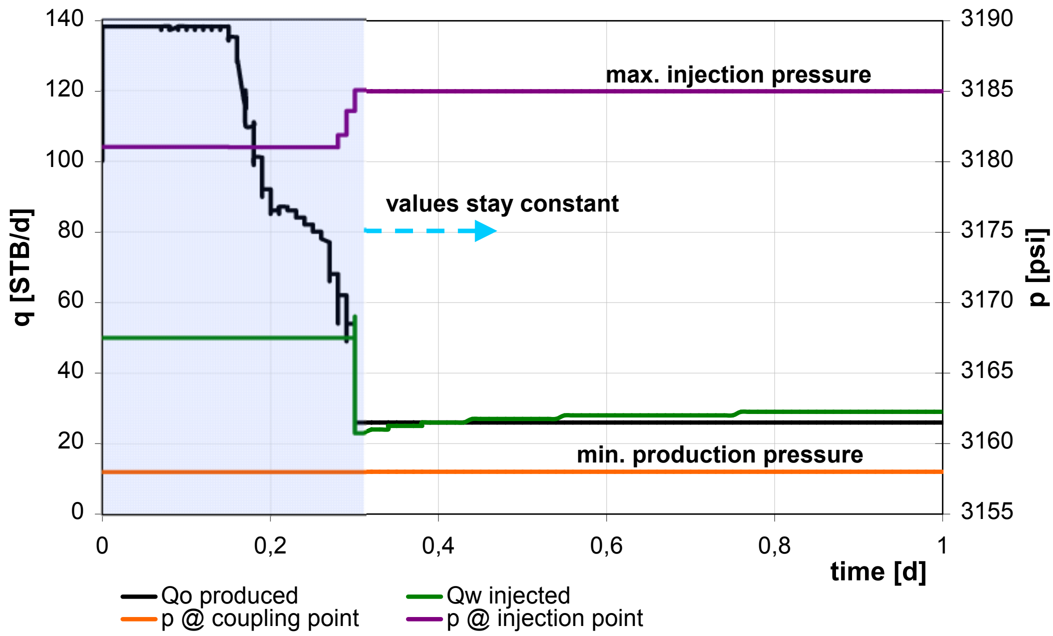


Figure 38: Influence of the injector on the producer

This chart shows that SENSOR starts straight with the given minimum production pressure. This value stays constant over the entire simulation period. But the oil flow rate starts to drop significantly quite early. This is caused by the fact that the reservoir is just not able to produce the rate from the beginning over time with this given minimum production pressure under the given conditions. When putting attention on the injection it can be seen that in the beginning the desired injection rate gets injected with a lower injection pressure than the given maximum value. In this case the given injection rate is the limiting factor. At approximately 0.3 days the pressure increases quite fast to its maximum. This causes a drop in the injection rate to a value lower than the desired injection rate. Now the maximum injection pressure is the limiting factor. After that point the injection pressure and the production pressure stay constant. Also the producing oil flow rate stays constant. The injection rate increases a little bit again, but not significant.

When having a look on a scenario with no water injection shows that the well produces at a constant oil rate of 98 bbl/d. The pressure at the producer also stays constant at the input value.

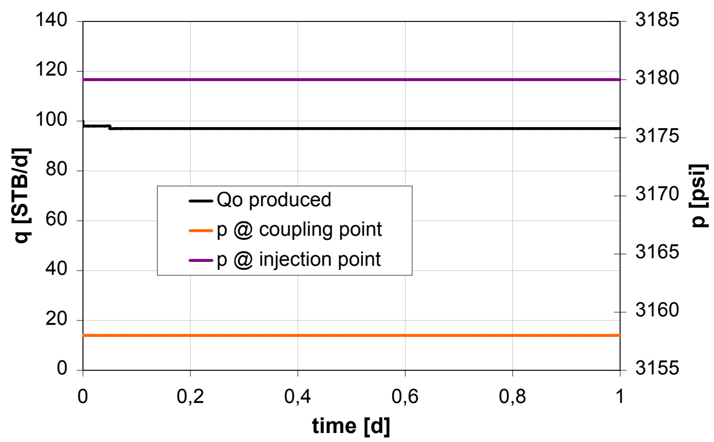


Figure 39: Flow behaviour with no injector

As a conclusion it can be stated that with this principal counter-current flow through the coupling-point in SENSOR is handled like an injector. If backflow through the coupling point occurs SENSOR takes this flow rate and uses it as an injection rate.

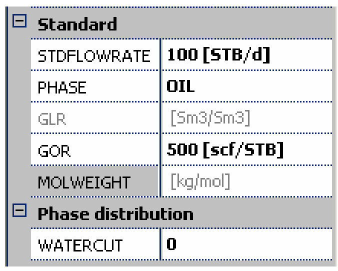
4.1.2 OLGA

In terms to get an integrated model for well start-up OLGA has to be able to calculate negative flow rates through the bottom node if they occur. This is a necessity to handle counter-current flow through the coupling point. When using the steady-state pre-processor this is not possible (cf. chapter 2.1.2).

To get the possibility to handle this effect the SSPP has to be turned off. Subsequently the dynamic OLGA is used for calculations. With the dynamic OLGA it is possible to end up for example with positive gas flow rates and negative oil flow rates simultaneously. This is exactly the desired behaviour needed from OLGA.

With the currently used reservoir model the probability to get counter-current flow is quite low. But if other scenarios are simulated counter-current flow is an important step to get a model that is very close to reality. In terms to test the model an experimental case has been set up. In the beginning of the simulation positive gas flow and negative oil flow is predominant. This proves that OLGA is able to calculate negative flow rates through the bottom node although it is mass flow defined.

Because the most likely case is that water flows downwards water needs to be introduced to the system. This should be possible with little effort because at the inflow node an oil flow rate, the GOR and WCUT can be defined. But due to a bug in OLGA this is not possible. If the WCUT is defined at the inflow node OLGA ends up with zero WCUT after a very short period of time. This bug has been reported to the SPT Group Support Centre. They will try to get rid of this bug by the next release of OLGA.



Standard	
STDFLOWRATE	100 [STB/d]
PHASE	OIL
GLR	[Sm3/Sm3]
GOR	500 [scf/STB]
MOLWEIGHT	[kg/mol]
Phase distribution	
WATERCUT	0

Figure 40: OLGA input inflow node

This bug makes a change in the system necessary. In order to end up with the defined water flow rate the water needs to be fed into the system via a source. The changed system is shown in figure 41. Now everything should work properly. But by investigating the behaviour in more detail it was discovered that there is another problem. OLGA multiplies the water inflow rate, defined via the source, with one thousand. For example if a flow rate of 1 STB/d is defined OLGA ends up with 1000 STB/d. This seems like a minor problem at first sight but this bug leads to some inconveniences within the integrated model set up in PipeIt. The reason is that the calculated water flow rate from SENSOR cannot just be transferred to OLGA. It needs to be divided by 1000. The associated inconveniences caused in the PipeIt project are discussed in chapter 1.3.

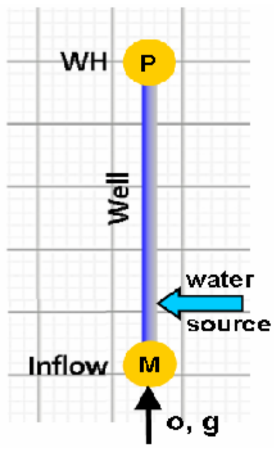


Figure 41: OLGA model for start-up

NOTE: *All discovered bugs in OLGA has been reported to SPT Group support centre and will help to improve the next release of OLGA. This is one of the typical ways how software gets improved. All users report inconveniences or bugs and SPT tries to correct them.*

The experimental case that has been set up in order to achieve positive and negative flow rates of different phases simultaneously is explained in the following. An inflow of 100 STB/d of oil is defined. The associated GOR is set to 500 scf/STB. This kind of inflow leads at the beginning to negative flow rates of the oil phase. The gas flow rate is right from the beginning positive.

The results of the simulated run are shown in the figure 42.

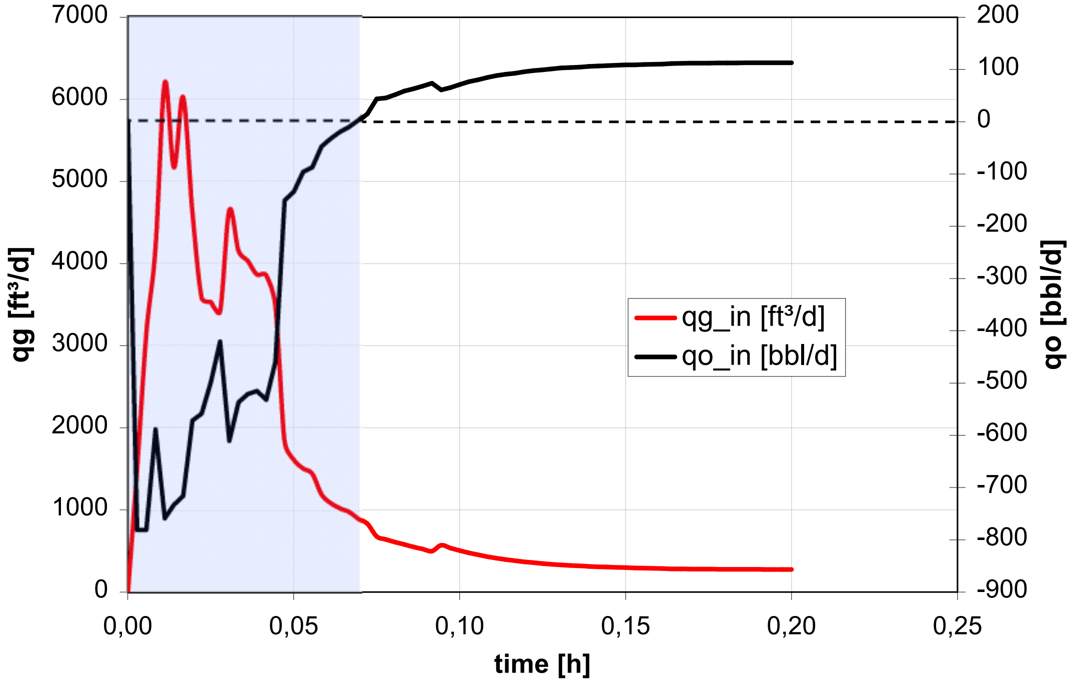


Figure 42: Counter-current flow through bottom node in OLGA

It can be seen that in the beginning of the simulation, highlighted in blue, the oil flow rate is negative and the gas flow rate is positive. This kind of result proves that OLGA is able to solve for negative flow rates through a node.

NOTE: *To encounter negative flow rates in OLGA the flow rates need to be calculated in reservoir conditions. OLGA does not calculate negative flow rates in standard conditions. Negative flow rates are zero for standard conditions.*

With this OLGA model and the before explained SENSOR model it is now possible to build an integrated model in PipeIt that is capable to deal with counter-current flow through the coupling point if necessary. This brings the model one big step closer to a model that can represent real behaviour during start-up.

4.2 Pipe-It Restart runs

This chapter deals with the actual optimization of the start up process. It gives an overview of how the optimizers work and it deals especially with the changes to the current model.

The goal of the optimization is to start up the well as quickly as possible but within certain boundaries which are defined in the optimizer. Those boundaries are chosen in a way that the well is not damaged or, as a worst case scenario, is killed during the start up. Problems that need to be avoided are:

- onset of sand production caused by too high changes in bottom-hole flow-rate within a certain period of time ($\Delta q/\Delta t$)
- damage to gravel packs or other down-hole equipments
- flooding of surface facilities (test separator) by liquid slugs (too high increase of the surface oil rate within a certain period of time)
- falling below the minimum operating pressure for surface facilities

4.2.1 Constraints

Too ensure that those problems will not occur the following constraints are set:

- max. change in inflow rate per minute at bottom [$-10 < \Delta Qo_{Sensor} < 200$ STB/d-min]
- max. change in outflow rate at top per minute [$10 < \Delta Qo_{Surface} < 400$ STB/d-min]
- max. change in BHP per minute [ΔBHP ±20 psi/min]
- min. THP [THP>500 psi]

Additionally the control variable (CV), in this case change in THP, has to be within ±50 psi/min. Moreover a constraint is defined [ΔTHP ±10 psi/min] to keep control activities at the well-head low. All those constraints are set in the restart loop optimizer.

In order that Pipe-It knows when the optimization is finished, final conditions of the system need to be defined. This is the main responsibility of the Head Master Optimizer (HMO). The HMO decides after each TS if the restart loop has to run again or whether the optimization is finished and steady-state flow in the entire system is predominant. To achieve this goal constraints are necessary. As long as these constraints are not fulfilled the HMO will run the restart loop again. Otherwise Pipe-It will go to the next higher level and by doing that the optimization is finished. Following constraints has been set to define steady-stat flow:

- surface flow-rate has reached target flow-rate
  [$-2.5 < Qo_{out,OLGA}$– target flow-rate < 2.5 STB/d]
- well-bore storage effect is not longer predominant
  [$-50 < Qo_{,Sensor} - Qo_{out,OLGA} < 50$ STB/d]
- no changes in BHP and/or THP in between TSs anymore
- no changes in flow-rates in between TSs, neither down-hole nor at top

Additionally to all those production specific constraints other constraints had to be specified, e.g. to ensure model stability. Both optimizers, the restart loop optimizer and the HMO, are described in detail in the appendix.

4.2.2 Restart runs Modus Operandi

The way the optimization has changed compared to the current model is basically only that now counter-current flow is also taken into consideration within the restart loop optimizer. If negative flow rates at the coupling point are encountered the integrated model takes these negative flow-rates from OLGA and uses them as injection rates in the SENSOR model. The model in SENSOR has been changed to be able of producing and injecting fluids simultaneously. The injection simulates the back flow of fluids into the reservoir. The SENSOR model that can handle counter-current flow is described in chapter 4.1.1. Also the OLGA model that is in use now to handle counter-current flow is described in chapter 4.1.2.

The deepest level of the composite "restart runs" is the actual optimization of the change in THP for each TS. One level above the Head Master Level (HML) can be located. The restart run optimizer gets its input data from the last TS of the HML. It then tries to find the best change in THP to stay within the boundaries given (cf. beginning of this chapter). This new THP pressure is then handed over to the HML. The restart run optimizer runs as long until it finds a solution for the new THP that fulfils all constraints. In order to keep the run time of the model in a reasonable range the maximum number of iterations for the restart loop optimizer is set to 20. This means if the optimizer has not found a perfect solution, which meets all requirements, within 20 iterations, it will take the best fitting value and hand over this value to the HML. The HML than runs only once in order of data recording and to see if final conditions (cf. beginning of this chapter) are reached. If final conditions are not reached yet the Pipe-It model will go back to the lower level and the circle starts from the beginning. If final conditions are predominant the Pipe-It model will jump one level above and subsequently the optimization of the well start-up is finished. The restart loop is always one step in front of the HML. It is easier understandable when having a look on figure 43.

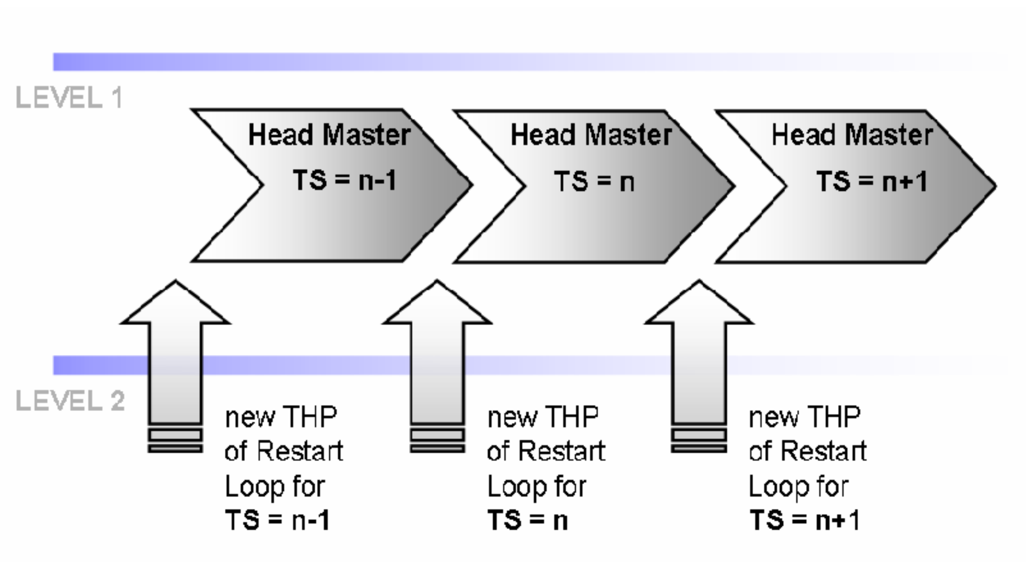


Figure 43: Modus operandi optimization

This however is the same system as used in the current model. Changes that have been necessary are influencing in first order the optimization within the restart loop.

Basically in one restart loop SENSOR and OLGA run alternately until a solution is found. The restart loop starts with the execution of SENSOR with the given BHP from the last HM OLGA run and calculates the corresponding flow-rates. Current data transfers between the models consist of oil flow-rate, GOR, WCUT and BHP. Qo, GOR and WCUT are handed over from the SENSOR model to the OLGA model and the BHP is handed over from OLGA to SENSOR. This data had to be changed caused by the different models that are used now and because of bugs in OLGA. In the new model SENSOR gives OLGA Qo, GOR and Qw. WCUT had to be replaced by water flow rate due to a bug in OLGA (cf. chapter 4.1.2).

However the biggest difference is given by the input data of SENSOR when the optimizer starts. In the new model negative flow-rates at bottom, encountered at the end of the last HM TS, are now introduced to the SENSOR model to capture the effects caused by negative flow-rates. Together with this injection rates also corresponding injection pressures need to be defined. How the injection pressures are defined is described in chapter 4.1.1. All this is handled now within the restart loop optimizer and leads after all to a more accurate simulation of well start-up.

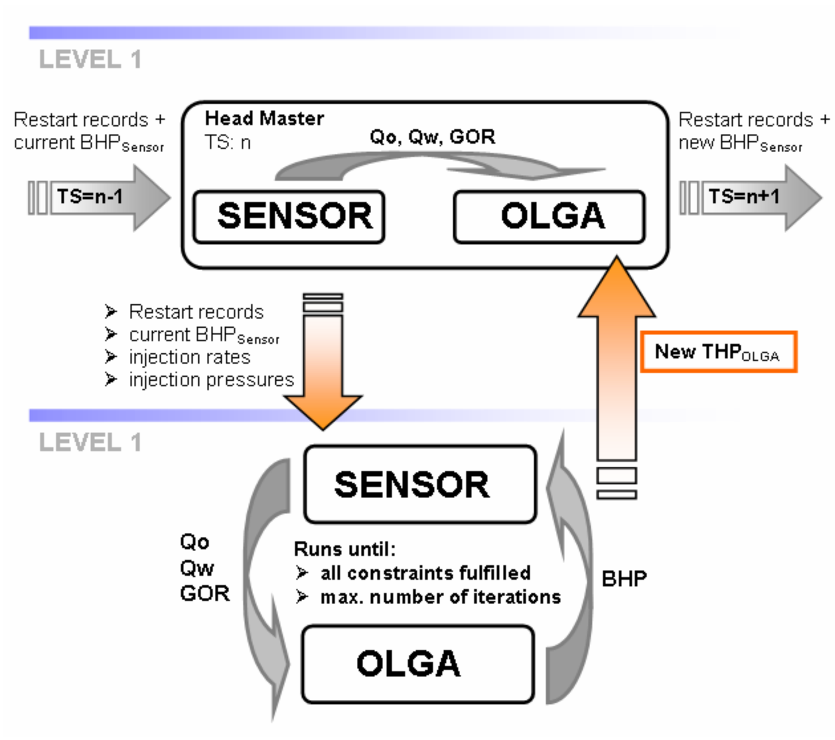


Figure 44: Restart Loop

The additions that have been necessary for the restart loop optimizer are shown in the appendix. [14]

To get more information about how the restart loop and the HML work together please refer to “Optimization of Well start-up”. [15]

[14] Appendix / Detailed description of the optimizer / Restart Loop Optimizer / Lines 13 -24

[15] Michael Schietz, Master Thesis “Optimization of well start-up”

4.3 Comparison between new and current model

To certify that the new model has improved the simulation of well start-up, the results of the two models have to be compared. Before this is done a few remarks are needed. Due to the fact that the new model is now able to handle counter-current flow the SSPP can not be used anymore. It is replaced by the dynamic solver of OLGA.

In the current model a slight mistake has been encountered by doing the PVT calculations. However this mistake leads to a significant different fluid definition in SENSOR. PVT calculations have to be done because OLGA and SENSOR use different parameters to define the fluid. The starting point to calculate PVT parameters for SENSOR is given by the PVT fluid file used in OLGA. Based on this data the needed parameters for SENSOR are calculated. But to get a meaningful comparison between the current and the new model the old PVT data for SENSOR is used here.

NOTE: *The PVT data including the error from the current model is also used for the new model to get a meaningful comparison.*

In the following the results of the current model for a start-up simulation with fixed TSs of 15 sec and variable TSs is shown. These results are then compared to the results of the new model.

4.3.1 Results of the current model

Figure 45 shows the result of a start-up simulation for an oil target rate of 7200 STB/d using fixed TS length (15 seconds).

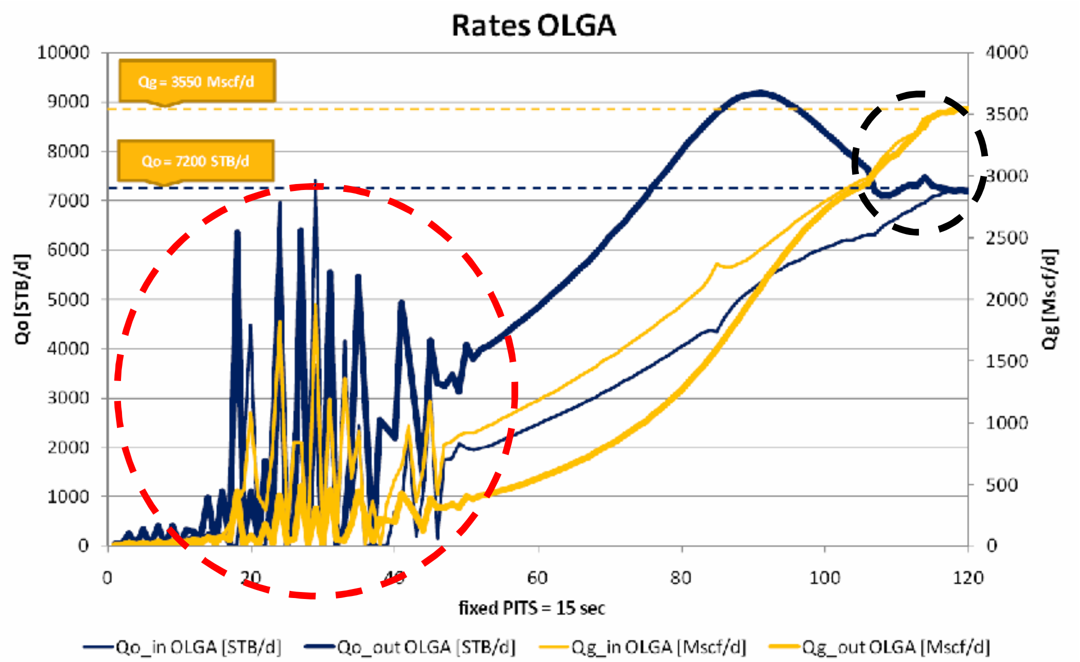


Figure 45: Current Model results fixed TS length [16]

What can be seen here is a highly erratic behaviour of the flow-rates in the beginning of the simulation, highlighted with the dashed red circle in figure 45. After a certain time the behaviour becomes much smoother. This erratic behaviour in the beginning led finally to the use of variable TS lengths. To get more information about the different attempts with fixed and variable TS length in the current model please refer to “Optimization of well start-up”. [17]

[16, 17] Michael Schietz, Master Thesis “Optimization of well start-up”

At the end of the simulation run, highlighted with the dashed black circle in figure 45, steady-state flow throughout the entire system is achieved. On the x-axis the number of time steps used to start up the well is shown. Multiplying this with the TS length of 15 seconds leads to an overall start-up time of 1800 seconds (=30 minutes) to reach the target rate of 7200 STB/d.

When changing the model from fixed TS length to variable TS length the erratic behaviour in the beginning vanishes. Figure 46 shows the start-up simulation for a target oil rate of 4000 STB/d using variable TS length.

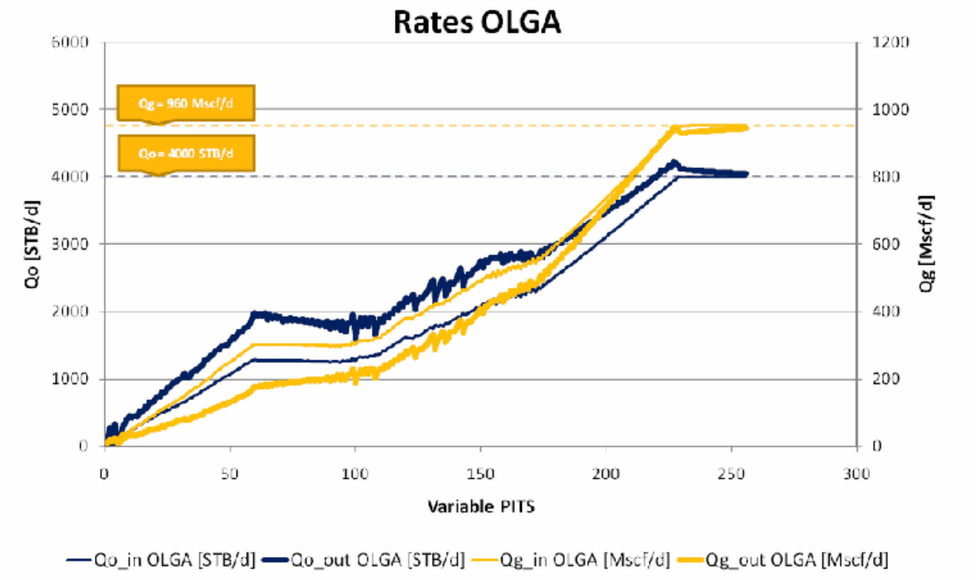


Figure 46: Current model with variable TS length [18]

Another difference besides vanishing of the erratic behaviour in the beginning is that the increase in surface oil flow-rate (Qo_out OLGA) is not linear anymore over the simulation length. It shows much more a step-wise behaviour. This actually indicates that the model does not work the same way when changing from fixed TS to variable TS, otherwise also the variable TS length result would show a linear increase in surface oil flow-rate.

Another observation that can be made is the difference in start-up time. The simulation shows that approximately 250 time steps are needed to start-up the well (target rate = 4000 STB/d). The average TS length over the entire simulation is 16 seconds. This means that it takes 4000 seconds (=67 minutes) to finish the start-up. Compared to the start-up simulation done for a target rate of 7200 STB/d it takes almost twice the time to start-up the well although the target rate is much lower. The results are significant different. This leads to the assumption that the current model is not consistent in terms of optimization results.

NOTE: *By using either fixed TS length or variable TS length the surface flow-rate shows a significant different behaviour. Moreover the start-up times for the wells are not consistent.*

To get more detailed information about the interpretation of results with the current model please see “Optimization of well start-up”. [19]

4.3.2 Results of the new model

When having a look on the start-up simulation conducted with the new model a very smooth behaviour over the entire simulation length can be recognized. This is most likely caused by the fact that the SSPP is replaced by the dynamic solver of OLGA. To get a good comparison to the current model a scenario with fixed TS length and variable TS length were simulated.

[18, 19] Michael Schietz, Master Thesis “Optimization of well start-up”

The same integrated model in Pipe-It like for the current model is used to simulate the well start-up. The most significant change was made by replacing the SSPP with the dynamic solver in OLGA. This obviously leads to quite different results when simulating well start-up in an integrated model like this.

Figure 47 shows a well start-up simulation with fixed TS length of 15 seconds and a target surface oil rate of 4000 STB/d.

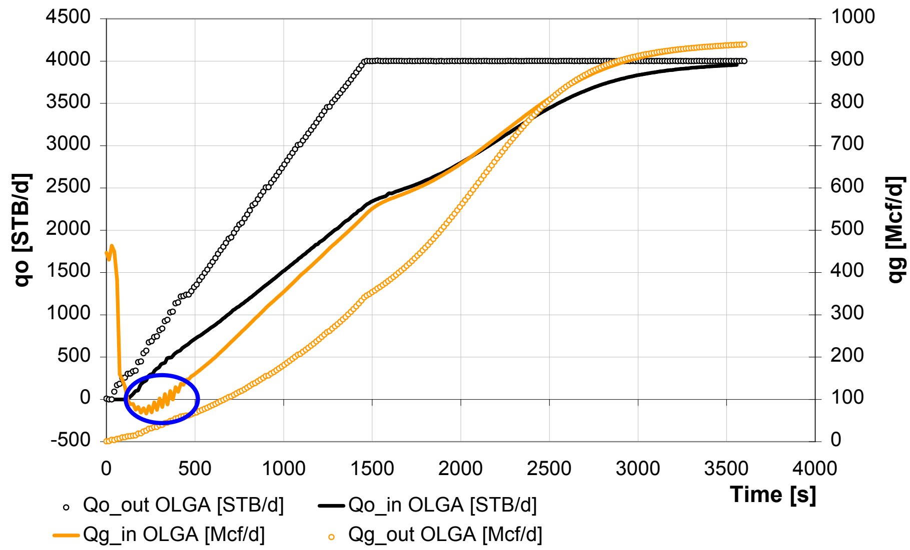


Figure 47: New model fixed TS length of 15 sec.

The increase in surface oil flow-rate is linear until it reaches the target oil rate (=4000 STB/d). Subsequent it stays constant. At the end of the simulation in- and outflow-rates for oil and gas are equal (=steady state flow). There is only a very short period in the beginning where the gas inflow-rate shows a slightly erratic behaviour (highlighted with the blue circle), the rest is totally smooth. Moreover also the corresponding bottom-hole inflow-rates show a very smooth behaviour.

To see if there is a change in results if variable TS length is used figure 48 shows the result for simulating with variable TS length.

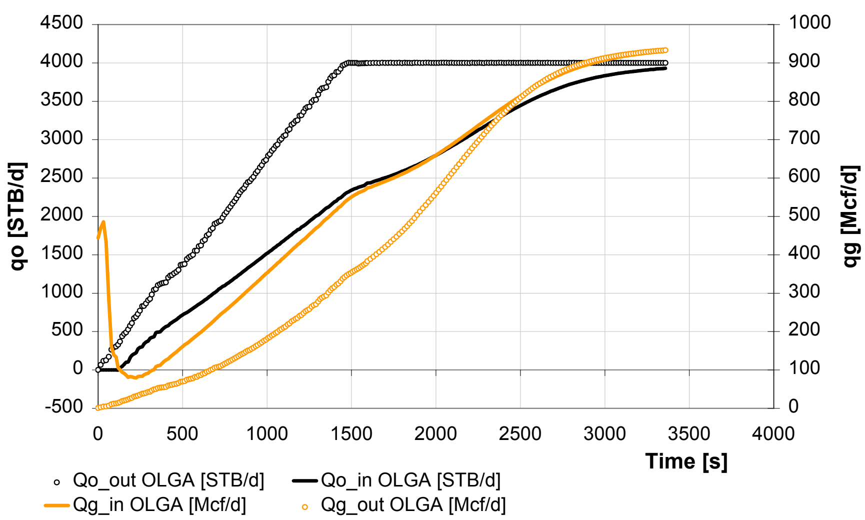


Figure 48: New model variable TS length

It is quite clear that it does not matter if fixed TS length or variable TS length is used in the new model. It leads to the same results in terms of in- and outflow-rates for oil and gas. The only advantage of variable TS length regarding flow-rates is that the slightly erratic behaviour of the gas inflow rate has vanished.

The results of the new model are not discussed in further detail because the PVT data with the error is used. Due to that the simulation does not show real physical behaviour. The aim of this comparison is only to show the differences in results obtained with the current and new model.

4.3.3 Conclusion

Due to using the dynamic solver in OLGA the new model leads to very smooth simulation results for the integrated model. Obviously for this kind of integrated simulation it makes a difference if the steady-state pre-processor is used or not. The results when using the SSPP are highly dependent on whether using fixed or variable TS length.

NOTE: *It makes a significant difference whether the SSPP or the dynamic solver in OLGA is used.*

However the new model shows a highly consistent behaviour independent from the chosen kind of TS length, fixed or variable. The question arising is, if it still makes sense to use variable TS length. To find an answer, the results of the new model are investigated in more detail.

4.4 Fixed or variable TS length in the new model?

As seen before the difference when simulating well start-up with the new model between fixed and variable TS length is not significant, at least when having a look on the in- and outflow-rates. Other important parameters of the well start-up process are the BHP and THP. Both pressures are shown in figure 49 for both variable and fixed TS length.

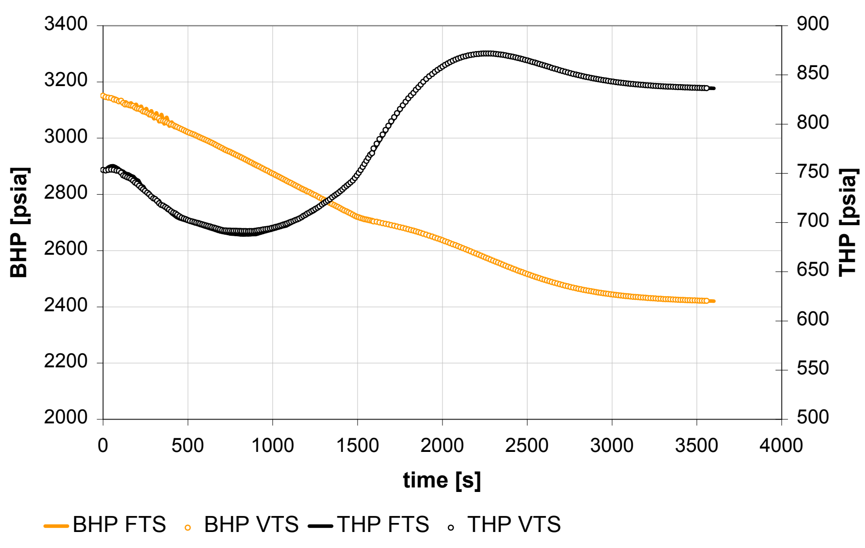


Figure 49: BHP and THP comparison FTS and VTS

It can be seen that the BHP and THP for both TS length methods are nearly identical. Results for production specific parameters like flow-rates and pressures are practically independent from the chosen TS length method (fixed or variable). But not only production specific parameters are important in an integrated model. Like for the chosen constraints (cf. chapter 4.2.1) also simulation specific parameters are important, e.g. the stability of the integrated model itself. The best way to investigate the model stability in this case is to have a look how good the optimizer works. This can be done by having a look on the pressure and flow-rate time derivatives. If the model is stable the values should be within the given boundaries during the entire simulation. This kind of investigation shows nicely the differences in stability of the models itself with no respect to production specific parameters. The next two charts show the change in THP and

BHP in psia/min for variable and fixed TS length. Also the chosen boundaries can be seen in the figure, which are:

- ± 20 psia/min for the BHP (orange)
- ± 10 psia/min for the THP (grey); this is the constraint for change in THP/min
- ± 50 psia/min for the THP variable (black); this is the range within the optimizer can choose

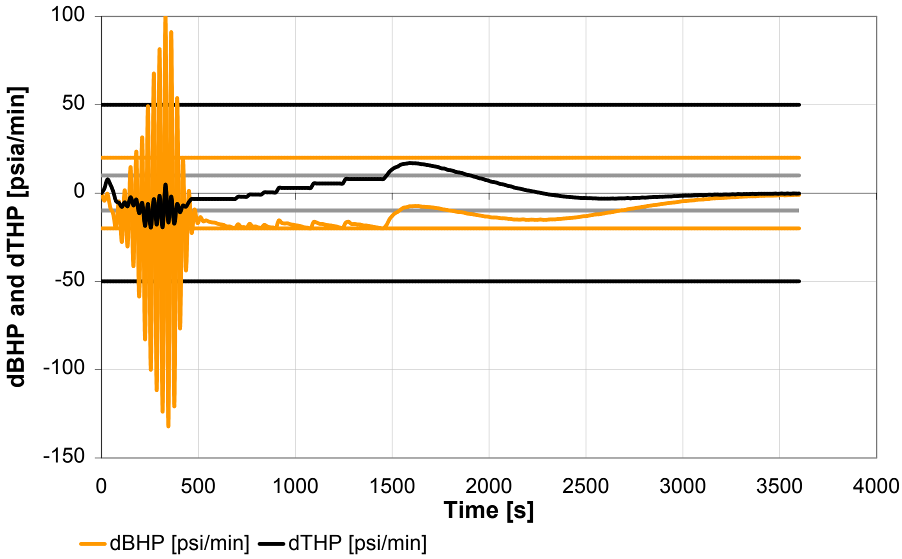


Figure 50: Pressure change per minute for fixed TS length

It can be seen that the change in BHP/min in the first 500 seconds is quite unstable when using fixed TS length. The shape of the BHP time derivative indicates numerical instabilities and not real physical behaviour. This shows that the model has some problems in the beginning to get on the right track, or with other words it is unstable. Having a closer look on the period of interest in figure 51 (first 500 seconds) will reveal the overall shape of the instabilities in more detail, indicated with the dashed blue line.

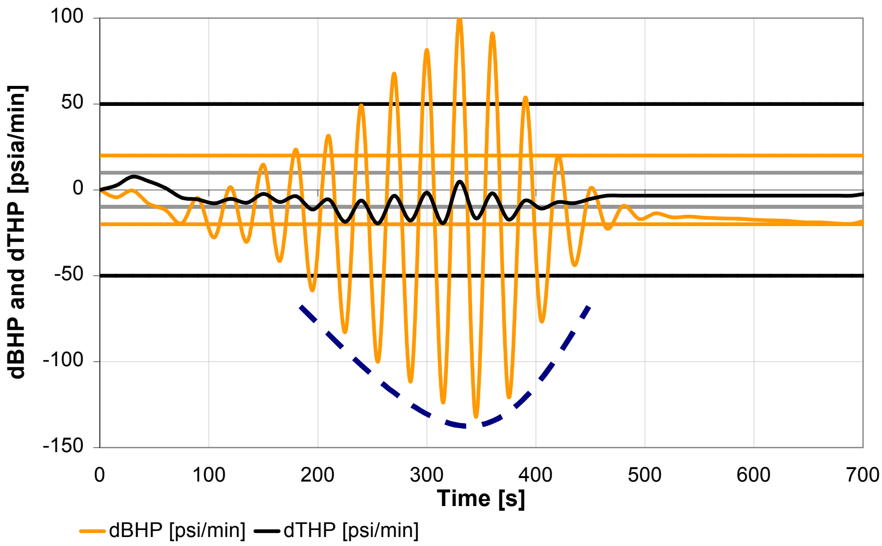


Figure 51: Shape of instabilities

When using variable TS length the instabilities vanish. This indicates that the optimizer is able to achieve better results for each TS within the maximum number of iterations. This does mean that the model is more stable. Based on this investigation the decision is made to use variable TS length.

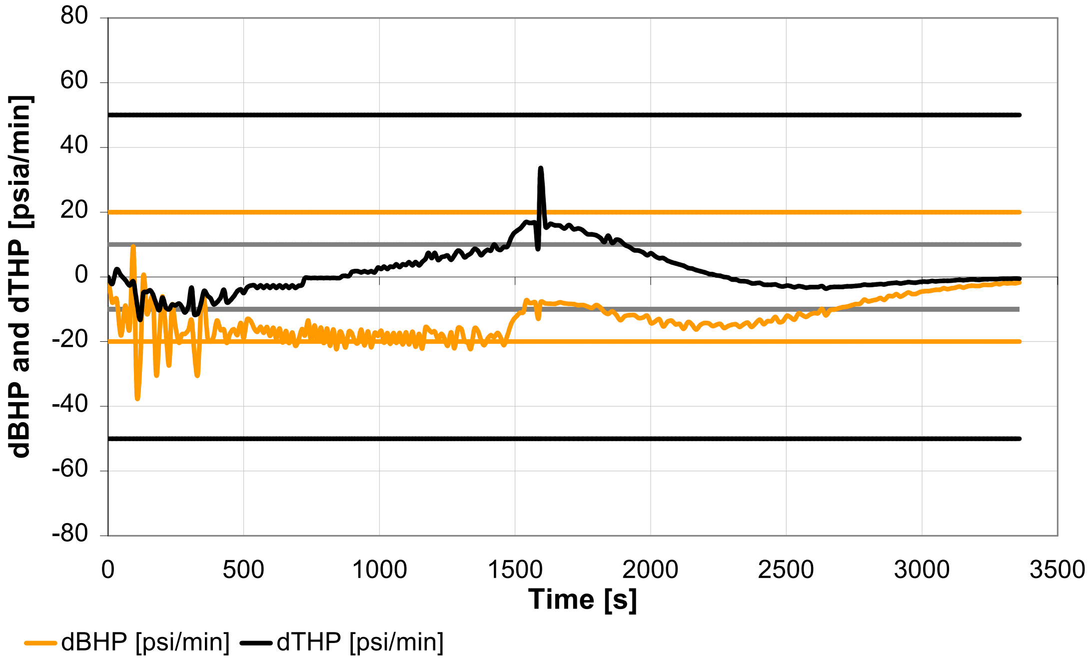


Figure 52: Pressure change per minute for variable TS length

The same behaviour is shown when having a look on the flow-rate changes per minute at bottom and surface.

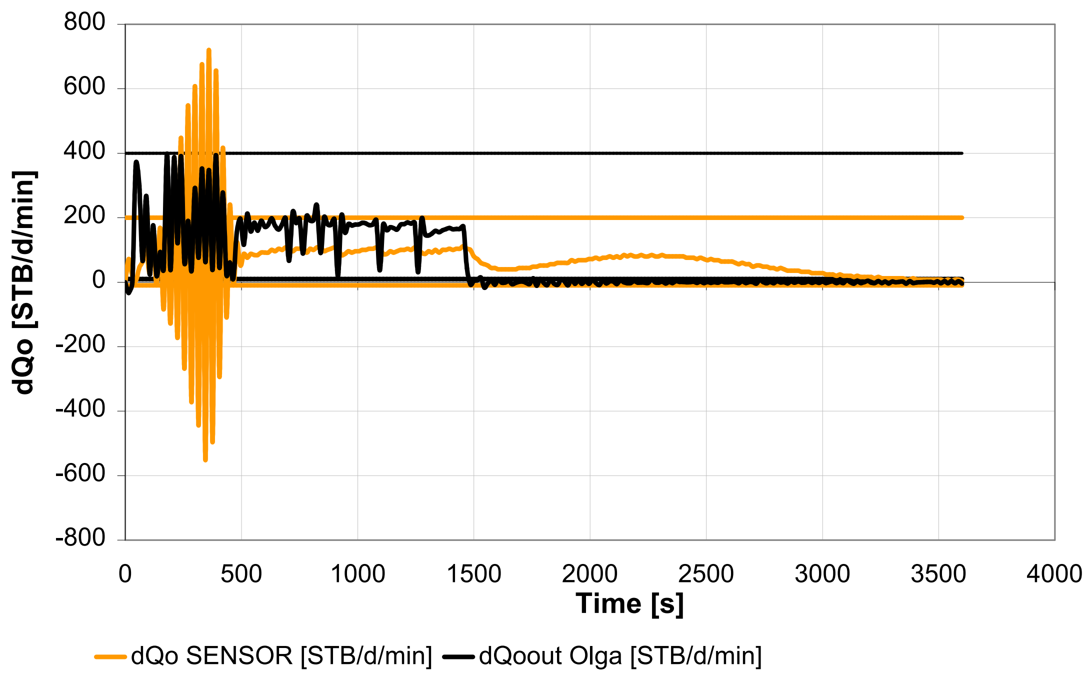


Figure 53: Flow rate change per minute for fixed TS length

Corresponding to the BHP time derivative the inflow-rate time derivative shows the same kind of instabilities. The boundaries in figure 53 and 54 are: [-10 < dQo SENSOR < 200 STB/d/min (orange)] and [10 < dQooout Olga < 400 STB/d/min (black)].

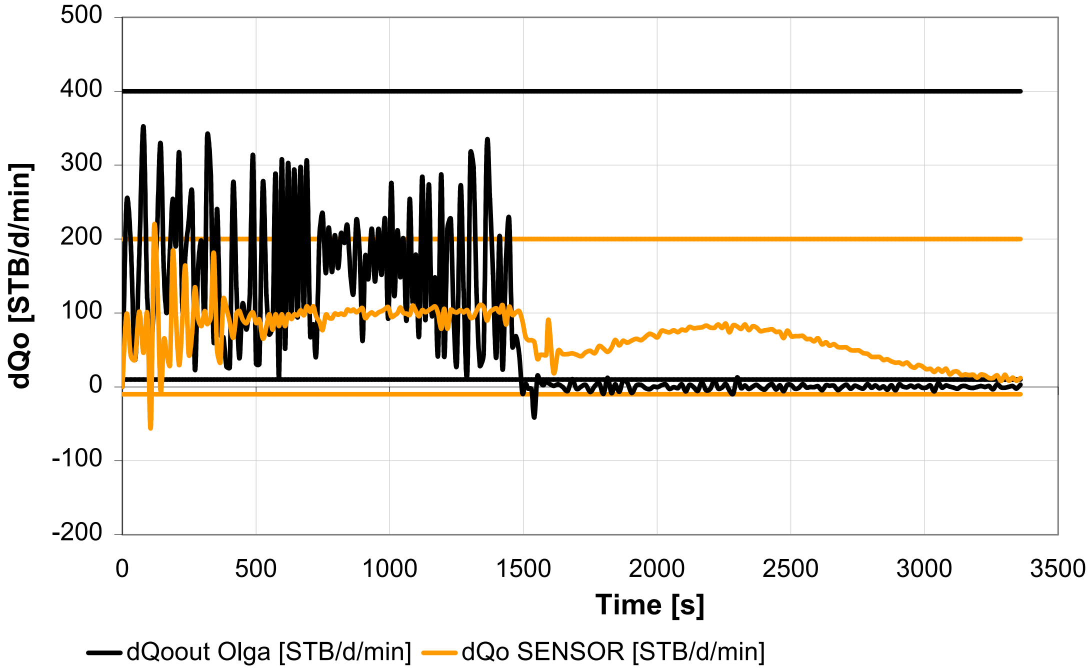


Figure 54: Flow rate change per minute for variable TS length

Using variable TS length also leads to a stable behaviour in terms of flow-rate time derivatives. Only a few minor peaks violate the given boundaries.

NOTE: *Using variable TS length leads to a much more stable behaviour of the integrated model.*

5. Results using correct PVT data

The last chapter discussed the differences of the current model and the new one. It was pointed out that the old PVT data was used. Old PVT data refers to the PVT calculations with the error. The error itself was that for the standard gas density 0.44 kg/m³ was used instead of 1.44 kg/m³. This looks like a minor mistake in the beginning, but it has a huge effect on the results of the PVT calculations. The most obvious difference is stated in the Rs value of the SENSOR model. It changes from 352.4 scf/STB to 1036.93 scf/STB at reservoir conditions. Subsequent also the GOR in the SENSOR model rises from 237 Mcf/STB to 755 Mcf/STB. This changes the flow pattern in the tubing dramatically, caused by the increased amount of gas. Due to that the complexity of the calculations in the OLGA model increases and subsequent the total time needed to simulate the start-up increases.

The outflow-rate at surface still shows a linear increase until it reaches the target value (=2000 STB/d) but the inflow-rates show a different behaviour. It can be seen that the oil inflow-rate shows a plateau period (highlighted with the blue rectangle). The reason for this is discussed in the following.

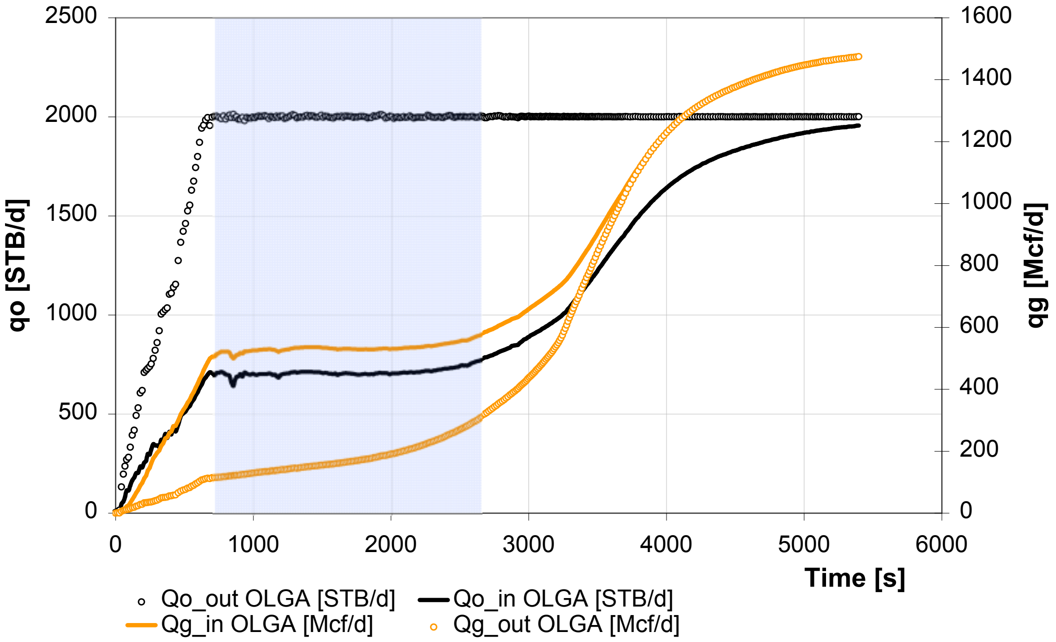


Figure 55: Rates vs. Time

The increase in surface oil flow-rate shows a linear behaviour until it reaches the target oil rate of 2000 STB/d. After reaching the target rate it stays constant. The oil inflow-rate also shows a linear behaviour until the oil outflow-rate reaches the target value. Subsequent it also stays a certain period of time quite constant. This is caused by the much higher gas content that is now predominant in the system. During this period the THP pressure increases linear and the BHP stays constant. Also the final conditions for steady-state production are changed. The final THP is 1546 psia and the final BHP is 2741 psia for a flow-rate of 2000 STB/d. At the end of the simulation the difference between inflow- and outflow-rates approaches zero.

The length of the plateau period is driven by the constraints set in the optimizer. Within Pipe-It the optimizer tries to increase the inflow-rate as fast as possible. However it is not possible to increase the inflow-rate directly after the model has reached the target oil outflow-rate without violating the defined constraints. The optimizer has to wait until the GOR value at top within the OLGA model has increased enough to increase the inflow-rate at bottom again.

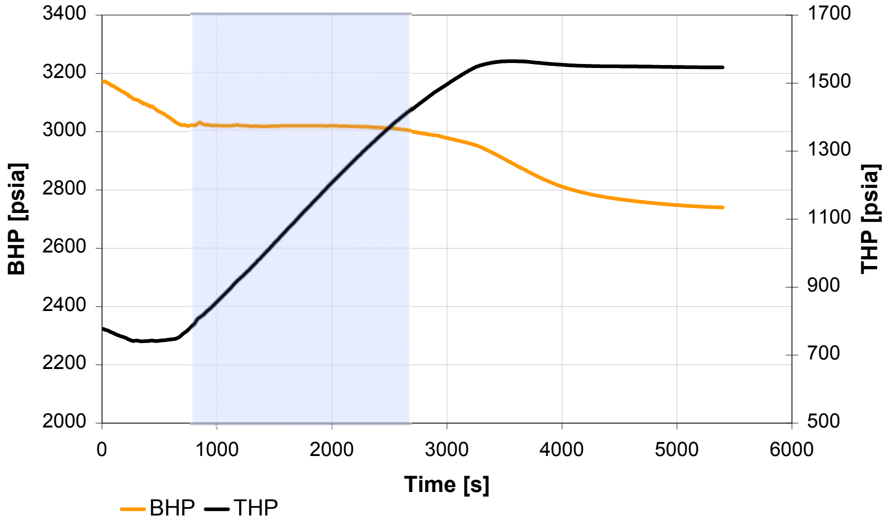


Figure 56: Pressure vs. Time

The pressure chart shows the behaviour of the THP and the BHP.

The THP drops slightly in the beginning of the simulation until the target oil out-flow rate is reached. After that a linear increase is shown. This linear increase covers the plateau period of the oil inflow-rate but also a certain time afterwards. When the BHP starts to drop again the THP can practically stay constant at its current value. This is the time when the oil inflow-rate starts to increase again. The BHP stays constant during the plateau period. Afterwards it starts to drop again until it reaches its final value.

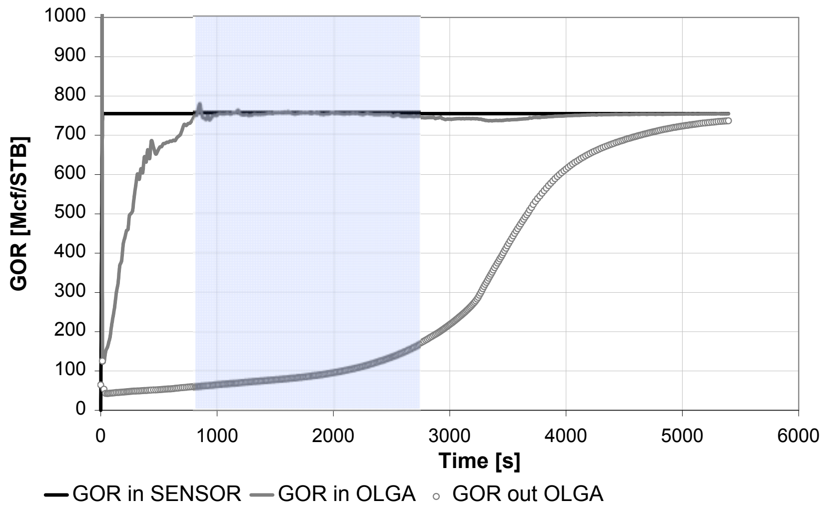


Figure 57: GOR vs. Time

The GOR chart shows three different GORs. The “GORin SENSOR” is the calculated GOR value in the SENSOR model at the coupling point. The “GORin OLGA” is the calculated GOR value at the coupling point of the OLGA model. It can be seen that it takes some time until these two values equilibrate. But they are equal exactly when the oil outflow-rate reaches its target value. The “GORout OLGA” value refers to the calculated GOR value in the OLGA model at surface. It can be seen that during the plateau period the increase in the GOR at top is very slow. Subsequent it rises quite quickly to finally reach the same value as the “GORin OLGA” value at the end of the simulation.

The behaviour of the oil inflow-rate at bottom is actually driven by the interrelation of BHP, THP and GOR. The objective of the optimizer after reaching the target oil surface flow-rate is to increase the oil inflow-rate at bottom to finally end up with steady-state flow. The plateau develops because it is not possible to lower the BHP within this time period. But this would be needed to increase the inflow-rate. The problem is that if the BHP would be lowered the target surface rate would not stay constant. It is needed to wait until a kind of equilibrium between BHP, THP and GOR is predominant. This period of waiting is driven by the increase of the GOR at top. Having a closer look on the oil flow-rates in the system shows the following:

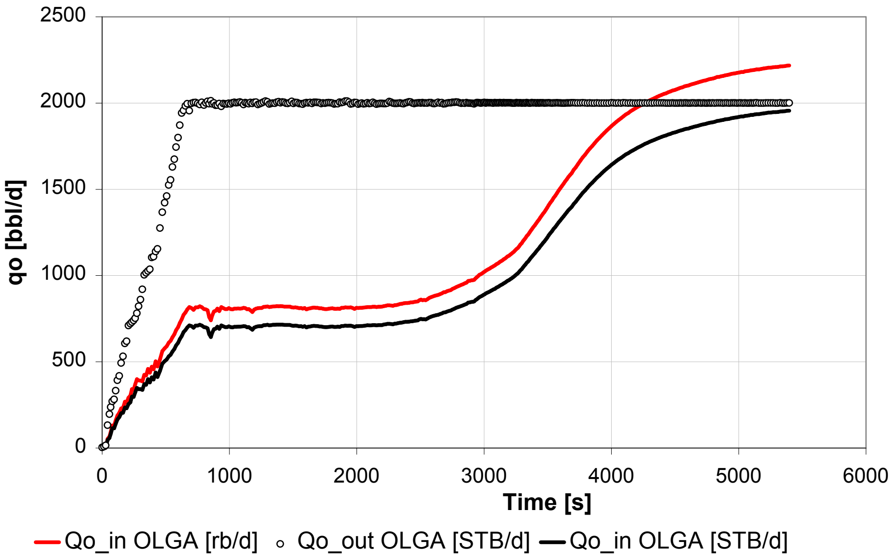


Figure 58: Oil flow-rates vs. Time

The oil inflow-rate in rb/d is also shown. The reason is that OLGA does not calculate negative flow rates in standard conditions. The chart shows that in this scenario no negative oil flow rates are detected. Also the behaviour of the integrated model itself has to be investigated again. To achieve a meaningful result the time derivatives for flow-rates and pressures are shown. Also the boundaries set in the optimizer are depicted in figure 59.

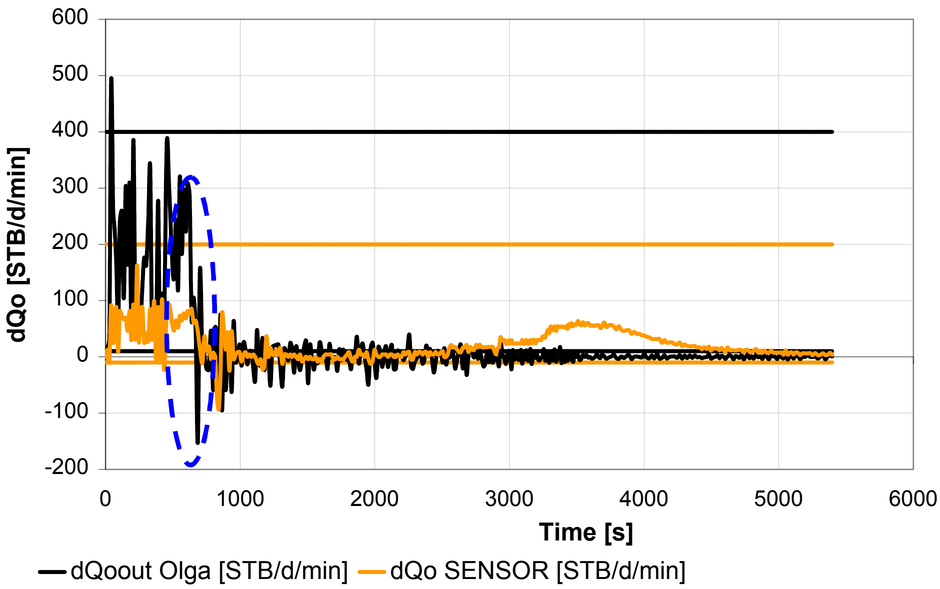


Figure 59: Flow-rate time derivatives vs. Time

A sharp drop in the oil outflow-rate derivative, shown by the black line, can be seen after the target oil rate is reached (highlighted with the dashed blue ellipse). Subsequent it oscillates around zero to stay within the constraints given for the outflow-rate.

Also the time derivative for the bottom hole flow rate oscillates around zero for the before mentioned plateau period. After that it increases again to finally approach zero when steady state conditions are reached.

Another observation that can be made is that the flow-rate time derivatives stay within the given boundaries over the entire simulation. This indicates a stable behaviour of the optimizer also when the new PVT data is used.

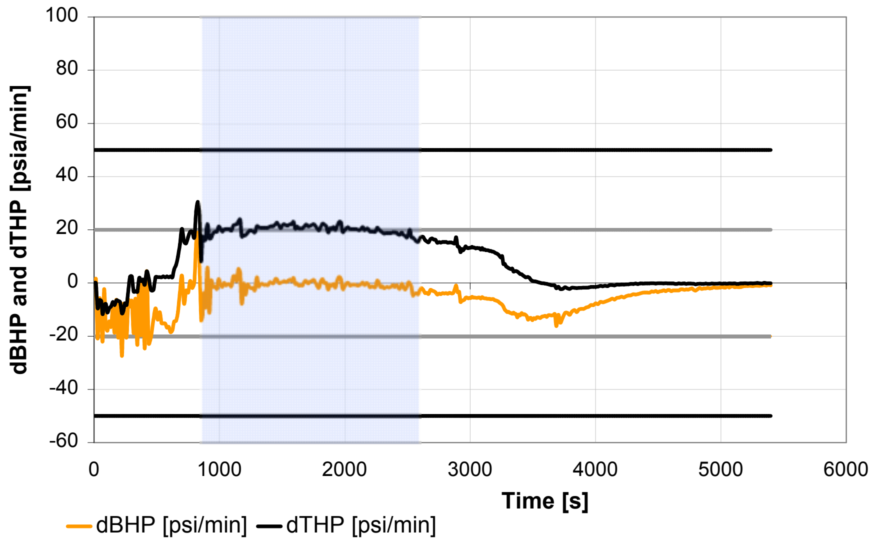


Figure 60: Pressure time derivatives vs. Time

The pressure values also stay within the given boundaries over the entire simulation. What can be seen here is that during the plateau period the value for the THP time derivative hits the upper boundary constantly to achieve an as fast as allowed increase in THP. At the end of the simulation both values approach zero (steady-state flow). To find the most time consuming periods during the simulation process, the needed number of iterations for each TS and the corresponding TS lengths are shown in figure 61:

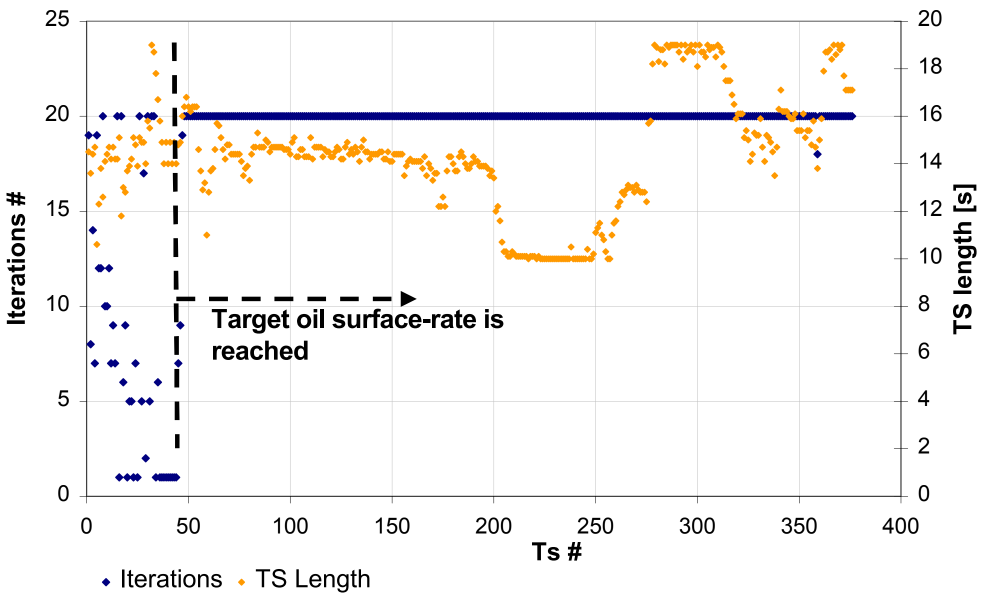


Figure 61: # of iterations and TS lengths

It can be seen that when the model has reached its defined target oil outflow-rate at surface the optimizer hits for every time step the maximum number of iterations allowed. Before that it takes most likely around 1 to 10 iterations. That means that after this point the optimization effort really increases because it is very difficult for the optimizer to stay within the given boundaries. But as shown in the time derivative plots the optimizer stays within the given constraints and that indicates that the given maximum number of iterations (20) is sufficient to achieve reliable results.

5.1 Counter current flow

Although in this specific case no counter current flow was discovered, it can be controlled if the procedures implemented in the Pipe-It project to deal with back flow are working. The two important parameters that have to be monitored are:

- the maximum injection pressure for oil and water in the Sensor model
- negative flow-rates at bottom in the OLGA model

The way how the maximum injection pressures for oil and gas are defined is discussed in chapter 4.1.1.

To see if this data is transferred correct in Pipe-It figure 62 shows the BHP and the corresponding maximum injection pressures for each TS.

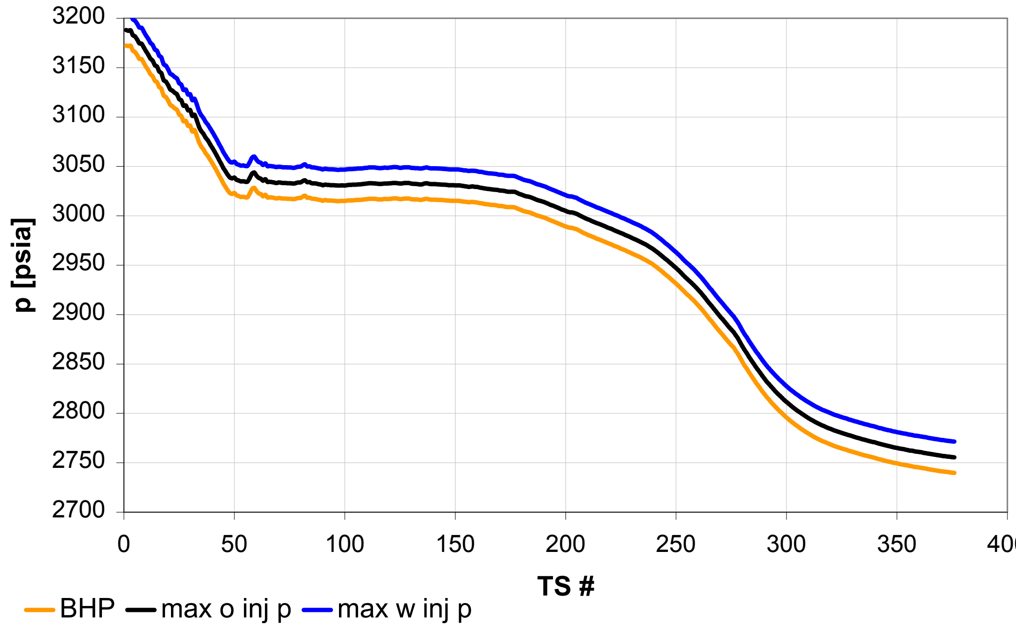


Figure 62: Maximum oil and water injection pressures

The chart shows that the maximum injection pressures for oil and gas follow the BHP with a constant shift. It proves that the maximum injection pressures for each TS are evaluated and transferred correctly within the Pipe-It project.

No negative flow-rates are predominant in the new model when using the correct PVT data. But to see that the new Pipe-It model is able to detect negative flow-rates, the following chart shows the inflow rates at bottom for the PVT data with the error. By using this data negative flow of the oil phase occurs in the beginning of the simulation.

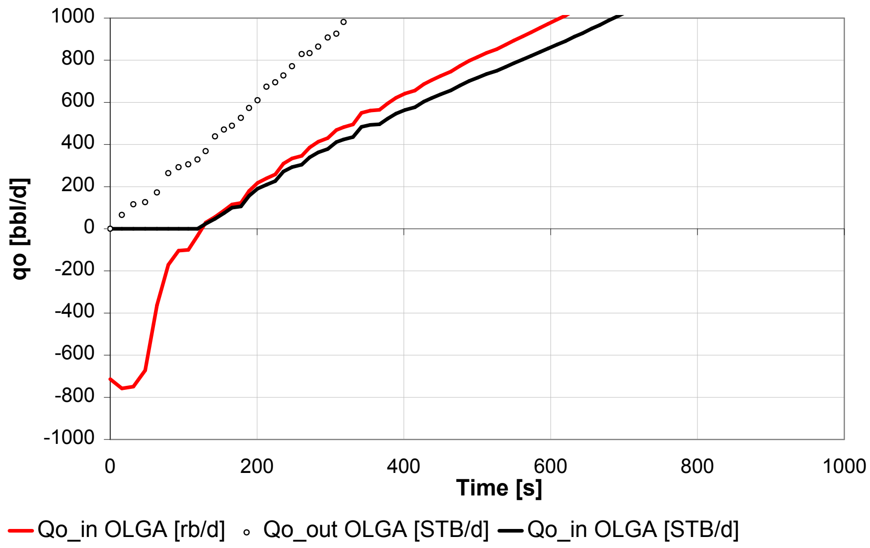


Figure 63: Detection of negative flow rates in OLGA

Focusing on the period of interest reveals negative flow-rates in the beginning of this simulation. Although this most likely does not show real physical behaviour it proves that the integrated model is able to detect negative flow-rates at the coupling point.

NOTE: *If in OLGA negative flow-rates have to be detected the flow-rates in rb/d have to be investigated. The corresponding flow-rates in standard conditions are zero.*

After detection these flow-rates are transferred to the SENSOR model as injection rates.

5.2 Water in the System

Another issue that is worth to mention is whether or not the system is now able to deal with water. This is important because of the encountered problems regarding water production in the previous chapters. It has to be checked if the reservoir and the well bore model work in a way that the integrated model leads to reliable results. With the until now used reservoir model water production will not occur. To check the integrated model the reservoir has to be adjusted in a way that water production becomes possible.

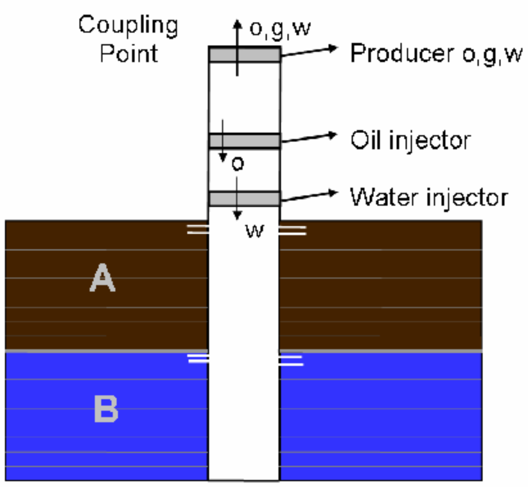


Figure 64: Reservoir model for water production

Figure 64 shows the adjusted reservoir model. The entire reservoir layer B is now filled with water. Moreover it is slightly over-pressurized to give an extra boost to the water production.

Otherwise the time needed for a start-up would be not long enough to see actual water-production.

The best scenario to test water production would be to have in majority oil production with a WCUT of 10 – 20 %. But what happens here is that in the beginning only little oil from the reservoir is produced and when the water reaches the coupling point the oil production from the upper reservoir is killed and only water is produced anymore. This is caused by the fact that the system gets forced to push water through the coupling point although this does not represent physical behaviour. To fill the lower reservoir layer with water and over-pressurize it is the only way to see water at the coupling point during the start-up period. Other attempts for example to increase the initial water saturation in reservoir and tubing or to fill the tubing initially with water didn't work.

However this reservoir is only used to verify that the integrated model can handle water production. The results from the simulation don't show real physical start-up behaviour. It is only used to see if the produced water by SENSOR is actually taken from OLGA. Moreover it can be tested if the used OLGA model comes up with realistic results.

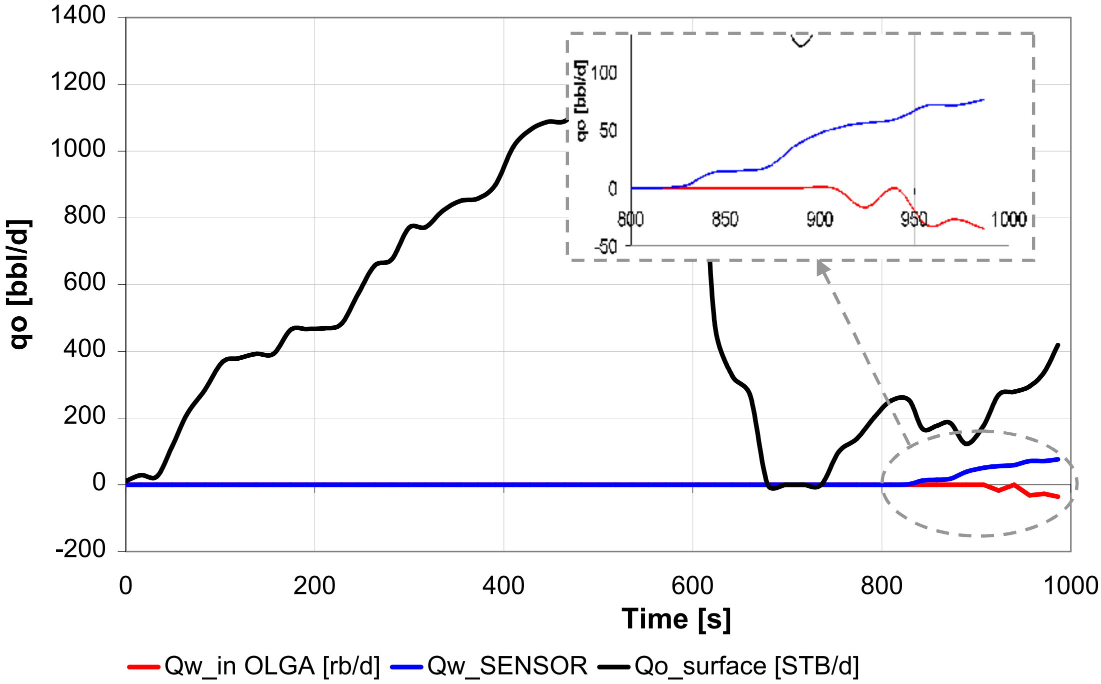


Figure 65: Flow rates with water production

It can be seen that at the end of this simulated period water passes the coupling point. This does mean that SENSOR (blue line) shows water flow-rates and OLGA takes them. It has been checked that OLGA exactly takes the flow-rates delivered by SENSOR. Due to the fact that the shown flow-rates of the OLGA model are taken from the coupling point no positive water flow-rate can be seen. This is caused by the fact that the water has to be fed into the OLGA model via a source (due to the bug described in chapter 4.1.2) which is located a couple of feet above the coupling point. This spatial distribution between the inflow node, where gas and oil are fed into the system, and the source is necessary to guarantee a stable behaviour of the model.

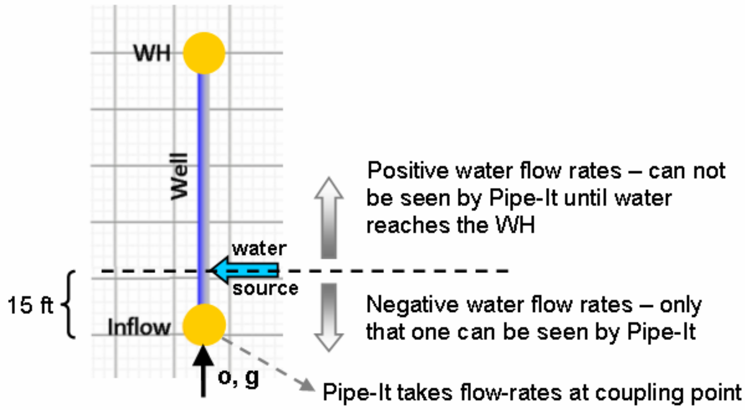


Figure 66: Positive and negative water flow-rates in the OLGA model

This spatial distribution also causes that some of the water flows down wards (shown by the red line in Figure 65). It can be seen that a certain amount of water can be handled via the source without leading to turbulences or negative flow-rates. However it is not possible to get rid of this inaccuracy until OLGA gets rid of the mentioned bug and all three phases can be introduced into the system via the inflow node. However the behaviour of incoming water was also tested in a stand-alone OLGA model. It can be stated that it shows a very good behaviour for water flow-rates that don't exceed more than 20% of the entire well stream. Flow-rates higher than that lead to inaccuracies caused by the spatial distribution of the source and the inflow node. It is not known if occurring negative flow-rates are real physical behaviour or only caused by the spatial distribution.

The questions that have been answered with this test are:

- Does OLGA take the given water flow-rates of SENSOR?
- Can OLGA handle the additional inflow of water?
- Are the detected negative water flow-rates at the coupling point injected into the reservoir model?
- Does the communication between OLGA and SENSOR work properly in the integrated model if water is introduced into the system?

All these questions can be answered with yes. This actually means that the integrated model also works when three phases are predominant.

However within the time frame of this master thesis it was not possible to set up a realistic case where during the start-up water would be produced. This does mean that the model still has to be tested with another type of reservoir to verify its functionality.

6. Conclusion

In the beginning of this Master Thesis a generic model was given that was able to simulate simplified cases for well start-up. After intensive investigation of this model some problems where encountered that had to be solved to bring the model closer to real behaviour. A lot of time was spent for these investigations to find out what has to be done to improve the existing model. The main problems that where encountered are:

1. problems with the PVT data causing inconsistent behaviour between the SENSOR and the OLGA model
2. no water production in OLGA was possible
3. no accurate simulation of well shut-in behaviour in the integrated model
4. no counter-current flow through the coupling point
5. inconsistent behaviour of the integrated model when using FTS or VTS

Those problems where caused by different sources and needed different approaches to be solved. That actually means that the model had to be changed in several ways.

ad 1)

The inconsistency in PVT data for the OLGA and SENSOR model was caused by a mistake made within the PVT calculations. This mistake led to different solutions at the coupling point for the SENSOR and OLGA model although it is supposed to be the same. The biggest change to the system was introduced due to the wrong Rs value. Subsequent the amount of predominant gas in the system increased significant and the entire start-up behaviour is changed. After redoing the PVT calculations this mistake was corrected and the inconsistencies between the models vanished. Moreover the real behaviour of the start-up for the given reservoir can be simulated now.

ad 2)

Due to a bug in OLGA the introduced water in the system dropped to zero immediately after starting the simulation. To get rid of this error the used OLGA model had to be changed. Changing the model in a way that water is now fed into the system via a source and not anymore via the inflow node led finally to a model that is able to handle water in the OLGA model.

ad 3)

Within the integrated Pipe-It model simulating shut-in conditions was done with a simplified approach that does not capture gravitational segregation. This finally leads to a wrong shut-in BHP and THP. The integrated model is now extended in a way that the simulation leads to very accurate conditions after shut-in and subsequent to the accurate predominant BHP and THP.

ad 4)

The major part of this Master Thesis actually deals with the problem of counter-current flow. It is very important for a start-up simulation to be able to solve this problem. To achieve a satisfying solution all parts of the integrated model had to be changed significantly. This does mean that now a new SENSOR, OLGA and Pipe-It model is predominant. The OLGA model uses now the dynamic solver and not anymore the steady-state pre-processor to be able to handle counter-current flow. Introducing counter-current flow into the SENSOR model made it necessary that not only production from the reservoir but also injection of oil and water into the reservoir is possible. Moreover additional processes within the Pipe-It model had to be introduced to ensure a correct communication between OLGA and SENSOR when counter-current flow occurs.

ad 5)

The current model does not show the same behaviour when using fixed time step length or variable time step length. The slope of the surface oil flow rate shows a significant different behaviour when using either FTS or VTS. Moreover the start-up times show highly different values. With the new developed model the differences between simulating with FTS and VTS for production specific parameters like flow rates and pressures vanish. This actually means that the integrated model is now highly consistent independent from the used time step method.

Bottom line can be stated that this Master Thesis shows now an integrated model that is able to capture all flow phenomena that can occur during a well start-up. Moreover the model itself shows a highly consistent behaviour.

However it also has to be pointed out that this model is still based on a very simple reservoir. Although many tests have been run to show that the model is now able to simulate a very realistic behaviour it still needs to be confirmed that the model is able to deal with more complex types of reservoirs and/or well bores.

Another issue arising with the higher complexity of the model is the needed run-time of the integrated model. For this Master thesis a standard laptop (2 GHz, 2 GB DDR memory) was used and it took between 2-3 days to run the fully coupled model for an entire start-up. When the model gets even more advanced it is highly recommended to use a machine with much higher CPU power than a standard computer can deliver.

B. References

Burke, N.E., Kashou, S.F. and Hawker, P.C. 1993. History Matching of a North Sea Flowline Startup. *Journal of Petroleum Engineers*. May 1993, S. 470-476

Coats Engineering Inc. 2009. Coats Engineering Inc. Offers SENSOR. [Online] 25 02 2010. [Cited: 25 02 2009] http://www.coatsengineering.com/.

Economides, Michael J., Hill, A. Daniel and Ehlig-Economides, Christine. 1994. *Petroleum Production Systems*. New York, USA: Prentice hall, Inc., 1994.

Golan, Michael and Whitson, Curtis Hays. 1996. *Well Performance*, SE. Trondheim, Norway: Tapir, 1996.

Petrostreamz AS. 2008. *Pipe-It Documentation*. [e-document]. Trondheim, Norway: Petrostreamz AS, 2008.

SPT Group AS. 2008. *Advanced Flow Assurance with OLGA*. Academy of Petroleum Dynamics. Kjeller, Oslo, Norway: SPT Group ASA, 2008.

SPT Group AS. 2008. *OLGA 6 Users Manual.* SPT Group AS. Kjeller, Oslo, Norway:, 2009.

Wehunt, C.D. 2006. *Well performance with operating limits under reservoir and completion uncertainties*. SPE Drilling and Completion. 04 2006, S. 200-211.

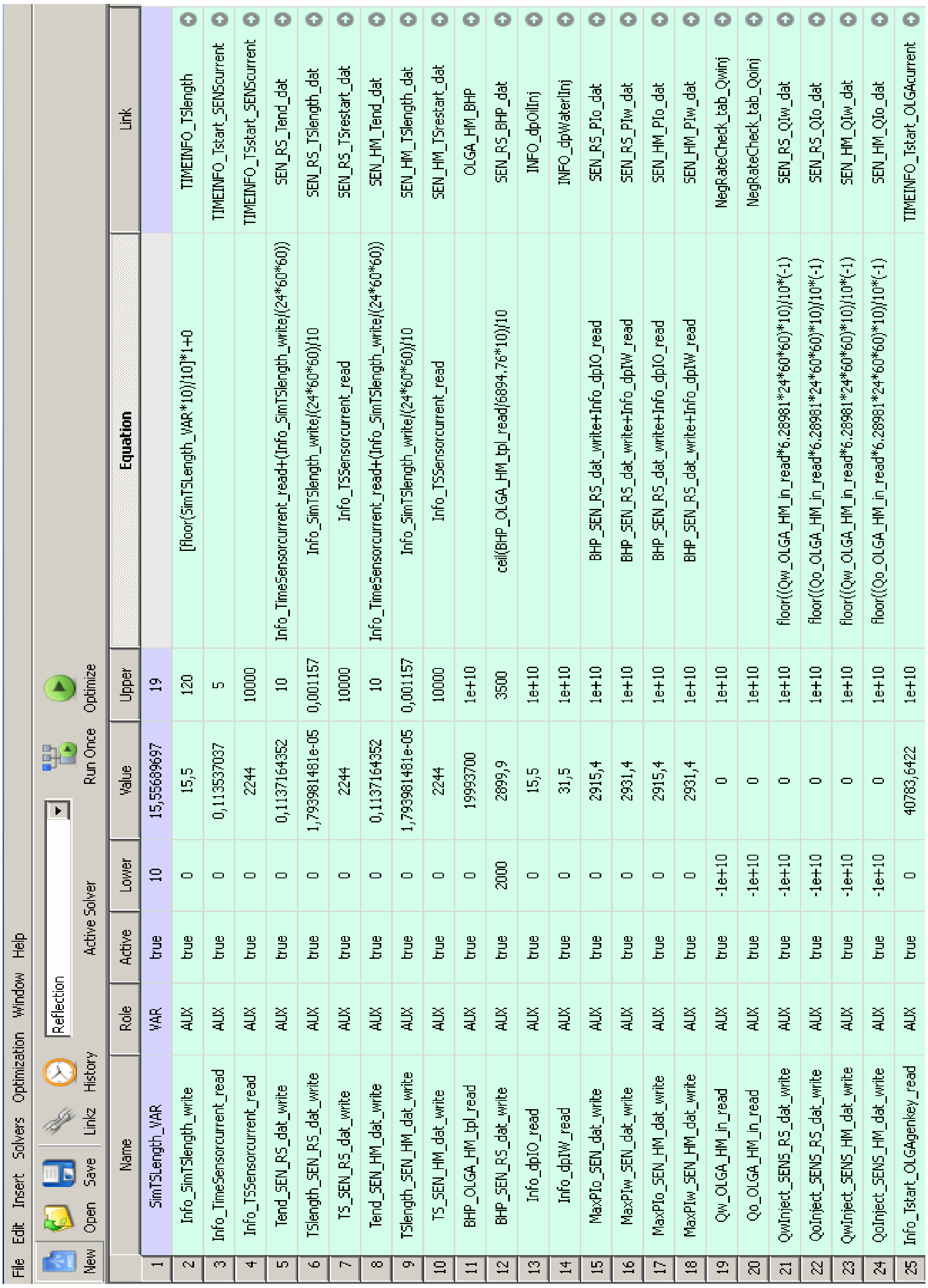
Whitson, Curtis H. and Brulé, Michael R. 2000. *Phase Behaviour*. Richardson, TS: Society of Petroleum Engineers, 2000.

Schietz, Michael. 2009. MSc Thesis. *Optimization of well start-up*. Leoben, Austria: University of Leoben, 2009.

C. Appendix

A. Detailed description of the optimizer

Restart-loop optimizer



	Name	Role	Active	Lower	Value	Upper	Equation	Link
1	SimTSLength_VAR	VAR	true	10	15,55689697	19		
2	Info_SimTSlength_write	AUX	true	0	15,5	120	[floor(SimTSLength_VAR*10)/10]*1+0	TIMEINFO_TSlength
3	Info_TimeSensorcurrent_read	AUX	true	0	0,113537037	5		TIMEINFO_Tstart_SENScurrent
4	Info_TSSensorcurrent_read	AUX	true	0	2244	10000		TIMEINFO_TSstart_SENScurrent
5	Tend_SEN_RS_dat_write	AUX	true	0	0,1137164352	10	Info_TimeSensorcurrent_read+(Info_SimTSlength_write/(24*60*60))	SEN_RS_Tend_dat
6	TSlength_SEN_RS_dat_write	AUX	true	0	1,793981481e-05	0,001157	Info_SimTSlength_write/(24*60*60)/10	SEN_RS_TSlength_dat
7	TS_SEN_RS_dat_write	AUX	true	0	2244	10000	Info_TSSensorcurrent_read	SEN_RS_TSrestart_dat
8	Tend_SEN_HM_dat_write	AUX	true	0	0,1137164352	10	Info_TimeSensorcurrent_read+(Info_SimTSlength_write/(24*60*60))	SEN_HM_Tend_dat
9	TSlength_SEN_HM_dat_write	AUX	true	0	1,793981481e-05	0,001157	Info_SimTSlength_write/(24*60*60)/10	SEN_HM_TSlength_dat
10	TS_SEN_HM_dat_write	AUX	true	0	2244	10000	Info_TSSensorcurrent_read	SEN_HM_TSrestart_dat
11	BHP_OLGA_HM_tpl_read	AUX	true	0	19993700	1e+10		OLGA_HM_BHP
12	BHP_SEN_RS_dat_write	AUX	true	2000	2899,9	3500	ceil(BHP_OLGA_HM_tpl_read/6894.76*10)/10	SEN_RS_BHP_dat
13	Info_dpIO_read	AUX	true	0	15,5	1e+10		INFO_dpOilInj
14	Info_dpIW_read	AUX	true	0	31,5	1e+10		INFO_dpWaterInj
15	MaxPIo_SEN_dat_write	AUX	true	0	2915,4	1e+10	BHP_SEN_RS_dat_write+Info_dpIO_read	SEN_RS_PIo_dat
16	MaxPIw_SEN_dat_write	AUX	true	0	2931,4	1e+10	BHP_SEN_RS_dat_write+Info_dpIW_read	SEN_RS_PIw_dat
17	MaxPIo_SEN_HM_dat_write	AUX	true	0	2915,4	1e+10	BHP_SEN_RS_dat_write+Info_dpIO_read	SEN_HM_PIo_dat
18	MaxPIw_SEN_HM_dat_write	AUX	true	0	2931,4	1e+10	BHP_SEN_RS_dat_write+Info_dpIW_read	SEN_HM_PIw_dat
19	Qw_OLGA_HM_in_read	AUX	true	-1e+10	0	1e+10		NegRateCheck_tab_Qwinj
20	Qo_OLGA_HM_in_read	AUX	true	-1e+10	0	1e+10		NegRateCheck_tab_Qoinj
21	QwInject_SENS_RS_dat_write	AUX	true	-1e+10	0	1e+10	floor((Qw_OLGA_HM_in_read*6.28981*24*60*60)*10)/10*(-1)	SEN_RS_QIw_dat
22	QoInject_SENS_RS_dat_write	AUX	true	-1e+10	0	1e+10	floor((Qo_OLGA_HM_in_read*6.28981*24*60*60)*10)/10*(-1)	SEN_RS_QIo_dat
23	QwInject_SENS_HM_dat_write	AUX	true	-1e+10	0	1e+10	floor((Qw_OLGA_HM_in_read*6.28981*24*60*60)*10)/10*(-1)	SEN_HM_QIw_dat
24	QoInject_SENS_HM_dat_write	AUX	true	-1e+10	0	1e+10	floor((Qo_OLGA_HM_in_read*6.28981*24*60*60)*10)/10*(-1)	SEN_HM_QIo_dat
25	Info_Tstart_OLGAgenkey_read	AUX	true	0	40783,6422	1e+10		TIMEINFO_Tstart_OLGAcurrent

Figure 67: Restart optimizer 1/2

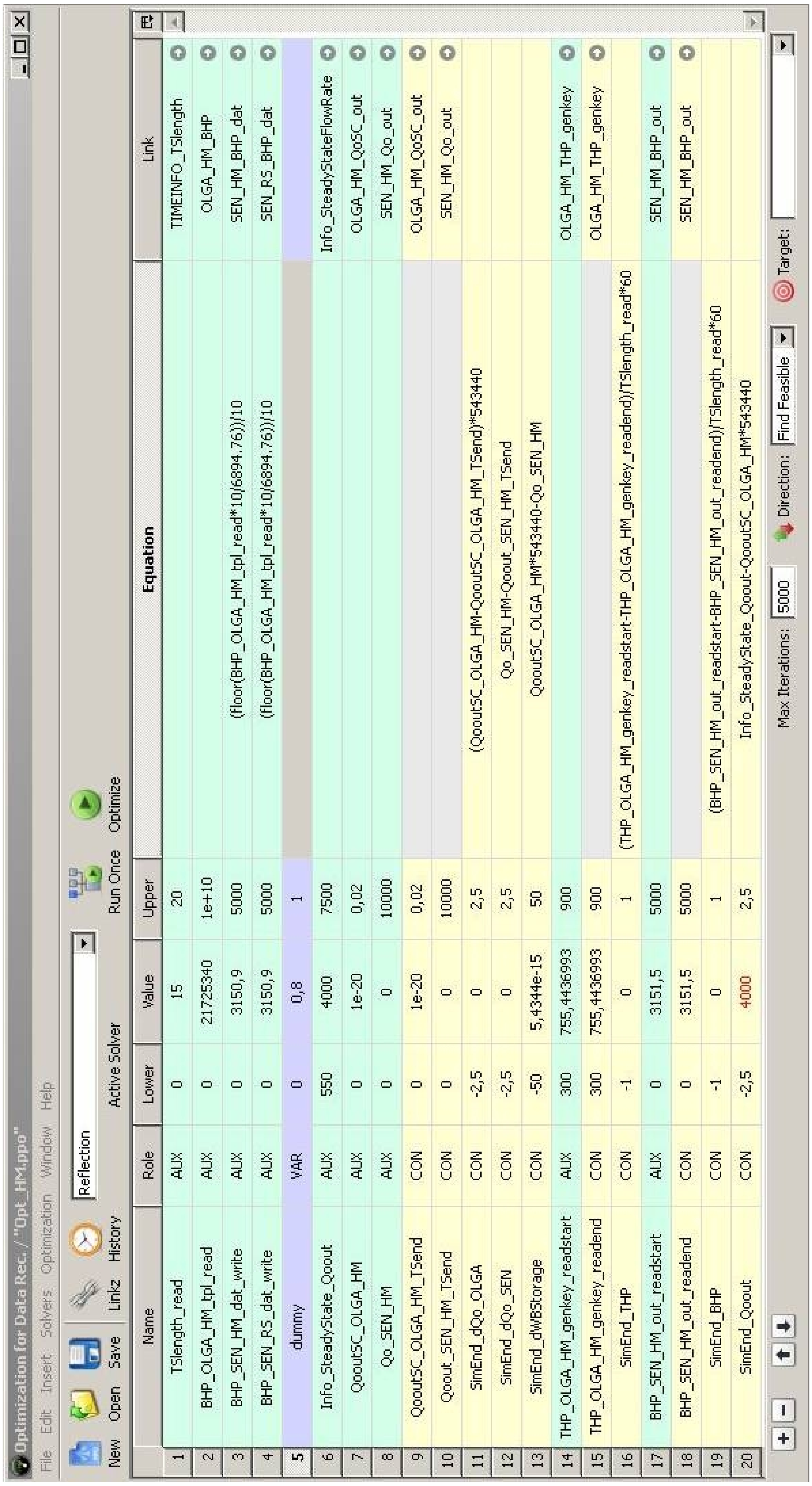
26	Tstart_OLGAgenkey_RS_write	AUX	true	0	40783,6422	1e+10	Info_Tstart_OLGAgenkey_read	OLGA_RS_Tstart_genkey
27	Tend_OLGAgenkey_RS_write	AUX	true	0	40799,1422	1e+10	Info_Tstart_OLGAgenkey_read+Info_SimTSlength_write	OLGA_RS_Tend_genkey
28	Trestart_OLGAgenkey_RS_write	AUX	true	0	40783,6422	1e+10	Info_Tstart_OLGAgenkey_read	OLGA_RS_Trestart_genkey
29	Tstart_OLGAgenkey_HM_write	AUX	true	0	40783,6422	1e+10	Info_Tstart_OLGAgenkey_read	OLGA_HM_Tstart_genkey
30	Tend_OLGAgenkey_HM_write	AUX	true	0	40799,1422	1e+10	Info_Tstart_OLGAgenkey_read+Info_SimTSlength_write	OLGA_HM_Tend_genkey
31	Trestart_OLGAgenkey_HM_write	AUX	true	0	40783,6422	1e+10	Info_Tstart_OLGAgenkey_read	OLGA_HM_Trestart_genkey
32	THP_OLGA_HM_read	AUX	true	0	795,0250737	1e+10		OLGA_HM_THP_genkey
33	dTHP_OLGA_genkey	VAR	true	-50	15,01708984	50		
34	dTHP_OLGA_genkey_CON	CON	true	-20	15,01708984	20	dTHP_OLGA_genkey	
35	THP_OLGA_RS_genkey	AUX	true	500	798,9044886	2000	THP_OLGA_HM_read+[dTHP_OLGA_genkey]/60*Info_SimTSlength_write	OLGA_RS_THP_genkey
36	THP_OLGA_RS_CON_read	CON	true	3447380	5312196	6205280		OLGA_RS_THP
37	Qgout_OLGA_RS_read	CON	true	0	3,43923e-06	60		OLGA_RS_QgSC_out
38	Qoout_OLGA_RS_read	CON	true	0	1e-20	0,03		OLGA_RS_QoSC_out
39	Qwout_OLGA_RS_read	CON	true	0	1e-20	0,03		OLGA_RS_QwSC_out
40	Info_SteadyState_Qoout_read	AUX	true	0	2000	10000		Info_SteadyStateFlowRate
41	Qoout_OLGA_RS_read_CON	CON	true	0	2000000	10000000	[Info_SteadyState_Qoout_read-Qoout_OLGA_RS_read*543440]*1000	
42	BHP_OLGA_RS_read_tpl	CON	true	0	21834280	22000000		OLGA_RS_BHP
43	BHP_OLGA_RS_CON	CON	true	0	3166793,333	3166700	[BHP_OLGA_RS_read_tpl/6894.76]*1000	
44	dBHP_OLGA_RS_CON	CON	true	-2000	103336,8211	2000	[((BHP_OLGA_RS_read_tpl-BHP_OLGA_HM_tpl_read)/6894.76)/Info_SimTSlength_write*60]*100	
45	Qo_SEN_HM_out_read	AUX	true	0	643	2000		SEN_HM_Qo_out
46	Qo_SEN_RS_out_read	CON	true	100	647	2000		SEN_RS_Qo_out
47	dQo_SEN_RS_out_CON	CON	true	-10000	15483,87097	200000	[(Qo_SEN_RS_out_read-Qo_SEN_HM_out_read)/Info_SimTSlength_write*60]*1000	
48	dQo_SENout_to_steadystate_...	CON	true	0	1353000	10000000	(Info_SteadyState_Qoout_read-Qo_SEN_RS_out_read)*1000	
49	Qoout_OLGA_HM	AUX	true	0	0,003665955	1		OLGA_HM_QoSC_out
50	dQoout_OLGA_RS_CON	CON	true	1000	-771184,4846	40000	[(Qoout_OLGA_RS_read-Qoout_OLGA_HM)*543440/Info_SimTSlength_write*60]*100	
51	Info_RestartIterations_read	AUX	true	0	14	40		TIMEINFO_RestartIt
52	Info_RestartIterations_write	AUX	true	0	15	40	Info_RestartIterations_read+1	TIMEINFO_RestartIt

Max Iterations: 20 Direction: Find Feasible Target:

Figure 68: Restart optimizer 2/2

Line	Description
1	Control variable for choosing time-step-length
2	Writes chosen time-step-length to TIMEINFO file
3-4	loads current time-step and current time of Sensor
5-7	Writes end-time, time-step-length and restart time-step to Sensor restart file in Restart Loop
8-10	Writes end-time, time-step-length and restart time-step to Sensor restart file in Head Master Loop
11	Loads BHP from last Head Master time-step
12	converts BHP from Pa into psia and writes it to Sensor restart file in Restart Loop
13-14	loads pressure differences for the oil/water injector compared to BHP
15-18	writes maximum oil/water injection pressures to Sensor restart file in Restart Loop and Head Master
19-20	reads oil/water flow rates @ bottom from last OLGA Head Master run
21-24	writes oil/water injection rates into Sensor restart file for Restart Loop and Head Master if predominant, otherwise set to zero
25-31	reads and writes start-, end- and restart-time info for OLGA .genkey file in Restart Loop and Head Master
32	reads current THP of the last Head Master time-step from OLGA .genkey
33-34	Defines change in THP for next time-step
35	Writes new THP to OLGA .genkey file in Restart Loop
36	Loads OLGA THP after simulation run
37-39	reads OLGA oil, gas, water flow-rates @ top after simulation run
40	loads desired final steady-state flow-rate
41	looks if flow rate has already reached desired one
42	loads OLGA BHP after simulation run
43	constraints OLGA BHP to stay below shut-in BHP
44	constraints the change in BHP from one time-step to the next one to ± 20 psia/min
45-47	reads Sensor oil flow-rates before and after the simulation run and compares them
48	compares current SENSOR oil flow-rate with desired steady-state flow-rate
49	reads current oil flow-rate from OLGA Head Master .tpl file
50	looks if OLGA oil flow-rate changes from the last TS to the current
51-52	keeps track of the needed restart iterations to find the optimal THP for the next TS

Headmaster optimizer



	Name	Role	Lower	Value	Upper	Equation	Link
1	TSlength_read	AUX	0	15	20		TIMEINFO_TSlength
2	BHP_OLGA_HM_tpl_read	AUX	0	21725340	1e+10		OLGA_HM_BHP
3	BHP_SEN_HM_dat_write	AUX	0	3150,9	5000	(floor(BHP_OLGA_HM_tpl_read*10/6894.76))/10	SEN_HM_BHP_dat
4	BHP_SEN_RS_dat_write	AUX	0	3150,9	5000	(floor(BHP_OLGA_HM_tpl_read*10/6894.76))/10	SEN_RS_BHP_dat
5	dummy	VAR	0	0,8	1		
6	Info_SteadyState_Qoout	AUX	550	4000	7500		Info_SteadyStateFlowRate
7	QooutSC_OLGA_HM	AUX	0	1e-20	0,02		OLGA_HM_QoSC_out
8	Qo_SEN_HM	AUX	0	0	10000		SEN_HM_Qo_out
9	QooutSC_OLGA_HM_TSend	CON	0	1e-20	0,02		OLGA_HM_QoSC_out
10	Qoout_SEN_HM_TSend	CON	0	0	10000		SEN_HM_Qo_out
11	SimEnd_dQo_OLGA	CON	-2,5	0	2,5	(QooutSC_OLGA_HM-QooutSC_OLGA_HM_TSend)*543440	
12	SimEnd_dQo_SEN	CON	-2,5	0	2,5	Qo_SEN_HM-Qoout_SEN_HM_TSend	
13	SimEnd_dWBStorage	CON	-50	5,4344e-15	50	QooutSC_OLGA_HM*543440-Qo_SEN_HM	
14	THP_OLGA_HM_genkey_readstart	AUX	300	755,4436993	900		OLGA_HM_THP_genkey
15	THP_OLGA_HM_genkey_readend	CON	300	755,4436993	900		OLGA_HM_THP_genkey
16	SimEnd_THP	CON	-1	0	1	(THP_OLGA_HM_genkey_readstart-THP_OLGA_HM_genkey_readend)/TSlength_read*60	
17	BHP_SEN_HM_out_readstart	AUX	0	3151,5	5000		SEN_HM_BHP_out
18	BHP_SEN_HM_out_readend	CON	0	3151,5	5000		SEN_HM_BHP_out
19	SimEnd_BHP	CON	-1	0	1	(BHP_SEN_HM_out_readstart-BHP_SEN_HM_out_readend)/TSlength_read*60	
20	SimEnd_Qoout	CON	-2,5	4000	2,5	Info_SteadyState_Qoout-QooutSC_OLGA_HM*543440	

Figure 69: Headmaster optimizer

Line	Description
1	reads current time-step-length from TIMEINFO file
2	reads current BHP from last Head Master time-step from OLGA .tpl file
3-4	writes current BHP into Sensor input files for the Head Master and Restart Loop
5	dummy variable, needed that the optimizer works but doesn't have any impact
6	reads desired steady-state oil flow-rate from Info file
7	reads OLGA oil flow-rate @ surface of the last time-step
8	reads SENSOR oil flow-rate @ surface of the last time-step
9	reads OLGA oil flow-rate @ surface of the current time-step
10	reads SENSOR oil flow-rate @ surface of the current time-step
11-12	looks if oil flow-rates in Sensor or OLGA change from one time-step to the next one
13	looks if well-bore storage effect is still predominant or not
14	reads OLGA THP of last time-step
15	reads OLGA THP of current time-step
16	looks if there is a change in THP from one time-step to the next one
17	reads Sensor BHP of last time-step
18	reads Sensor BHP of current time-step
19	looks if there is a change in BHP from one time-step to the next one
20	compares current OLGA oil outflow rate with desired steady-state flow-rate

B. OLGA input file for restart loop

```
!*********************************************************************************
!      CASE
!*********************************************************************************
CASE AUTHOR="JMS", DATE="09.11.2009", PROJECT="MSc Thesis",
TITLE="Single Well S.U."
INTEGRATION ENDTIME=40017.5697 s, MAXDT=10 s, MINDT=1e-008 s,
STARTTIME=39999.5697 s, DTSTART=0.001 s
FILES PVTFILE="./OLGA_FluidTable_extended.tab"
OPTIONS TEMPERATURE=FASTWALL, STEADYSTATE=OFF,
FLASHMODEL=HYDROCARBON, DEBUG=OFF
OUTPUT DTOUT=500 h
TREND DTPLOT=500 h
PROFILE DETAILEDPROFILEPLOT=OFF, DTPLOT=500 h
RESTART WRITE=OVERWRITE, WRITEFORMAT=BINARY,
READFILE=ON, FILE="./OLGA_HM.rsw", READTIME=39999.5697 s
```

General Information to the case and integration

```
!*********************************************************************************
!      LIBRARY
!*********************************************************************************
MATERIAL LABEL="Annulus", CAPACITY=4180 J/kg-C, CONDUCTIVITY=0.58 W/m-C,
DENSITY=900 kg/m3,\
  TYPE=FLUID, EXPANSION=2.46e-005 1/C, VISCOSITY=1.04 CP
MATERIAL LABEL="Formation", CAPACITY=2000 J/kg-C, CONDUCTIVITY=2 W/m-C,
DENSITY=2500 kg/m3
MATERIAL LABEL="Tubing", CAPACITY=500 J/kg-C,
CONDUCTIVITY=50 W/m-C, DENSITY=7850 kg/m3
MATERIAL LABEL="Casing", CAPACITY=500 J/kg-C,
CONDUCTIVITY=50 W/m-C, DENSITY=7850 kg/m3
WALL LABEL="WALL_Tubing", THICKNESS= (0.3, 0.8, 1, 0.4, 1, 1.5, 2.1, 3, 4.3) in,
MATERIAL=("Tubing",\ "Annulus", "Annulus", "Casing", "Formation",
"Formation", "Formation", "Formation",\ "Formation")
TABLE LABEL="LEAK", POINT=(0, 30.34152, 1000, 30.34152, 10000, 30.34152),
LOOKUP=OFF,\ XVARIABLE=DELTAP, YVARIABLE=FLOW kg/h

!*********************************************************************************
!      FLOWPATH
!*********************************************************************************
NETWORKCOMPONENT TYPE=FLOWPATH, TAG=FLOWPATH_0
 PARAMETERS LABEL="Well"
 BRANCH GEOMETRY="Flow28", INIFLOWDIR=POSITIVE, FLUID="1"
 GEOMETRY XSTART=0 M, YSTART=-2072.64 M, ZSTART=0 M,
LABEL="Flow28"
PIPE ROUGHNESS=2.5E-05 M, LABEL="Well", WALL="WALL_Tubing",
NSEGMENT=28, LSEGMENT=(0.03048,\
3.01752, 4.42265, 6.41299, 9.3025, 13.4935, 19.5712, 28.3891,
41.1785, 59.7286, 86.6394, 125.672, 182.289, 264.411, 383.524, 264.411,
182.289, 125.672, 86.6394, 59.7286, 41.1785, 28.3891, 19.5712, 13.4935,
9.3025, 6.41299, 4.42265, 3.04712) M, XEND=0 M, YEND=0 M,
DIAMETER=0.0889 M
HEATTRANSFER PIPE="Well", INTERPOLATION=VERTICAL,
HOUTEROPTION=WATER, INTAMBIENT=75 C, OUTTAMBIENT=15 C
SOURCE LABEL="SOUR-1", ABSPOSITION=5 m, TEMPERATURE=75 C,
TIME=0 s, PHASE=OIL, STDFLOWRATE=0 STB/d, WATERCUT=99.9 %
```

Definition of the flow path

```
OUTPUTDATA VARIABLE=(GG, GL, PT, ROG, ROHL, ROL, ROWT,
STDROG, STDROHL, STDROWT, VISG, VISHLTAB, VISWTTAB)
 TRENDDATA VARIABLE=(ACCGAQ, ACCOIQ, ACCWAQ, GORST,
```

```
PT, QG, QGST, QLTHL, QLTWT, QOST, QWST, WCST), PIPE="Well",
SECTION=(1, 28)
PROFILEDATA VARIABLE=(ACCLIQ, ACCLIQBR, ACCOILBR, ACCOIQ,
ACCWAQ, ACCWATBR, AL, HOL, HOLHL, HOLWT, PT, QG, QLT, QLTHL,
QLTWT, TM)
ENDNETWORKCOMPONENT

!*******************************************************************************
!     NODE
!*******************************************************************************
NETWORKCOMPONENT TYPE=NODE, TAG=NODE_0
PARAMETERS LABEL="Inflow", TYPE=MASSFLOW, Y=-6800 m, FLUID="1",
STDFLOWRATE=0 STB/d, PHASE=OIL, GOR=0 scf/STB, WATERCUT=0,
TEMPERATURE=75 C, PRESSURE=3151.4 psia, TIME=0 s
ENDNETWORKCOMPONENT
```

Definition of the Inflow node

```
NETWORKCOMPONENT TYPE=NODE, TAG=NODE_1
 PARAMETERS LABEL="WH", TYPE=PRESSURE, FLUID="1", GOR=-1 scf/STB,
TEMPERATURE=75 C, PRESSURE=846.369226191206 psia
ENDNETWORKCOMPONENT
```

Definition of the WH node

```
!*******************************************************************************
!     CONNECTIONS
!*******************************************************************************
CONNECTION TERMINALS = (NODE_0 FLOWTERM_1, FLOWPATH_0 INLET)
CONNECTION TERMINALS = (NODE_1 FLOWTERM_1, FLOWPATH_0 OUTLET)

ENDCASE
```

OLGA input file for shut in simulation

Only the changed sections in comparison to the above shown file are printed for the shut in OLGA file.

```
!*******************************************************************************
!     CASE
!*******************************************************************************
CASE AUTHOR="JMS", DATE="09.11.2009", PROJECT="MSc Thesis",
TITLE="Single Well S.U." INTEGRATION ENDTIME=10 h, MAXDT=10 s,
MINDT=1e-006 s, STARTTIME=0 s, DTSTART=0.001 s
FILES PVTFILE="./OLGA_FluidTable_extended.tab"
OPTIONS TEMPERATURE=FASTWALL, STEADYSTATE=OFF,
FLASHMODEL=HYDROCARBON, DEBUG=OFF
OUTPUT DTOUT=15 h
TREND DTPLOT=15 h
PROFILE DETAILEDPROFILEPLOT=OFF, DTPLOT=15 h
RESTART WRITE=OVERWRITE, WRITEFORMAT=BINARY, READFILE=OFF

!*******************************************************************************
!     NODE
!*******************************************************************************
NETWORKCOMPONENT TYPE=NODE, TAG=NODE_0
 PARAMETERS LABEL="Inflow", TYPE=PRESSURE, Y=-6800 m, FLUID="1", GOR=0 scf/STB,
WATERCUT=0, TEMPERATURE=75 C, PRESSURE=3151.5 psia, TIME=0 s
ENDNETWORKCOMPONENT
NETWORKCOMPONENT TYPE=NODE, TAG=NODE_1
 PARAMETERS LABEL="WH", TYPE=CLOSED
ENDNETWORKCOMPONENT
```

C. SENSOR input file / model reaches till surface

```
TITLE
--------------------------------------------------------------------------------------
            Radial 3-Phase Model
            2 Reservoir Zones (Tilje & Aare) are devided into 22 and 29 Layers.
            Tbg entirely modeled with SENSOR.

            Radial Gridding is used: 28 radial   1 angular   206 in z-direction
--------------------------------------------------------------------------------------

ENDTITLE

GRID   28  1  206
RUN
CPU
IMPLICIT

C Radial grid geometric using 1.4 as multiplier to get block boundary radii

RADIAL
3                        ! read nx rbi
0   1315 ! Tbg_ID = 3 1/2 in

C rbi:
0.0  0.145833334  0.204166667  0.285833334  0.400166667  0.560233334  0.784326668
1.098057335  1.537280268  2.152192376  3.013069326  4.218297057  5.905615879  8.267862231
11.57500712  16.20500997  22.68701396  31.76181955  44.46654736  62.25316631  87.15443283
122.0162060  170.8226884  239.1517637  334.8124692  468.7374568  656.2324396  918.7254154

360                          ! dely(j) = deltheta [degrees]

C --------------------------------------------------------------------------------------
C PVT DATA
C --------------------------------------------------------------------------------------

C Bw = Bwi(1-cw(p-pref))      cw = water compressibility
C denw = water density at std T&P (sp. gr. or lb/ft3)
C dw = (denw/Bwi)(1+cw(p-pref)) lb/ft3
C PV = PVbase(1+cf(p-pbase))  entered pv (or porosity) PVbase            pbase are block PVinit
C pinit unless POROSBASE pbase is entered

C      Bwi     cw        denw     visw   cf        pref
C      rb/stb   1/psi      lb/ft3   cp      1/psi      psia
MISC   1.03     3.406E-6   62.3096   0.38   4.702E-6   3709.  ! -> Source: Geir Frode Kvilaas

PVTBO         ! NOPRINT

C     deno     deng         coil      ! cvoil(= oil- & gas surface density, oil compressib., oil viscosity C coeff.)
C     lb/ft3    lb/ft3
DENSITY   52.8255   0.0904694     6.34E-06   ! 9.02E-5
C values are taken from PVT data used in OLGA

PRESSURES   4   8   PSIG
 0.   710.66   1725.89   2741.12   3756.35   4771.57   5786.80   6802.03
```

```
C SATURATED PVTBO Table
C       psig        rb/stb       rb/scf       scf/stb       cp      cp
        PSAT        BO           BG           RS            VISO    VISG
        2741.12     1.44221162   0.00170934   1136.93425    0.1290  0.0196
        1725.89     1.26199129   0.0028025    657.71436     0.1330  0.0159
        710.66      1.10196071   0.00699485   234.60582     0.1380  0.0135
        0           1.02443933   0.19854607   0.000000      0.1380  0.0135


C UNDERSATURATED PVTBO Table
C       psig        rb/scf       cp
        P           BG           VISG
        6802.03     0.00042804   0.039
        5786.80     0.00051829   0.036
        4771.57     0.00084905   0.0361
        3756.35     0.0012175    0.025

PRINTKR 0

C -----------------------------------------------------------------------------------
C ROCK TYPES: 1 = tubing ; 2 = reservoir // Grid Block Properties are entered as Arrays!
C
C                                         // MOD sets rocktype for layers i to other than
C                                         // the type defined globally with CON
C -----------------------------------------------------------------------------------

ROCKTYPE CON
1
MOD
 2 28 1 1  142 206 = 2

C -----------------------------------------------------------------------------------
C DEFINITION OF REGIONS: 1 = tubing ; 2 = Tilje ; 3 = Aare
C -----------------------------------------------------------------------------------

REGION CON
1
MOD
 2 28 1 1 142  170 = 2
 2 28 1 1 171  204 = 3

C -----------------------------------------------------------------------------------
C Tubing "relative permeability"
C -----------------------------------------------------------------------------------

KRANALYTICAL 1
 0.0  0.0  0.0  0.0    ! Swc      Sorw     Sorg     Sgc
 1.0  1.0  1.0         ! krw(Sorw)  krg(Swc)  kro(Swc)
 1.0  1.0  1.0  1.0    ! nw       now      ng       nog

C -----------------------------------------------------------------------------------
C ROCK                                                   <- Source: Zein Wijaya, 2006
C       (-> exponents obtained by trial an error to fit rel-perm curves in source paper)
C -----------------------------------------------------------------------------------

KRANALYTICAL 2
 0.18  0.02  0.2  0.05     ! Swc      Sorw     Sorg     Sgc
 0.98  0.75  1.0           ! krw(Sorw)  krg(Swc)  kro(Swc)
 5.0   4.0   1.8  2.8      ! nw       now      ng       nog

KRANALYTICAL 3
```

```
 0.18  0.02  0.2  0.05
 0.98  0.75  1.0
 5.0   4.0   1.8  2.8




C ---------------------------------------------------------------------------------
C PERMEABILITY                    <- values obtained or estimated from SPOR Monograph
C                                 (used 10 times the k_min for each layer)
C ---------------------------------------------------------------------------------

KX CON                   ! fitted by: KX = 2,211211E+12 * Q_liq^(- 1,023924)

1900000000               ! match to Olga Tubing dp for 0 - 1000 STBLIQ

MOD
 2 28 1 1 142  170 = 300          ! Tilje
 2 28 1 1 171  206 = 400          ! Aare

KY EQUALS KX

KZ EQUALS KX
MOD
 2 28 1 1 142  169 = 100          ! Tilje   (k_v/k_h = 0.5)
 2 28 1 1 171  206 =  50                  ! Aare   (k_v/k_h = 0.1)

 2 28 1 1 170 170 =   0           ! make layers non-communicating (KZ = 0 is tighter than TZ = 0!!)

TRMINUS

C ---------------------------------------------------------------------------------
C POROSITY                        <- values obtained or estimated from SPOR Monograph
C ---------------------------------------------------------------------------------

POROS CON
1
MOD
 2 28 1 1 142  170 = 0.25         ! Tilje
 2 28 1 1 171  206 = 0.27         ! Aare
 2 28 1 1  1   141 = 0.0          ! deactivate outside-tubing cells!

C ---------------------------------------------------------------------------------
C DEPTH
C ---------------------------------------------------------------------------------

DEPTH CON
0

THICKNESS ZVAR
 10  20  30  40
 134*50
 0.1  0.1
 89.8                    ! SENSOR-Tbg part reaches till surface
 20*5.0    9*40.0        ! Fmt gross thicknesses: Tilje = 460 ft (29 cells) and
 22*5.0    14*40.0       !                                Aare  = 670 ft (36 cells)

C ---------------------------------------------------------------------------------
C Assigning grid blocks to the 3 initialization regions; 1 = tubing ; 2 = Tilje ; 3 = Aare
C ---------------------------------------------------------------------------------
```

```
INITREG CON
 1
MOD
 2 28 1 1 142  170 = 2
 2 28 1 1 171  206 = 3

INITIAL 1
 DEPTH PSATBP
 6890     2000

  ZINIT   8020.    ! [ft]
  PINIT   3600.    ! [psia]
C  GOC 600.
C  HWC 6800.    ! [ft]

INITIAL 2
 DEPTH PSATBP
 6890     2000

  ZINIT   8020.    ! [ft]
  PINIT   3600.    ! [psia]
C  GOC 6890.
C   HWC            7300.     ! [ft]

INITIAL 3
 DEPTH PSATBP
 7350     2000

 ZINIT    8020.    ! [ft]
 PINIT    3600.    ! [psia]
C GOC  7350
C  HWC 7970.    ! [ft]

ENDINIT

MODIFY TX
 2  2   1 1   146 170 = 0              ! limits perforated interval to
 2  2   1 1   175 206 = 0              ! 20 ft at the top of each interval

 2  3   1 1   142 145 = .2             ! limits T_perf
 2  3   1 1   171 174 = .2

C -------------------------------------------------------------------------------------
C Define Well A1
C -------------------------------------------------------------------------------------

WELL
          I  J  K   RW        PI
  A1      1  1  1  .145833334   1000

WELLTYPE

  A1  STBLIQ                ! producer in stboil/d

WINDOWS
    1  1 1    1 1    1 206             XZY  ! Surface
C   2  1 1    1 1    139 139           XZY  ! Coupling Point
C   3  1 1    1 1    1 206             XZY
C   4  1 1    1 1    142 142           XZY  ! Top of Tilje
C   5  1 1    1 1    171 171           XZY  ! Top of Aare
```

```
 MAPSPRINT 1    P  ! SO  SG  SW

MAPSFILEFREQ  -1                  !          -> Mapsfile Printout
MAPSFREQ      0                   !          -> Maps Printout: Print only at times entered
STEPFREQ      1                   !          -> Timestep Table
WELLFREQ      1                   !          -> Well Table
SUMFREQ      -1                   !          -> EndOfRun Summary (every timestep)
PRINTZERO    -1                   !          -> To include zero rate lines
PRINTREG     -1                   !          -> Region table printout (insert # of regions to be plotted)

C --------------------------------------------------------------------------------
C Pressure/Rate Schedule
C --------------------------------------------------------------------------------

RATE
 A1  0 STBOIL

DT 0.0000001

TIME   1    .1

END
```

D. SENSOR input file / reservoir only

```
TITLE

------------------------------------------------------------------------------------
            Radial 3-Phase Model

            2 Reservoir Zones (Tilje & Aare) are devided into 22 and 29 Layers.
            Tbg in the Reservoir section + 10 Gridcells is modeled with SENSOR.
            The Pressure at this point is determined by OLGA and returned after each step:
            SENSOR then delivers the phase rates accordingly.

            Radial Gridding is used: 28 radial   1 angular   77 in z-direction
------------------------------------------------------------------------------------

ENDTITLE

GRID   28  1  75
RUN
CPU
IMPLICIT

C Radial grid geometric using 1.4 as multiplier to get block boundary radii

RADIAL
3                       ! read nx rbi
0   1315 ! Tbg_ID = 3 1/2 in

C rbi:
0.0  0.145833334  0.204166667  0.285833334  0.400166667  0.560233334  0.784326668
1.098057335  1.537280268  2.152192376  3.013069326  4.218297057  5.905615879  8.267862231
11.57500712  16.20500997  22.68701396  31.76181955  44.46654736  62.25316631  87.15443283
122.0162060  170.8226884  239.1517637  334.8124692  468.7374568  656.2324396  918.7254154

360                        ! dely(j) = deltheta [degrees]


C ----------------------------------------------------------------------------------
C PVT DATA
C ----------------------------------------------------------------------------------

C Bw = Bwi(1-cw(p-pref))      cw = water compressibility
C denw = water density at std T&P (sp. gr. or lb/ft3)
C dw = (denw/Bwi)(1+cw(p-pref)) lb/ft3
C PV = PVbase(1+cf(p-pbase))  entered pv (or porosity) PVbase            pbase are block PVinit
C pinit unless POROSBASE pbase is entered

C       Bwi     cw            denw      visw   cf        pref
C       rb/stb   1/psi          lb/ft3     cp     1/psi      psia
MISC   1.03     3.406E-6   62.3096   0.38   4.702E-6   3709.              ! -> Source: Geir Frode Kvilaas


PVTBO         ! NOPRINT
C  deno      deng         coil      ! cvoil (= oil- & gas surface density, oil compressib., oil viscosity coeff.)
C        lb/ft3   lb/ft3
DENSITY  52.8255   0.0904694     6.34E-06   ! 9.02E-5     ! values are taken from PVT data used in
OLGA
```

```
PRESSURES   4   8   PSIG
 0.   710.66   1725.89   2741.12   3756.35   4771.57   5786.80   6802.03

C SATURATED PVTBO Table
C        psig         rb/stb        rb/scf        scf/stb       cp       cp
         PSAT         BO            BG            RS            VISO     VISG
         2741.12      1.44221162    0.00170934    1136.93425    0.1290   0.0196
         1725.89      1.26199129    0.0028025     657.71436     0.1330   0.0159
         710.66       1.10196071    0.00699485    234.60582     0.1380   0.0135
         0            1.02443933    0.19854607    0.000000      0.1380   0.0135

C UNDERSATURATED PVTBO Table
C        psig         rb/scf        cp
         P            BG            VISG
         6802.03      0.00042804    0.039
         5786.80      0.00051829    0.036
         4771.57      0.00084905    0.0361
         3756.35      0.0012175     0.025

PRINTKR 0

C -------------------------------------------------------------------------------------
C ROCK TYPES: 1 = tubing ; 2 = reservoir // Grid Block Properties are entered as Arrays!
C
C                                                  // MOD sets rocktype for layers i to other than
C                                                  // the type defined globally with CON
C -------------------------------------------------------------------------------------

ROCKTYPE CON
1
MOD
 2 28 1 1  11 68 = 2

C -------------------------------------------------------------------------------------
C DEFINITION OF REGIONS: 1 = tubing ; 2 = Tilje ; 3 = Aare
C -------------------------------------------------------------------------------------

REGION CON
1
MOD
 2 28 1 1 11 39 = 2
 2 28 1 1 40 75 = 3

C -------------------------------------------------------------------------------------
C Tubing "relative permeability"
C -------------------------------------------------------------------------------------

KRANALYTICAL 1
 0.0  0.0  0.0  0.0    ! Swc       Sorw      Sorg      Sgc
 1.0  1.0  1.0        ! krw(Sorw)  krg(Swc)  kro(Swc)
 1.0  1.0  1.0  1.0    ! nw        now       ng        nog

C -------------------------------------------------------------------------------------
C ROCK                                                                  <- Source: Zein Wijaya, 2006
C        (-> exponents obtained by trial an error to fit rel-perm curves in source paper)
C -------------------------------------------------------------------------------------

KRANALYTICAL 2
 0.18  0.02  0.2  0.05  ! Swc       Sorw      Sorg      Sgc
 0.98  0.75  1.0        ! krw(Sorw)  krg(Swc)  kro(Swc)
 5.0   4.0   1.8  2.8   ! nw        now       ng        nog
```

```
KRANALYTICAL 3
 0.18  0.02  0.2  0.05
 0.98  0.75  1.0
 5.0   4.0   1.8  2.8

C ------------------------------------------------------------------------------------
C PERMEABILITY                                  <- values obtained or estimated from SPOR
Monograph
C                                          (used 10 times the k_min for each layer)
C ------------------------------------------------------------------------------------

KX CON
1900000000              ! best match to Olga Tubing dp for 0 - 1000 STBLIQ

MOD
 2 28 1 1 11 39 = 300          ! Tilje
 2 28 1 1 40 75 = 400          ! Aare

KY EQUALS KX

KZ EQUALS KX
MOD
 2 28 1 1 11 38 = 100          ! Tilje   (k_v/k_h = 0.5)
 2 28 1 1 40 75 =  50          ! Aare   (k_v/k_h = 0.1)

 2 28 1 1 39 39 =   0          ! make layers non-communicating (KZ = 0 is tighter than TZ = 0!!)

TRMINUS

C ------------------------------------------------------------------------------------
C POROSITY                                  <- values obtained or estimated from SPOR Monograph
C ------------------------------------------------------------------------------------

POROS CON
1
MOD
 2 28 1 1 11 39 = 0.25         ! Tilje
 2 28 1 1 40 75 = 0.27         ! Aare
 2 28 1 1  1 10 = 0.0          ! deactivate outside-tubing cells!


C ------------------------------------------------------------------------------------
C DEPTH
C ------------------------------------------------------------------------------------

DEPTH CON
6799.9

THICKNESS ZVAR
 0.1
 9*10                   ! SENSOR-Tbg part reaches 90.1 ft cells above top of Tilje
 20*5.0     9*40.0      ! Fmt gross thicknesses: Tilje = 460 ft (29 cells) and
 22*5.0    14*40.0      !                              Aare  = 670 ft (36 cells)

                        ! OLGA-Wellmodel reaches to 6800 ft (Depth of Fluid handover)
                        ! In effect there is an overlap of 0.1 ft for acurate mid-cell
                        ! pressure and rate calculation!!
```

```
C ------------------------------------------------------------------------------------
C Assigning grid blocks to the 3 initialization regions; 1 = tubing ; 2 = Tilje ; 3 = Aare
C ------------------------------------------------------------------------------------

INITREG CON
 1
MOD
 2 28 1 1 11 39 = 2
 2 28 1 1 40 75 = 3


INITIAL 1
  DEPTH          PSATBP
  6799.9 2000

  ZINIT   8020.   ! [ft]
  PINIT   3600.   ! [psia]
 C HWC 6800.   ! [ft]


INITIAL 2
  DEPTH          PSATBP
  6890    2000

  ZINIT   8020.   ! [ft]
  PINIT   3600.   ! [psia]
C  HWC 7300.   ! [ft]


INITIAL 3
  DEPTH          PSATBP
  7350    2000

  ZINIT   8020.   ! [ft]
  PINIT   3600.   ! [psia]
C  HWC 7970.   ! [ft]

ENDINIT

MODIFY TX
 2  2   1 1   15 39 = 0                  ! limits perforated interval to
 2  2   1 1   44 75 = 0                  ! 20 ft at the top of each interval

 2  3   1 1   11 14 = .2       ! limits T_perf
 2  3   1 1   40 43 = .2

C ------------------------------------------------------------------------------------
C Define Well A1
C ------------------------------------------------------------------------------------

WELL
        I  J  K   RW        PI
  A1    1  1  1  .145833334   1000        ! Producer in STB/d oil

        I  J  K   RW        PI
  IO    1  1  6  .145833334   1000        ! Oil Injector

        I  J  K   RW        PI
  IW    1  1  9  .145833334   1000        ! Water Injector
```

```
WELLTYPE

  A1  STBLIQ                        !  producer in stboil/d
  IO  RBGASINJ                      !  oil injector in rb/d
  IW  RBWATINJ                      !  water injector in rb/d

 INJGAS

 IO
  1

WINDOWS
    1  1 1   1 1   1 1      XZY   ! Coupling Point
C   2  1 1   1 1   2 2      XZY
C   3  1 1   1 1   4 4      XZY   ! Top of Tilje
C   4  1 1   1 1   20 20    XZY   ! Top of Aare
C   5  1 1   1 1   33 33    XZY
C   6  1 1   1 1   42 42    XZY
C   7  1 1   1 1   55 55    XZY
C   8  1 1   1 1   68 68    XZY


  MAPSPRINT 1  P

 MAPSFILEFREQ  -1               !         -> Mapsfile Printout
 MAPSFREQ     0                 !         -> Maps Printout: Print only at times entered
 STEPFREQ     1                 !         -> Timestep Table
 WELLFREQ     1                 !         -> Well Table
 SUMFREQ     -1                 !         -> EndOfRun Summary (every timestep)
 PRINTZERO    -1                !         -> To include zero rate lines
 PRINTREG    -1                 !         -> Region table printout (insert # of regions to be plotted)

C ------------------------------------------------------------------------------------------
C Pressure/Rate Schedule
C ------------------------------------------------------------------------------------------

RATE
  A1 100000 STBOIL                        ! oil rate for producer
  IO 0                                    ! injection rate for oil injector
  IW 0                                    ! injection rate for water injector

 BHP
  A1 500
  IO 3500
  IW 3152

 RESTART

DT        0.000001

TIME   1  .1

END
```

E. SENSOR restart file

```
TITLE
--------------------------------------------------------------------------------
           Radial 3-Phase Model

           2 Reservoir Zones (Tilje & Aare) are devided into 22 and 29 Layers.
           Tbg in the Reservoir section + 10 Gridcells is modeled with SENSOR.
           The Pressure at this point is determined by OLGA and returned after each step:
           SENSOR then delivers the phase rates accordingly.

           Radial Gridding is used: 28 radial   1 angular   77 in z-direction
--------------------------------------------------------------------------------

ENDTITLE

RESTART 159
C --------------------------------------------------------------------------------
C Pressure/Rate Schedule
C --------------------------------------------------------------------------------

WELL
          I  J  K   RW       PI
 A1       1  1  1  .145833   1000
 IO       1  1  6  .145833   1000
 IW       1  1  9  .145833   1000

WELLTYPE

 A1  STBOIL               ! oil producer in stboil/d
 IO  RBGASINJ             ! oil injector in rb/d
 IW  RBWATINJ                       ! water injector in rb/d

 INJGAS

 IO
  1

C --------------------------------------------------------------------------------
C Pressure/Rate Schedule
C --------------------------------------------------------------------------------

RATE
 A1 100000 STBOIL
 IO 0
 IW 0

BHP
 A1 3137.3
 IO 3500
 IW 3182.5


DT   0.0000001
RESTART
TIME     1.00090393518518          1.67824074074074e-05

END
```

F. PVT calculations

BASIC DATA			
	rho o sc =	846,265	[kg/m³]
	rho g sc =	1,44918	[kg/m³]
sweet	rho w sc =	998,104	[kg/m³]
salty	rho sw sc =	1026	[kg/m³]
	rho air sc =	1,225	[kg/m³]
	γg =	1,18300408	
	γo =	0,84787257	
	γAPI =	35,3882868	

$$\gamma_o = \frac{(\rho_o)_{sc}}{(\rho_w)_{sc}}$$

$$\rightarrow (\rho_w)_{sc} = 998.1039 \quad kg/m^3$$

$$\gamma_g = \frac{(\rho_g)_{sc}}{(\rho_{air})_{sc}}$$

$$\rightarrow (\rho_{air})_{sc} = 1.22381 \quad kg/m^3$$

$$R_s = \gamma_g \cdot \left[\frac{(0.055\ p + 1.4) \cdot 10^{0.0125\ \gamma_{API}}}{10^{0.00091\ T}} \right]^{1.205}$$

Temp [°F]	32	60	77	122	167	212
Temp [°C]	0	15,55	25	50	75	100
	Rs [scf/STB]					
14,50	9,6163	8,9599	8,5834	7,6616	6,8387	6,1042
725,19	329,5847	307,0885	294,1862	262,5896	234,3866	209,2127
1740,45	923,9871	860,9194	824,7478	736,1671	657,1003	586,5255
2755,72	1597,2174	1488,1977	1425,6709	1272,5490	1135,8730	1013,8764
3770,98	1597,2174	1488,1977	1425,6709	1272,5490	1135,8730	1013,8764
4786,24	1597,2174	1488,1977	1425,6709	1272,5490	1135,8730	1013,8764
5801,51	1597,2174	1488,1977	1425,6709	1272,5490	1135,8730	1013,8764
6816,77	1597,2174	1488,1977	1425,6709	1272,5490	1135,8730	1013,8764

$$B_o(p) = \frac{62.4\gamma_o + 0.0136\ \gamma_g \cdot R_s(p)}{\rho_o(p)}$$

Temp [°F]	32	60	77	122	167	212
Temp [°C]	0	15,55	25	50	75	100
	Bo [rb/STB]					
14,50	1,0000	1,0042	1,0068	1,0125	1,0244	1,0367
725,19	1,1022	1,1002	1,0992	1,0986	1,1018	1,1074
1740,45	1,3096	1,2968	1,2896	1,2727	1,2616	1,2546
2755,72	1,5374	1,5134	1,4999	1,4675	1,4415	1,4215
3770,98	1,5374	1,5158	1,5036	1,4767	1,4544	1,4381
4786,24	1,5299	1,5072	1,4944	1,4638	1,4415	1,4215
5801,51	1,5225	1,4987	1,4852	1,4548	1,4289	1,4089
6816,77	1,5152	1,4914	1,4780	1,4458	1,4200	1,3965

$$\rho_{g(p)} = \frac{\rho_{\overline{g}} + \rho_{\overline{o}} * rs_{(p)}}{Bg_{(p)}}$$

$$\Rightarrow rs(p) \approx 0$$

$$\Rightarrow Bg_{(p)} = \frac{\rho_{\overline{g}}}{\rho_{g(p)}}$$

Temp [°F]	32	60	77	122	167	212
Temp [°C]	0	15,55	25	50	75	100
	Bg [rb/Scf]					
14,50	0,1720731	0,1795171	0,1843641	0,1911924	0,1985459	0,2064878
725,19	0,0055388	0,0059096	0,0061601	0,0066352	0,0069948	0,0072098
1740,45	0,0019119	0,0021167	0,0022641	0,0025555	0,0028025	0,0029770
2755,72	0,0011421	0,0012677	0,0013585	0,0015549	0,0017093	0,0018436
3770,98	0,0008380	0,0009197	0,0009777	0,0011078	0,0012175	0,0013102
4786,24	0,0006453	0,0006881	0,0007170	0,0007893	0,0008490	0,0008839
5801,51	0,0004898	0,0004950	0,0004983	0,0005081	0,0005183	0,0005300
6816,77	0,0004090	0,0004127	0,0004150	0,0004211	0,0004280	0,0004345

	gas viscosity SENSOR [cp]					
14,50	1,16E-02	1,20E-02	1,22E-02	1,29E-02	1,35E-02	1,42E-02
725,19	1,16E-02	1,20E-02	1,22E-02	1,29E-02	1,35E-02	1,42E-02
1740,45	1,65E-02	1,60E-02	1,57E-02	1,57E-02	1,59E-02	1,63E-02
2755,72	2,54E-02	2,32E-02	2,18E-02	2,02E-02	1,96E-02	1,95E-02
3770,98	3,87E-02	3,39E-02	3,10E-02	2,70E-02	2,50E-02	2,40E-02
4786,24	6,13E-02	5,33E-02	4,85E-02	4,06E-02	3,61E-02	3,39E-02
5801,51	5,20E-02	4,83E-02	4,60E-02	4,10E-02	3,60E-02	3,10E-02
6816,77	5,40E-02	5,09E-02	4,90E-02	4,40E-02	3,90E-02	3,40E-02

	liquid viscosity SENSOR [cp]					
14,50	1,52E-01	1,50E-01	1,48E-01	1,43E-01	1,38E-01	1,32E-01
725,19	1,52E-01	1,50E-01	1,48E-01	1,43E-01	1,38E-01	1,32E-01
1740,45	1,47E-01	1,44E-01	1,42E-01	1,38E-01	1,33E-01	1,28E-01
2755,72	1,45E-01	1,42E-01	1,40E-01	1,34E-01	1,29E-01	1,24E-01
3770,98	1,46E-01	1,43E-01	1,41E-01	1,35E-01	1,29E-01	1,22E-01
4786,24	1,49E-01	1,46E-01	1,44E-01	1,38E-01	1,33E-01	1,27E-01
5801,51	1,52E-01	1,48E-01	1,46E-01	1,42E-01	1,36E-01	1,31E-01
6816,77	1,54E-01	1,51E-01	1,49E-01	1,44E-01	1,39E-01	1,34E-01

The highlighted values are needed for the SENSOR input files.

PVT data for Sensor input file

C SATURATED PVTBO Table						
C	psig	rb/stb	rb/scf	scf/stb	cp	cp
	PSAT	BO	BG	RS	VISO	VISG
	2741,22	1,441506	0,001709	1135,873	0,1290	0,0196
	1725,95	1,261591	0,002802	657,1003	0,1330	0,0159
	710,69	1,101821	0,006995	234,3866	0,1380	0,0135
	0,00	1,024436	0,198546	0	0,1380	0,0135

C UNDERSATURATED PVTBO Table			
C	psig	rb/scf	cp
	P	BG	VISG
	6802,27	0,000428	0,0390
	5787,01	0,000518	0,0360
	4771,74	0,000849	0,0361
	3756,48	0,001217	0,0250

G. Pipe-It info file for initialization

INFO FILE

SENSOR Shut-In		
BHP extended	3151.5	[psia]
THP extended	676.0	[psia]
BHP	3151.5	[psia]
TS end	37	
TIME end	1.0	[d]
OLGA Shut-In		
BHP	3151.59193358435	[psia]
THP	755.322447771931	[psia]
TIME end	36003.0096	[s]
Void Fraction	0.02578125	
OLGA after gas release		
BHP	3150.99292796268	[psia]
THP	755.443699273071	[psia]
Time end	39605.7434	[s]
Oil V flow top	-0.2384632393003	[bbl] accumulated
Oil V flow bottom	-0.2126382229118	[bbl] accumulated
Gas V flow top	0.06786032	[m3] accumulated
Gas V flow bottom	-0.03380678	[m3] accumulated
Water V flow top	6.28981e-20	[bbl] accumulated
Water V flow bottom	6.28981e-20	[bbl] accumulated
General Information		
dp Oil Injector	15.5	[psi]
dp water injector	31.5	[psi]
Steady state flow rate	4000	[STB/d]

H. Pipe-It info file for restart runs

INFO FILE

SENSOR Times after Initialization

T end	1.0	[d]
TS end	37	[-]

OLGA Time after Initialization

T end	39605.7434	[s]

SENSOR Current Time and Time step

T Sensor start	1.0243865740741	[d]
TS Sensor start	4580	[-]
T Sensor end	0	[d]
TS Sensor end	0	[-]

OLGA current Time

T OLGA start	41713.4012	[s]
T OLGA end	0	[s]

Simulation:

Shut-in period	15	[h]
TS length	14.9	[s]

INFO Restart Iter.	0	[-]

I. Miscellaneous

List of abbreviations, subscripts & nomenclature

Δ	-	differential
BHP	–	bottom hole pressure
cf.	-	confer
CPU	-	central processing unit
CV	–	control variable
e.g.	-	exempli gratia (for example)
Eq.	-	equation
et al.	-	et alii (and others)
FVF	-	formation volume factor
FTS	--	fixed time step
GLR	-	gas-liquid ratio
GOR	–	gas-oil ratio
GUI	--	graphical user interface
HSE	--	health, safety & environment
HWC	–	hydrocarbon water contact
i.e.	--	id est (that is)
IPR	–	inflow performance relationship
mD	--	milli-Darcy
o, g and w	–	phases oil, gas and water
p	–	pressure
PI	--	productivity index
PVT	–	pressure-volume-temperature
Rs	--	solution oil gas ratio
SENSOR	–	system for efficient simulation of oil recovery
SI units	--	Systéme Inernational d'Unités
SPE	–	Society of Petroleum Engineers
SSPP	--	Steady state pre-processor
STB	–	stock-tank barrel
T	–	temperature
TH	--	tubing head
THP	–	tubinghead pressure
TS	–	time-step
TVD	--	true vertical depth
VLP	–	vertical lift performance
VTS	–	variable time step
WC	--	water cut
WH	--	well head
WOC	--	water oil contact
ρ	–	density
μ	–	viscosity
γ	–	specific weight

SI metric unit conversions

bar	x 100,000	= Pa
bbl	X 0.158987	= m^3
bbl/d	x 1/543,440	= m^3/s
cp	x 0.001	= $N\text{-}s/m^2$
F	(F-32)/1.8	= °C
ft	x 0.3048	= m
ft^3, cuf	x 0.0283169	= m^3
in	x 0.0254	= m
lb, lbm	x 0.453593	= kg
lb/ft^3, lbm/ft^3	x 16.0185	= kg/m^3
psi, psia, psig	x 6894.76	= Pa
scf/STB	x 0.17811	= m^3/m^3