Published by
NIGERIAN SOCIETY OF PHYSICAL SCIENCES
Available online @ https://asr.nsps.org.ng/index.php/asr

African Scientific Reports 3 (2024) 167

African Scientific Reports

# Development of 3D flow Diagnostic Numerical Simulator and Rock Property Modelling for a homogeneous faulted Reservoir ☆

Surajo Muhammed Gwio*, Ibrahim Ayuba, Umar Faruk Aminu

*Department of Petroleum Engineering, Abubakar Tafawa Balewa University Bauchi, Bauchi State, Nigeria*

## Abstract

Research has shown that faults, especially sealing faults, have become a barrier to flow. Flow diagnostics, on the other hand, are straightforward and controlled numerical flow experiments that are performed to examine a reservoir model, build links and offer rudimentary volume estimates as well as to quickly produce a qualitative representation of the reservoir's flow patterns. To evaluate, assess, and validate reservoir models and production scenarios, the 'diagnostics' module in the Matlab Reservoir Simulation Toolbox (MRST) provides a computationally less expensive alternative to running fully featured multiphase simulations. This is done to determine qualitatively regions for well placement to maximize hydrocarbon recovery. Monitoring flow fields, timelines, and tracers—neutral particles in the fluid that flow with the fluid—are all part of flow diagnostics. These techniques are used to determine volumetric communication between a pair of injector-producer wells, as well as flow patterns between injection-production wells and the arrival time between them. To create the grid, the simulation schedule, the wells, upscaling, arrival time computation, and flow diagnostics (drainage and sweep regions) were all done using the Matlab reservoir simulation toolbox (MRST) package by "Makemodel 3" on the data obtained from "SPE $10^{th}$ comparative solution project" on the MRST package. This was done for a homogeneous anisotropic faulted porous medium. The simulation runs for about 473,739 milliseconds on a fine grid having 20040 active cells while in the coarse model with faster computational time of about 200 seconds, 112 milliseconds. At the early stages of the reservoir bottom-hole pressure for injector I1(about 7000psia) and three producers P1, P2 and P3 remains constant over time (about 3600psia) thanks to pressure support by I1. The extent of residence time after 10 years for the fine, coarse and more coarser model is that fluid element find it easier to transverse through the fault line from I1 to P1, P2 then P3 which takes lesser time than crossing across two intersecting faults, It can also be seen that the sweep efficiency of the fine grid is more effective than the coarse grid. Finally, it has been seen clearly that the fine grid model gives more accurate details but slower computational time while the coarse model which is less accurate but with faster computational time.

DOI:10.46481/asr.2024.3.1.167

*Keywords:* Faults, Flow diagnostics, Numerical modeling and simulation, Sweep efficiency, Reservoir

*Article History :*
Received: 26 July 2022
Received in revised form: 05 October 2022
Accepted for publication: 14 October 2022
Published: 01 February 2024



## 1. Introduction

The phrase "flow diagnostics" and the term "numerical flow experiments" as used in this article refers to a group of simple and carefully controlled numerical flow experiments carried out to verify a reservoir model, establish correlations, and provide basic volume estimates as well as to quickly produce a qualitative depiction of the reservoir's flow patterns. The techniques may also be

*Corresponding author: Tel.: +234-806-3935-643;
*Email address:* Intelligent207@gmail.com (Surajo Muhammed Gwio)

used to do what-if and sensitivity analyses in parameter ranges around current simulations, as well as to provide quantitative data about the recovery process in circumstances that are a little less complex than those encountered in a real field. These methods provide an alternative to fully featured multiphase simulations that are computationally less expensive for ranking, assessing, and verifying reservoir models, upscaling tactics, and production scenarios. As flow diagnostics are constructed using a (single-phase) pressure solution, they can be useful tools for examining flow patterns and well allocation factors. To evaluate, assess, and validate reservoir models and production scenarios, the 'diagnostics' module in the Matlab Reservoir Simulation Toolbox (MRST) provides a computationally less expensive alternative to running fully featured multiphase simulations. The data "SPE $10^{th}$ comparative solution project" which is contained in the MRST toolbox, and added to path is computed by the fundamental diagnostic routine: Time of Flight, Tracer Partition, and Tracer distributions [1].

The planning and implementation of production plans for field development is another area where flow diagnostics may be beneficial. Here, the continuing digitization creates opportunities for decision-support technologies that employ mathematical models to organize experts' expertise with growing volumes of data to assist quicker and wiser judgments. To put it another way, models with embedded uncertainty may be utilized in addition to stringent optimization methods to improve decisions about the well site, drilling order, completion of the well with a variety of inflow control devices, and selection of time-varying rate/pressure targets. Despite continuous advancements in technology, the high computational cost of a full forward reservoir simulation still limits the number of trials that one may afford while addressing an optimization problem [2].

Prior research that aims at creating diagnostic tools mainly concentrated on streamline simulation techniques. As a result, the diagnostic tools produced have been used, for example, in the ranking and scaling of geostatistical models, in water flooding well rate optimization, in pattern-by-pattern flood surveillance, and in tight gas reservoirs fracture stage optimization and horizontal well completion optimization. According to [3] the flexibility and computing complexity of streamline-based approaches are constrained, even though they can be quite successful for this and similar aims. Standard reservoir models made up of hexahedral cells are very efficient for streamline tracing, however mismatched cells need special handling, and suggested expansions to unstructured grid models are far less effective. Thus, for complicated earth models with unstructured grid cells, the use of streamlines may be prohibited by efficiency considerations or challenging to use for multi-continuum models. Additionally, when presented as grid cells rather than intricate bundles of intertwined streamlines from a visual perspective [4].

This study aims to provide a whole procedure for analyzing a simulation model and computing flow diagnostic values. The model we built up is a two-phase oil/water system with several wells that is somewhat compressible. The reservoir includes four intersecting faults and stratified strata. A full reservoir model, on the other hand, integrates the mathematical model, the numerical techniques, and the computer algorithm. The actions done include the following;

## 2. Basic definitions

### 2.1. Time of flight

Aarnes *et al*. [5] defined time of flight as the amount of time it would take a hypothetical, neutral particle to travel from the closest injector to a specific location in the reservoir. Natural timelines in the reservoir are defined by time-of-flight from all injectors. According to Hui *et al*. [6] the distance between a given place and the closest producing well may be defined as the reverse time-of-flight. The total residence time of the associated flow route may be calculated by adding the two time-of-flight values. Time-of-flight calculations can also be made using the linear steady-state transport equations, although streamline methods are often associated with it.

$$\begin{aligned} \overrightarrow{v} \,.\, \nabla_{\tau f} &= 0, \quad \tau f_{|inflow} = 0, \\ -\overrightarrow{v} \,.\, \nabla_{\tau b} &= 0, \quad \tau b_{|outflow} = 0. \end{aligned} \tag{1}$$

### 2.2. Tracer partitions

Pal *et al*. [7] state tracer partition to be Communication patterns within the reservoir which may be discovered by simulating the evolution of an artificial, neutral tracer with concentration into injection wells or fluid sources. Although the steady state is typically difficult to reach in field trials, it is simple to calculate mathematically. Therefore, by doing the tracer test again for each well, well completion, or section of the inflow border, one may quickly divide a model into sweeping volumes. Additionally, one may construct comparable tracer distributions linked to producers and outflow borders by simply reversing the flow field [8]. Despite providing two different types of transport equations [9],

$$\begin{aligned} \nabla \,.\, \left(\overrightarrow{v} c_i\right) &= 0, \quad c_{i\,inflow} = 1, \\ -\nabla \,.\, \left(\overrightarrow{v} c_p\right) &= 0, \quad c_{p_{outflow}} = 1. \end{aligned} \tag{2}$$

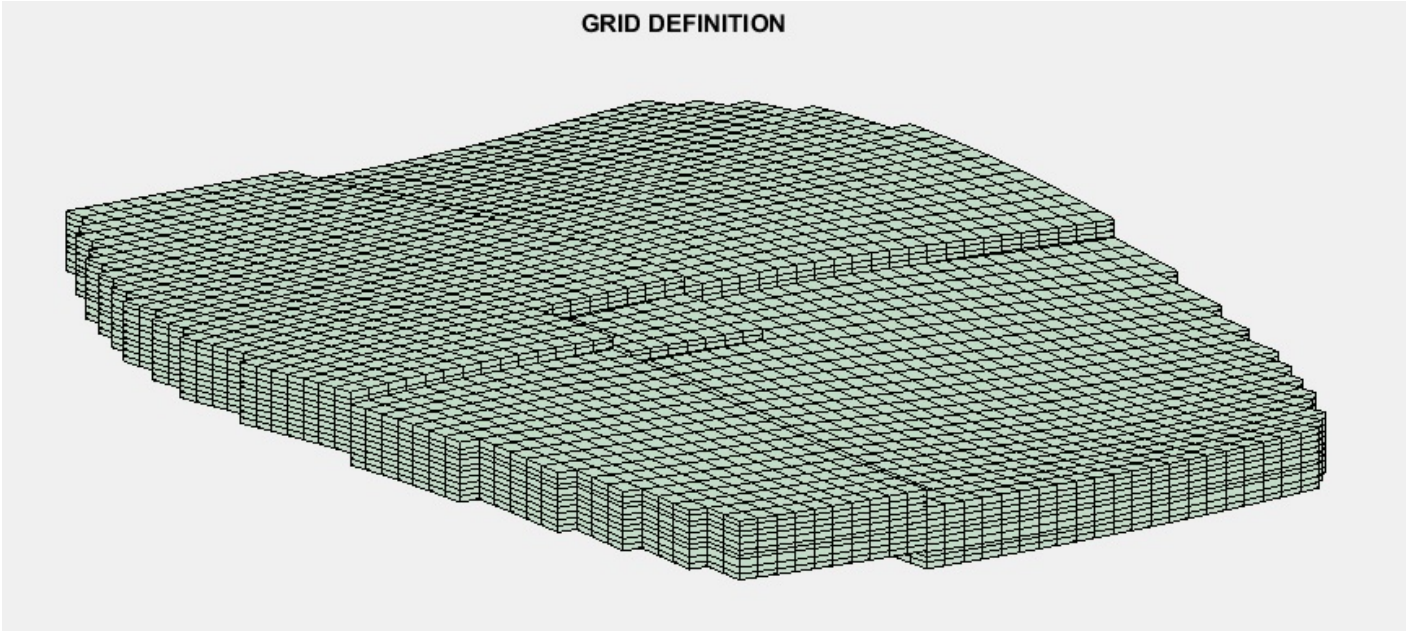


Figure 1: Reservoir topology with two intersecting faults created using ”makeModel3” routine in MRST.

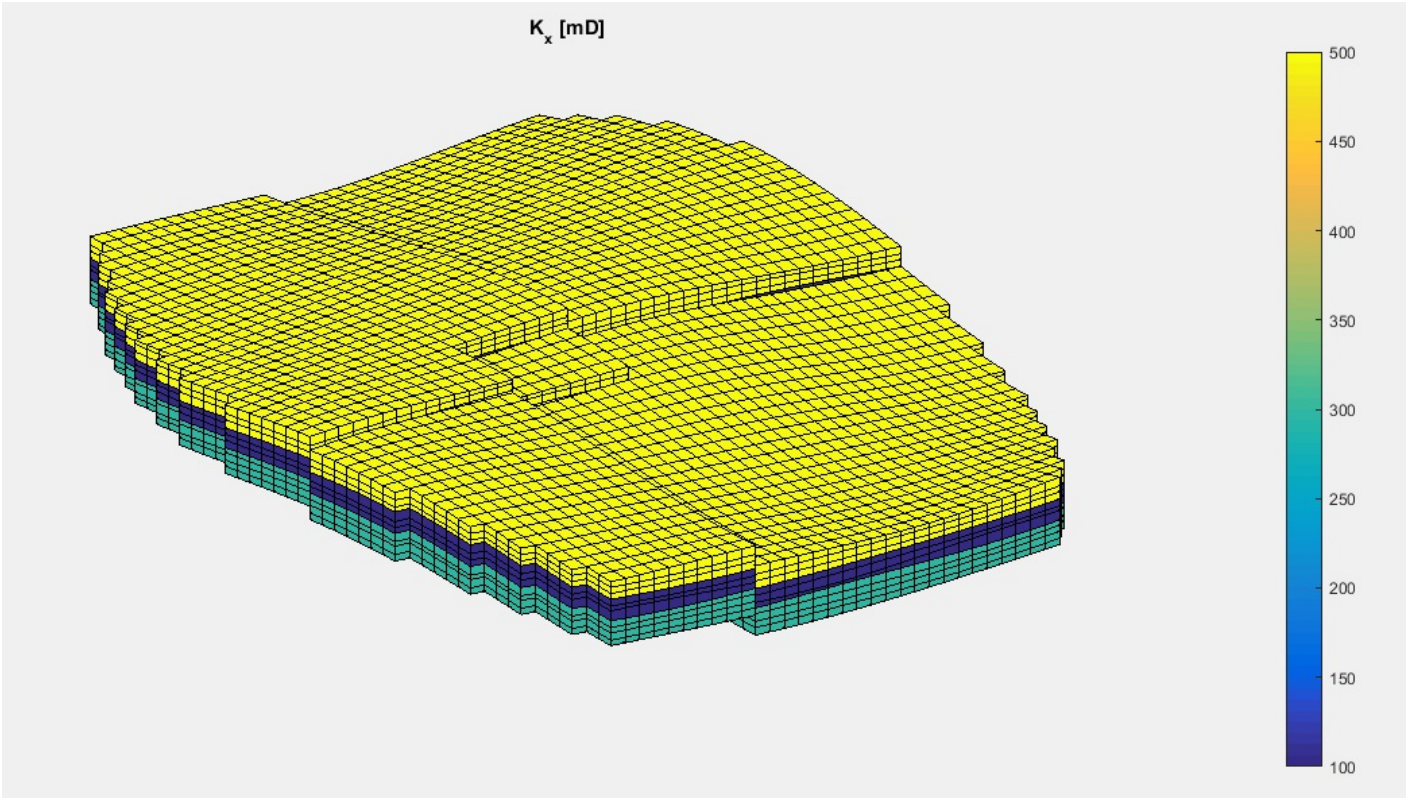


Figure 2: Layered anisotropic permeability distribution.

## 3. Methodology

The methodology of this research involves modeling a 3Dimensional flow diagnostics simulator, by deploying Matlab reservoir simulation toolbox (MRST). The data set used was obtained from SPE $10^{th}$ comparative solution project, which is part of the packages contained in the toolbox. The model developed is a slightly compressible two-phase oil/water system with multiple wells, it also contains layered stratigraphy and four intersecting faults [10]. The complete sequence of the methodology to achieve the research

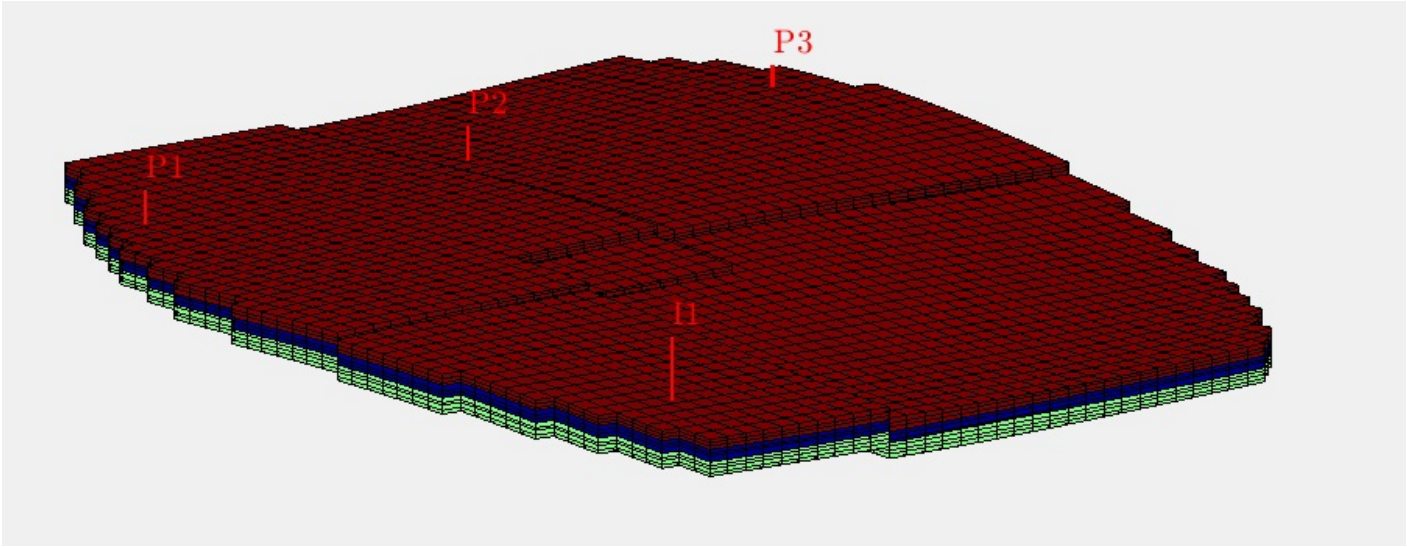


Figure 3: Complete Simulation schedule with three producers, single injector.

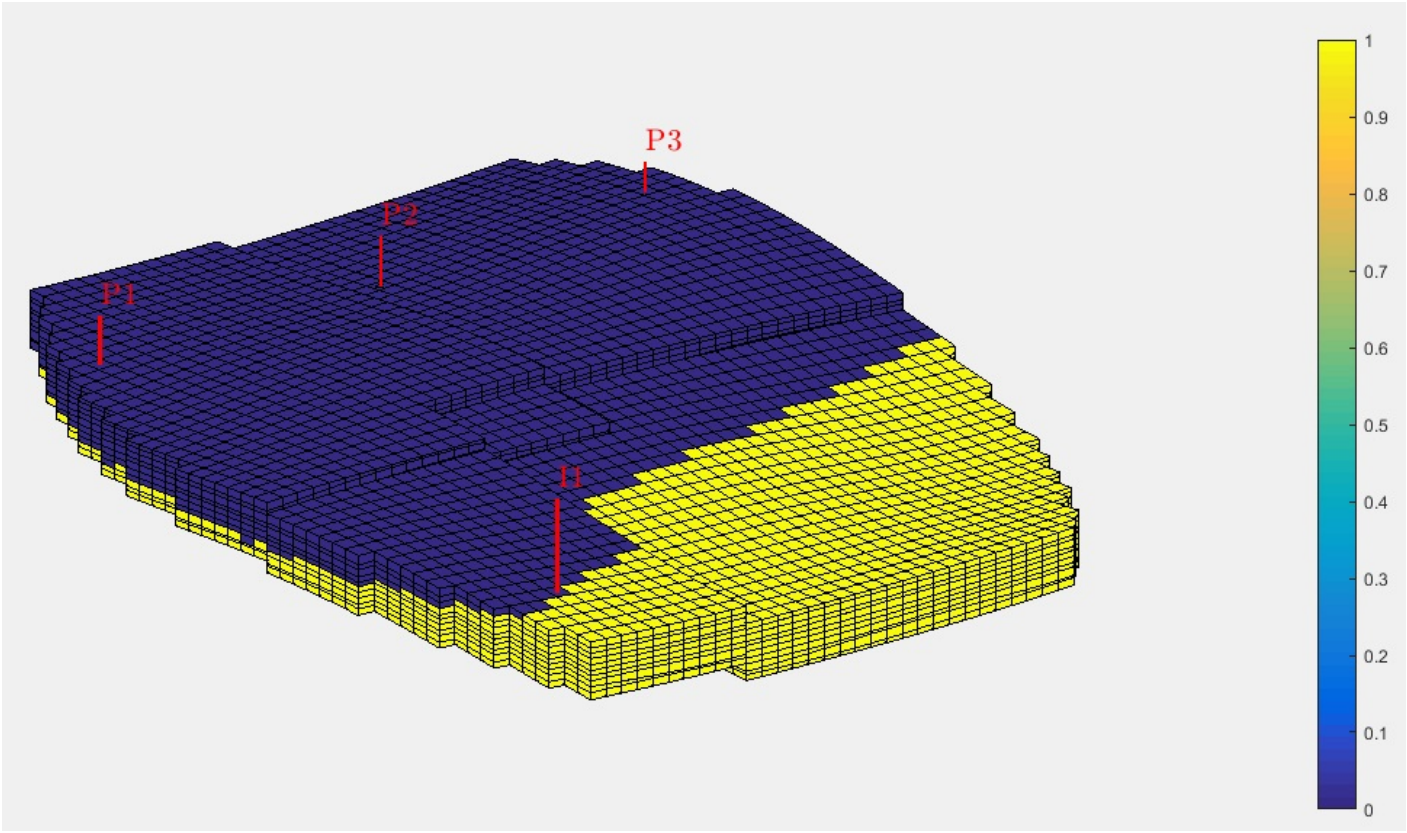


Figure 4: Initial Reservoir State completely oil and water wet in two partitioned zones.

objectives, are well outlined chronologically below;

### *3.1. Geometry and petrophysical characteristics of reservoirs*

The grid and the rock constructions are first constructed. ”makeModel3” generates the grid by building a structured model with intersecting faults. We assume that the lowest, middle, and top layers of the layered permeability structure have thicknesses of 300, 100, and 500 md, respectively.

Figure 1 shows the reservoir topology with two intersecting faults created using ”makeModel3” routine in MRST. Since it’s supposed that the model has a layered permeability, we populate the empty grid shown on Figure 2 petrophysical properties and

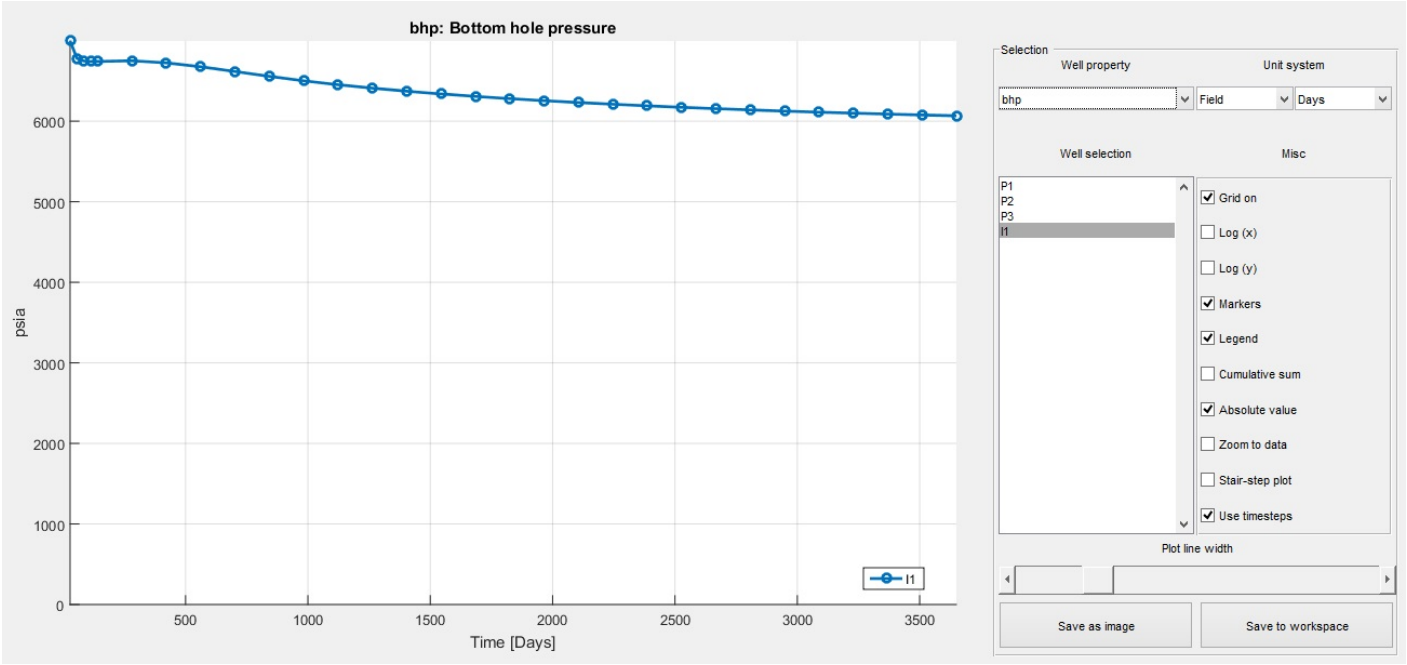


Figure 5: Plot of bottom hole pressure against Time for the injector I.

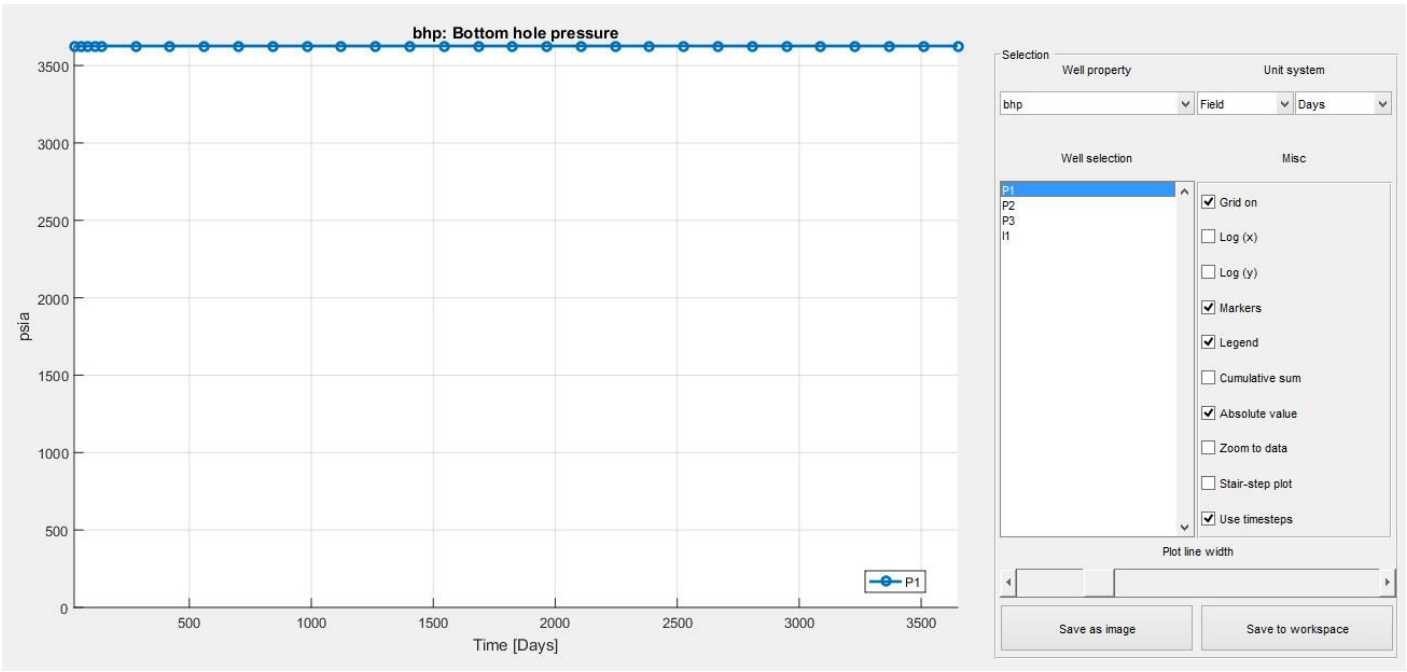


Figure 6: Plot of bhp against Time for producer P1.

porosity averaged across and permeability using logical K indices in MATLAB [11]. The resulting geological model is shown in Figure 2;

### *3.2. Schedule for defining wells and simulations*

All levels of the model are perforated as was the case for the SPE $10^{th}$ model, with hydrocarbon that has been collected from producers and is functioning at a constant bottom-hole pressure. A single water injector that is programmed to inject a single pore volume during a 10-year simulation period supports the manufacturers. Additionally, we created a plan that started with 5 modest control measures and progressed to 25 bigger milestones. The well controls are fixed for the duration of the simulation. The model

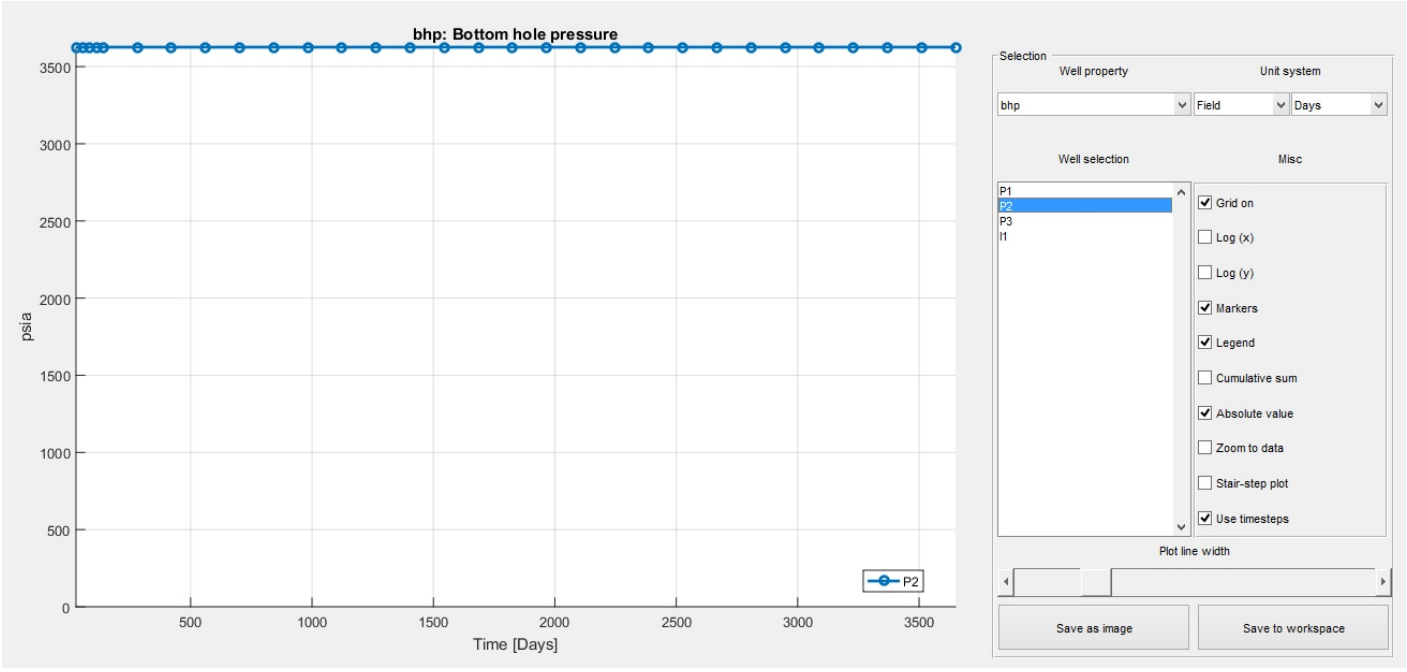


Figure 7: Plot of bhp against Time for producer P2.

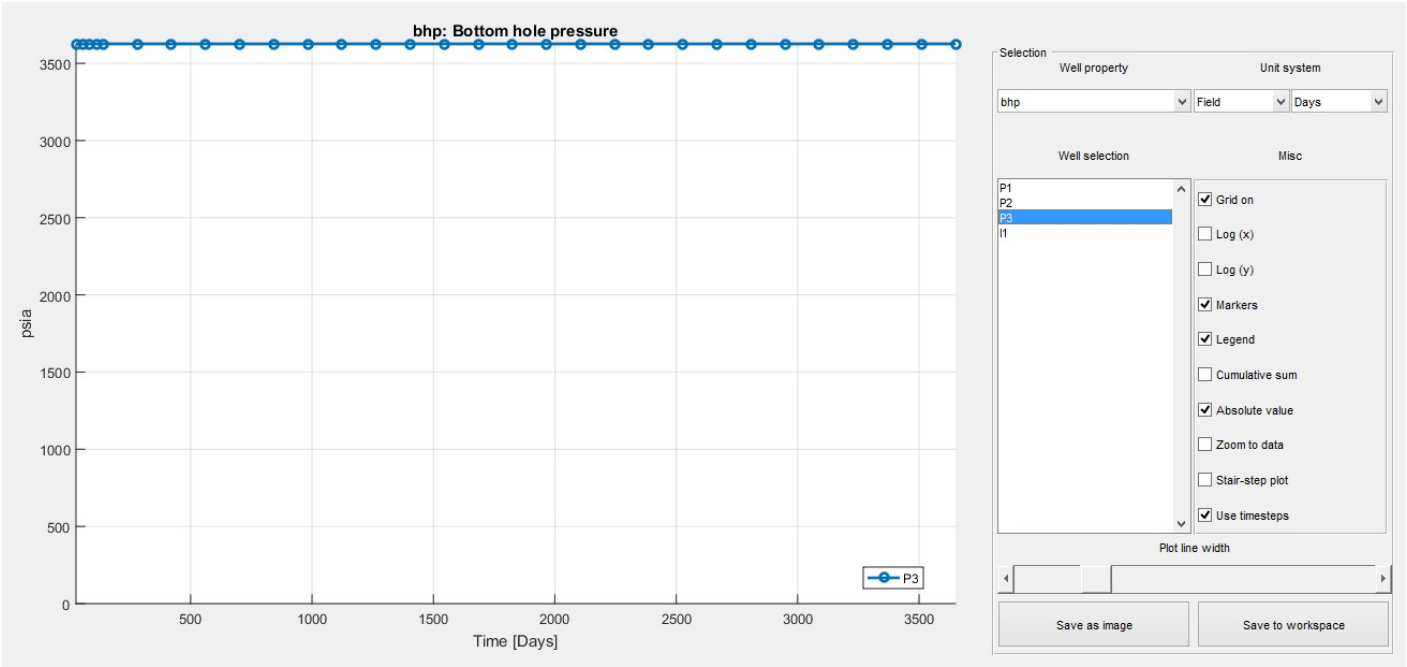


Figure 8: Plot of bhp against Time for producer P3.

has four wells with a single injector isolated from three producers by two bounding intersecting faults as shown in Figure 3. Figure 4 shows the initial reservoir state completely oil and water wet in two partitioned zones.

### *3.3. Setting-up a model for simulation*

Based on automated differentiation, we created a two-phase oil-water simulation model. The result is a specific instance of a two-phase generic model. Once we create the two-phase object, the gas attributes won't be considered [12]. Oil has constant compressibility compared to water's supposed incompressibility, resulting in a reciprocal formation volume factor of the form,

$$B_o(p) = B_{oi}\, Exp[c(P - P_o)]. \tag{3}$$

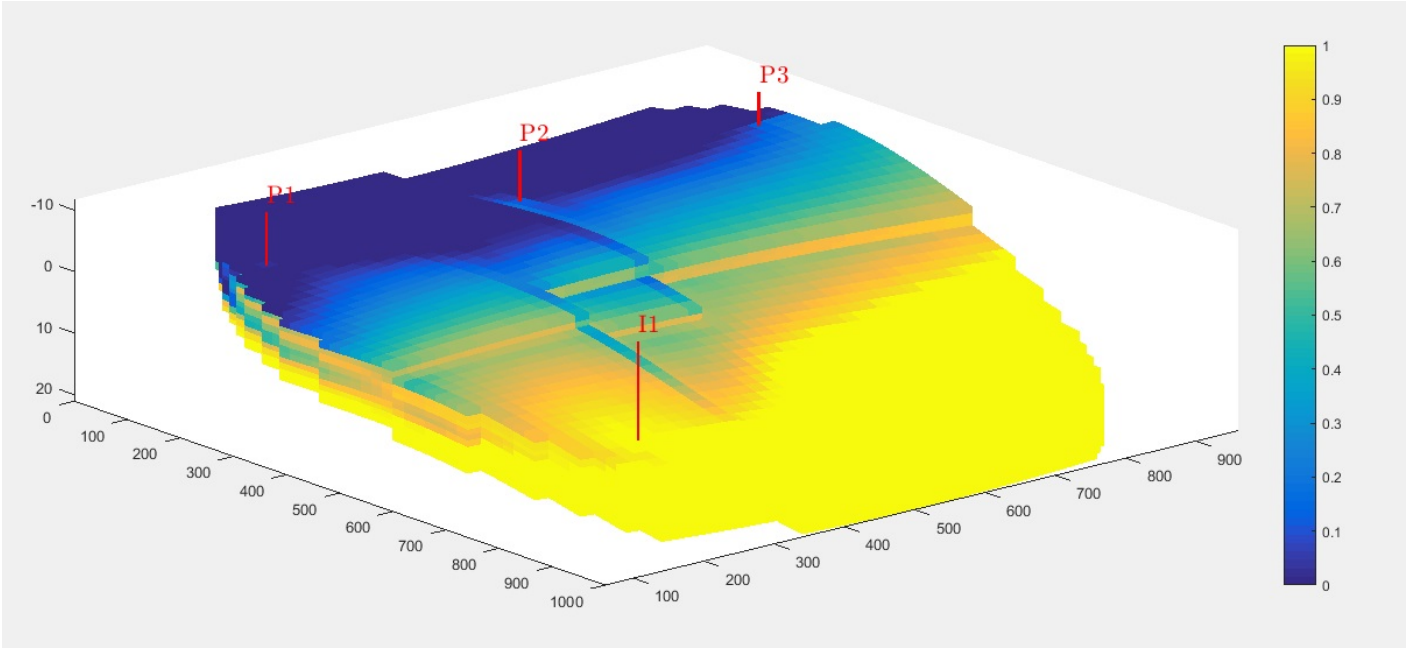


Figure 9: Net Reservoir State after 10 years.

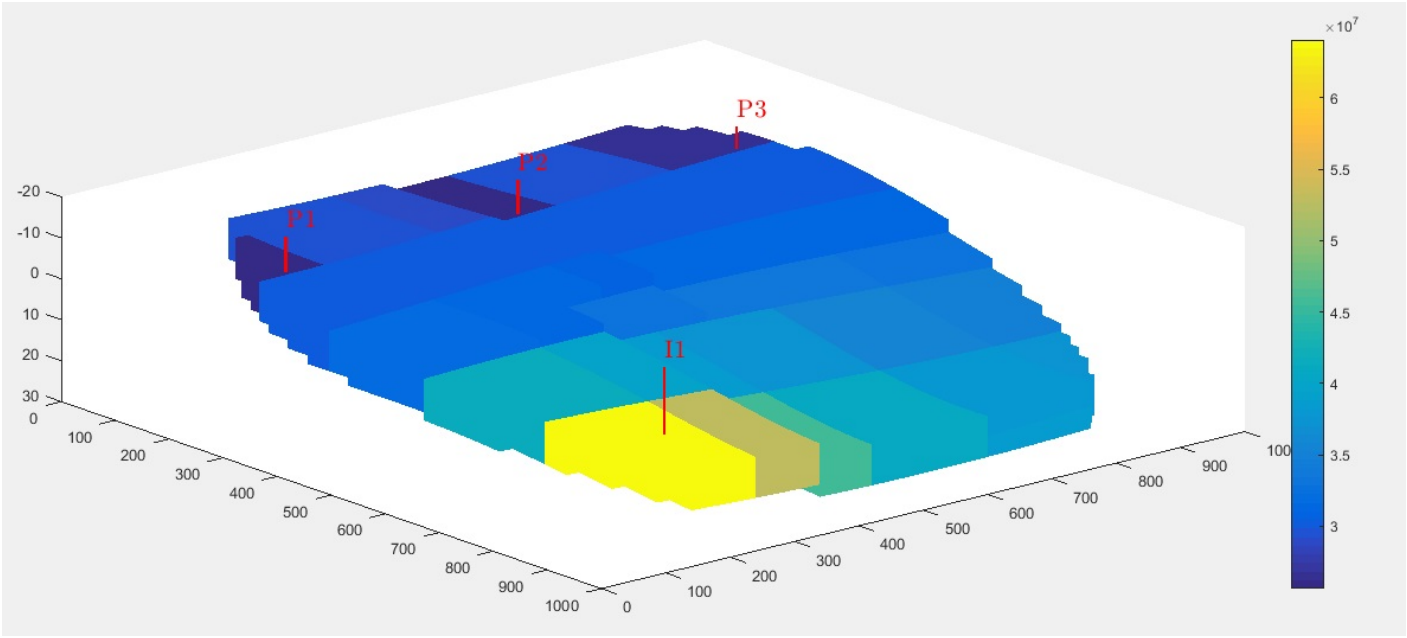


Figure 10: Net Reservoir State after 10 years for coarse blocks.

### *3.3.1. Defining initial state and simulation model*

After creating a model, we must establish an initial state. We created a relatively straightforward beginning condition by filling the reservoir's bottom half entirely with water and its upper half entirely with oil. We put water in the first column and oil in the second for the saturations since MRST internally employs the ordering of water, oil, and gas.

### *3.3.2. Make a more complex, coarser model*

There are about 20000 cells in the fine size model. An upscaled model is simple to define if we desire a smaller model. Here, we create a straightforward, uniform split of about 50 cells based on the grid's IJK indices.

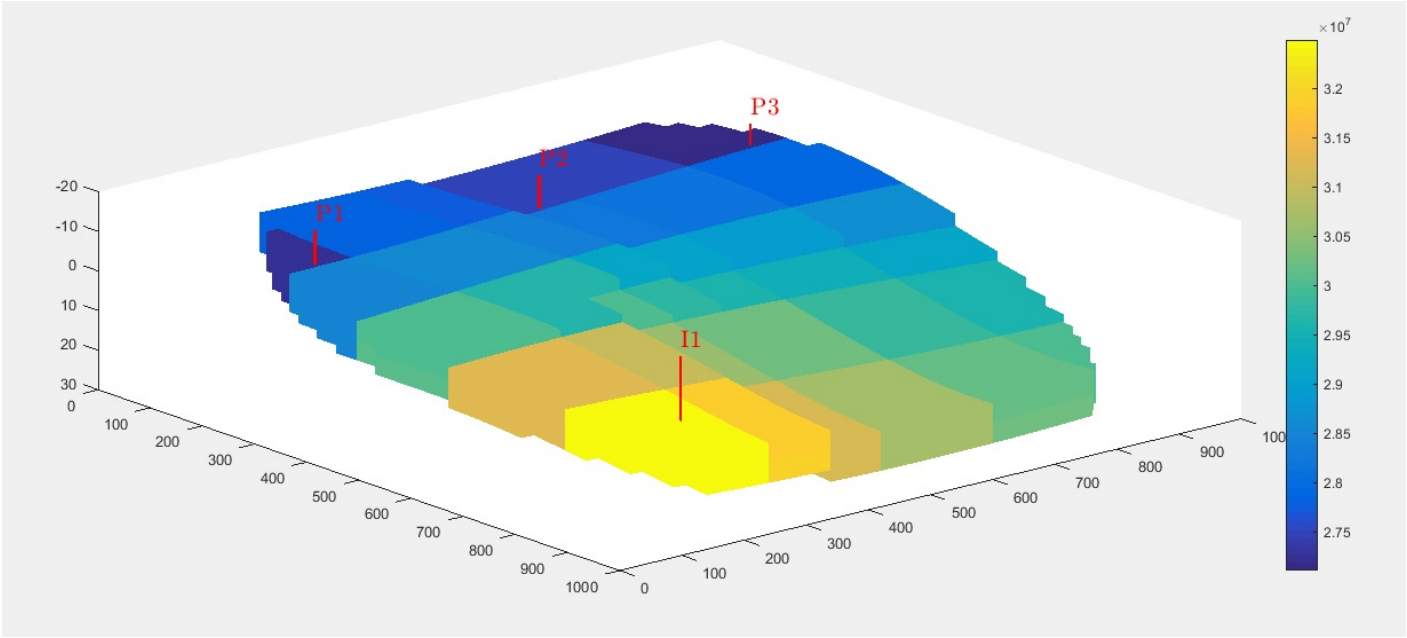


Figure 11: Net Reservoir State after 10 years for flow based coarse blocks.

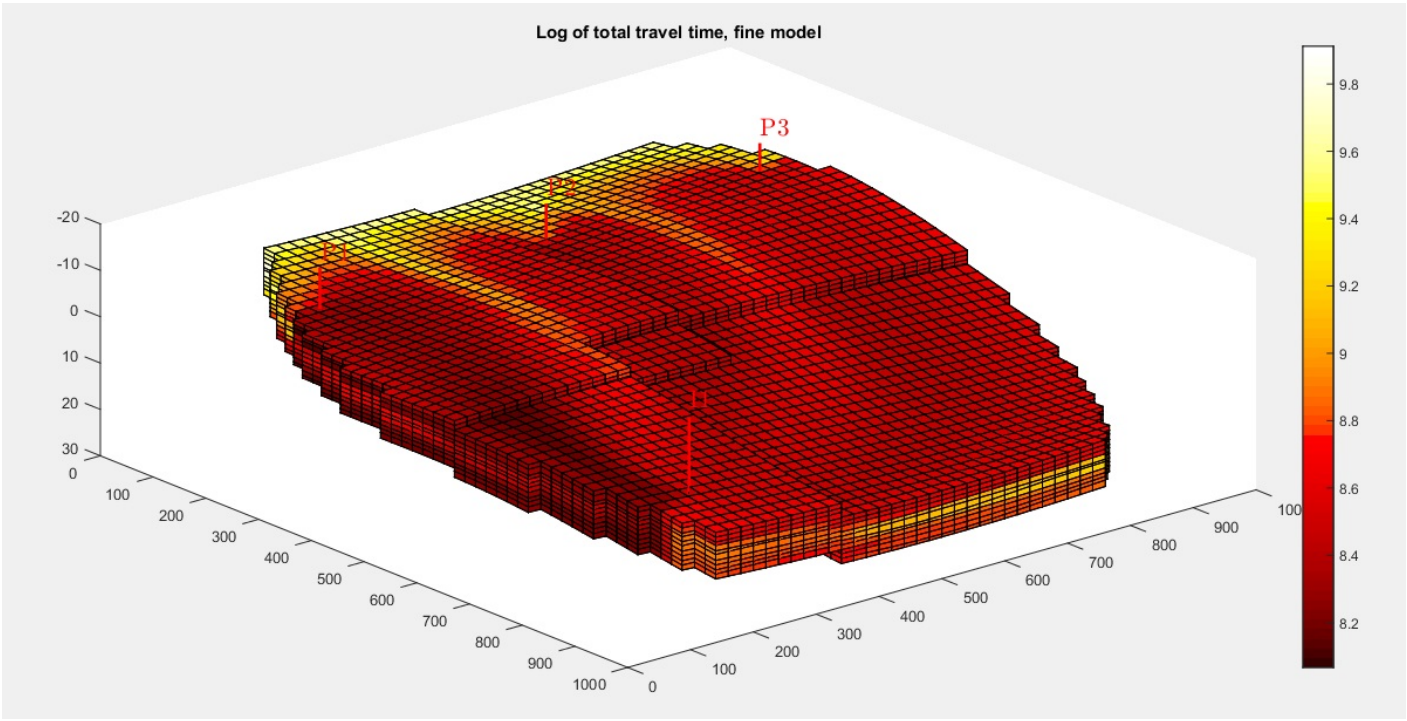


Figure 12: Plot of log of total time for fine Grid model

### *3.4. Perform flow diagnostics*

We may also examine the flow diagnostics of the models as an alternative to looking at well curves. A qualitative knowledge of reservoir dynamics is sought by flow diagnostics, which are straightforward procedures [13]. Here, we construct time-of-flight and well tracers using the end-of-simulation states as a snapshot for both the fine and coarse model.

#### *3.4.1. Schedule arrival times*

After the abovementioned procedure, the next thing is to plot arrival time as log of normal time for the based mode (fine grid), the Coarse and subsequently for the more coarser grid model. This is to monitor which grid type will have less time of flight and

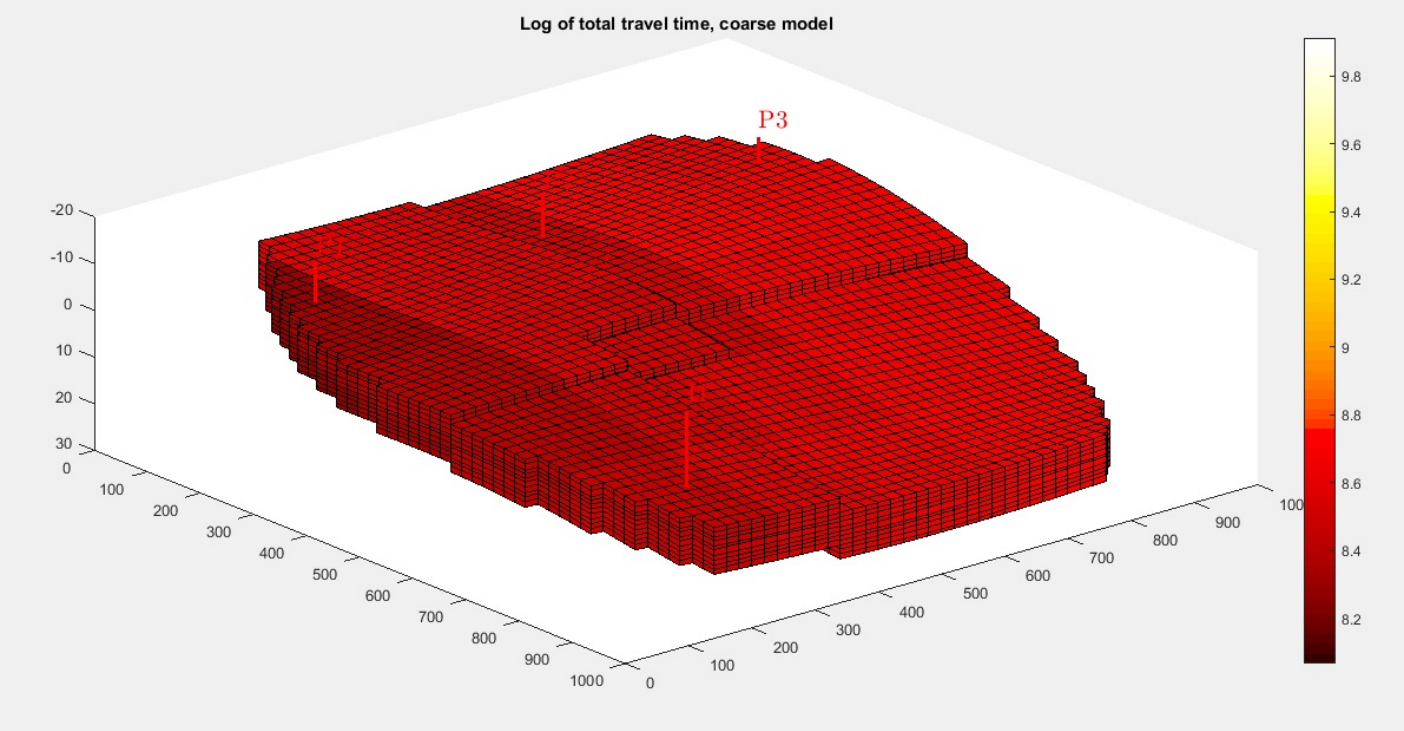


Figure 13: Plot of log of total time for coarse Grid model.

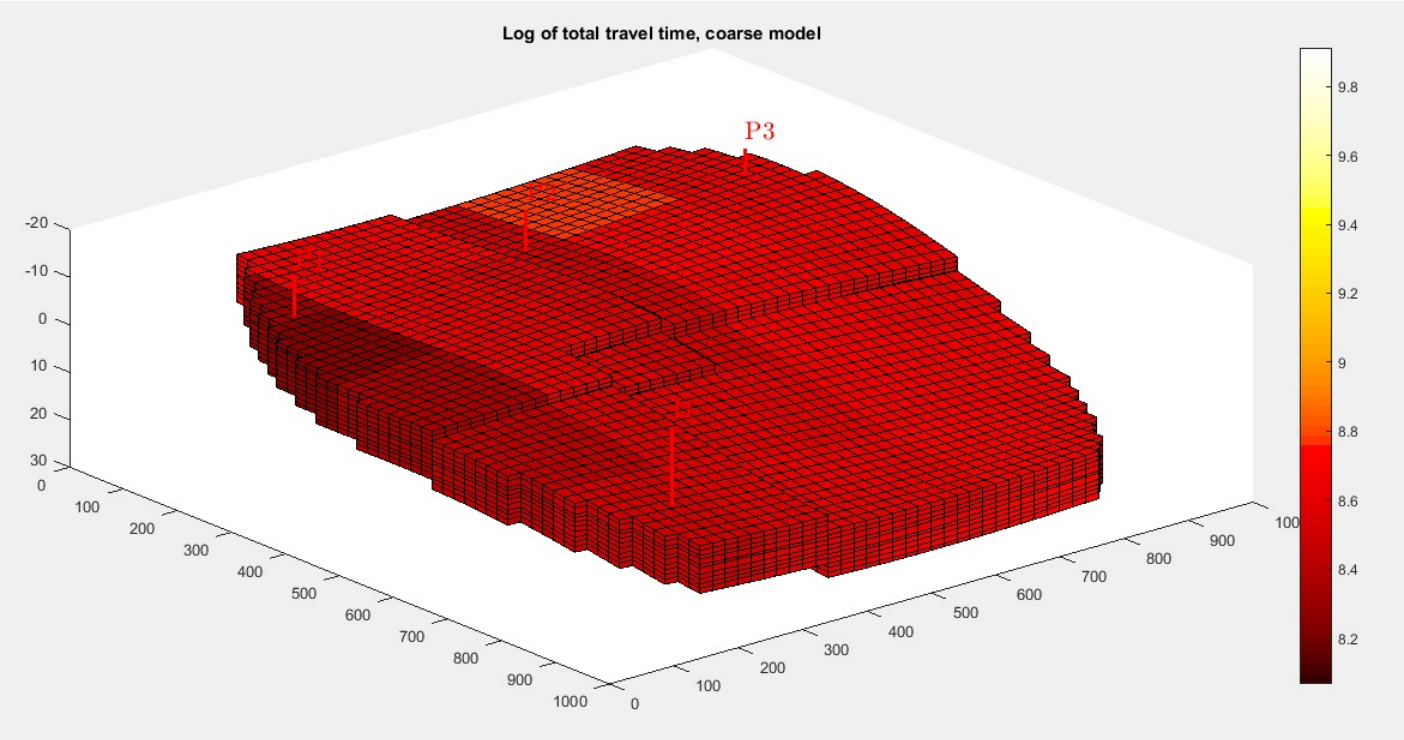


Figure 14: Plot of log of total time for flow based coarse Grid model.

tracer partitioning based on the travel time and particle velocity [14].

### *3.4.2. Tracer partitioning plot*

The drainage areas for the many wells are shown by the tracer partitioning for the producers, which is another option.

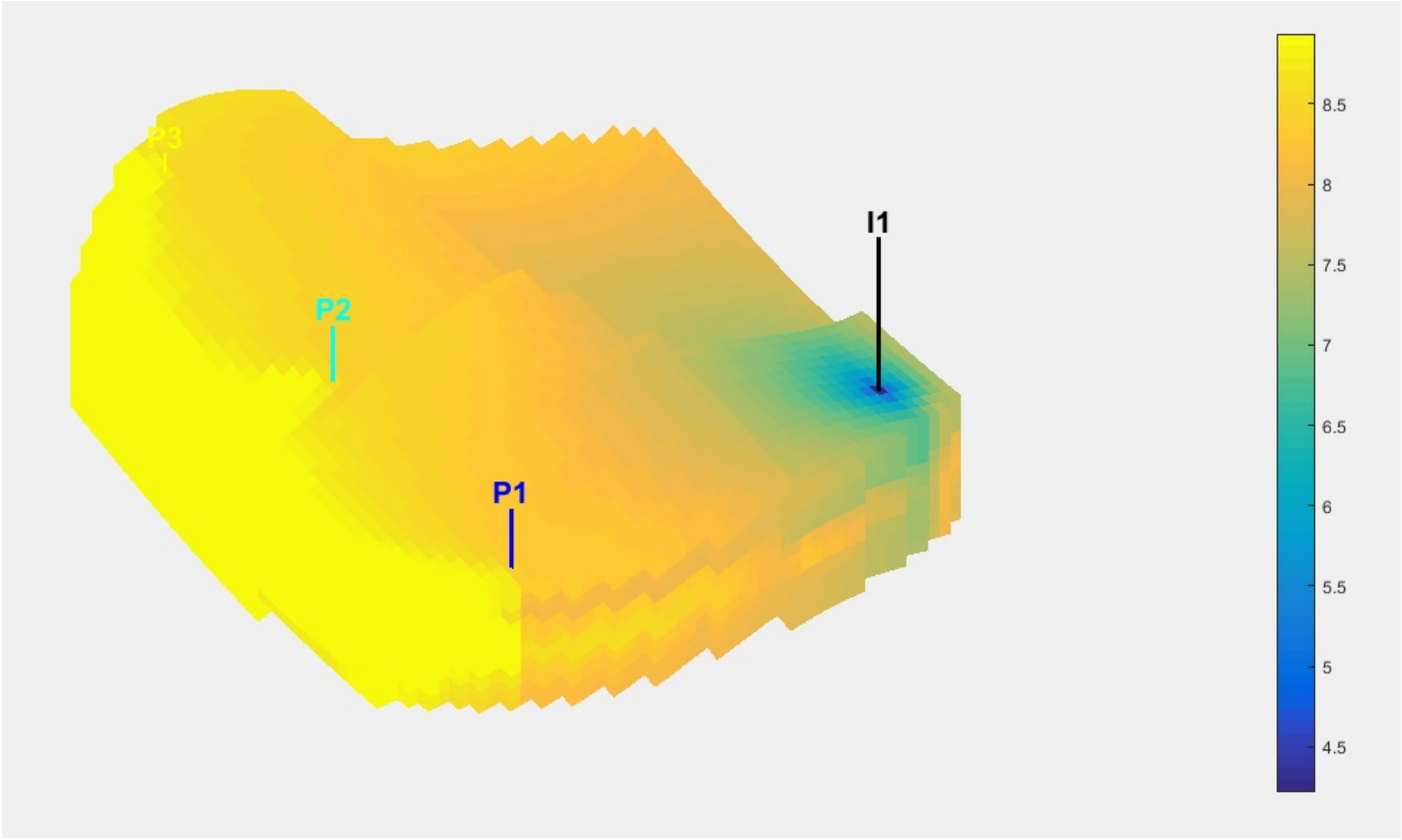


Figure 15: Plot of Tracer Volumetric Partitioning for fine Grid Model.

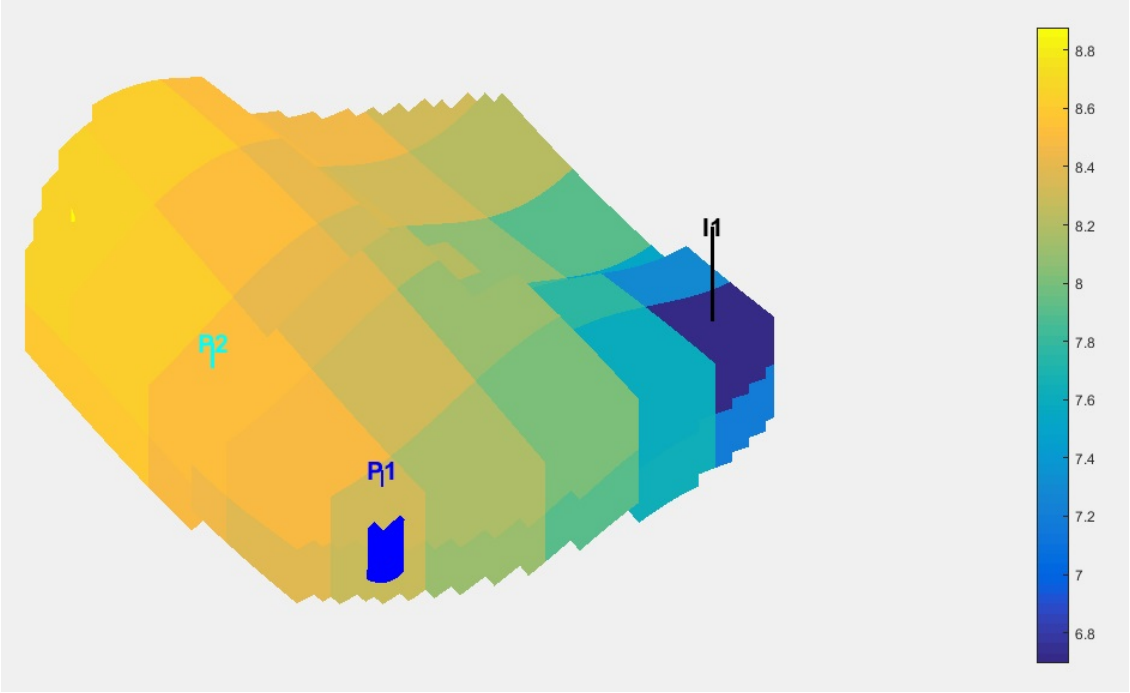


Figure 16: Plot of Tracer Volumetric Partitioning for coarse Grid Model.

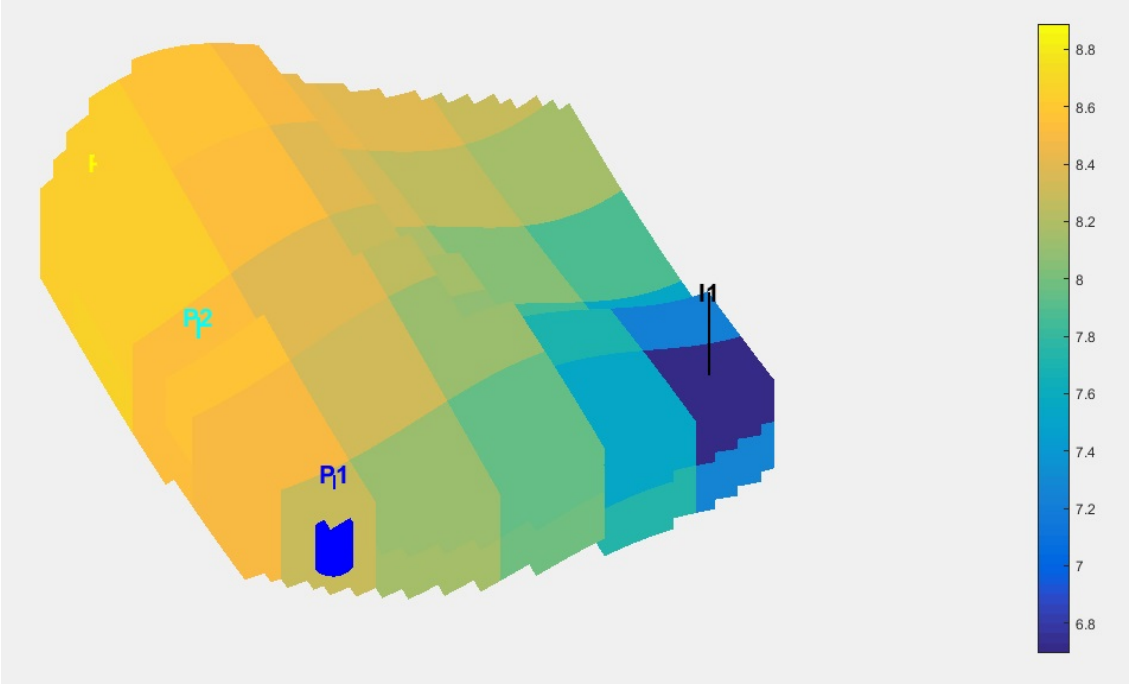


Figure 17: Plot of Tracer Volumetric Partitioning for flow based coarse Grid Model.

## 4. Results and Findings

After the model is constructed, we then visualize our findings to draw conclusions from them. We will begin by showing the initial reservoir state and consequently the initial bottom hole pressure.

The Blue marked regions indicate areas of 100% oil saturation; hence, we placed our production wells at the region, while the yellow region is that of 100% water saturation, at an initial state [15].

### *4.1. Bottom-hole Pressure*

The pressure of the wells (Injector and Producers) is depicted to give us an insight to the well pressure underground, as depicted in the Figures 5, 6, 7 and 8. Figures 9, 10 and 11 show the net reservoir state after the simulated period of 10 years.

### *4.2. Arrival times*

The question now is but the computational time is too long and may take longer if we were to make the model finer, breaking it into a greater number of grid cells [16] . We found an answer to that by creating a coarse grid with the same topology and geometry in the same domain and repeated all the steps taken above for the finer simulation model. Figure 9 shows solutions slightly different from the finer model but faster computational time (about 200 seconds 112 milliseconds) with less complexity. From this it can be deduced that the finer model analysis is more accurate but more computed in time whereas the coarser model takes less time but also less accurate than the finer model. Figures 10 and 11 show an overall picture of coarser model of reservoir dynamics.

Now that we have pressure and well solutions in both the base case and the coarser model, we computed the diagnostic quantities, i.e., the time of flight and tracer partition. The time-of-flight measures dynamic heterogeneity of the porous media, higher time show more complexity in flow field to be established quickly due to heterogeneity whereas establishing connections between well pairs requires tracing regions affected by anyone source or sink forces taken as a fraction of the total volume as volume partition (tracer volumetric partition) [17].

Figures 12, 13 and 14 show a qualitative description of residence time and evolution of multiphase displacement over a period of 10years for fine, coarse and coarser model representation with a color bar indicating the extent of residence time. Fluid element may find it easier to transverse through the fault line from I1 to P1, P2 then P3 which takes lesser time in that path than rather crossing across two intersecting faults as shown by the deep brown color while the yellow color between the production wells shows less flow complexity and heterogeneity and therefore, the region between them has a lucrative volumetric connection and can easily map out the sweep and drainage volumes just as required in fluid displacement efficiencies. I observed that such a connection is quite hard to depict if we upscale the physical model in the coarse model representation.

Figures 15, 16 and 17 show tracer volumetric partitioning representing the regions affected by different fluid elements reported as a returned value fraction of the total volume. For the finer model, the yellow color indicates regions of high volumetric connectivity

and efficient fluid displacement as a whole part of a partition while the blue shows less of such connections [18]. This effect is less accurately noticeable in the coarse models. To ramp everything up it has been clearly seen that the fine model is more accurate and gives more details but slower computational time than the coarser model.

## 5. Conclusion

Flow diagnostics in the context of this research has been used to get a clearer view of the flow occurring in the reservoir and offer answers to queries about where a specific producer drains and where an injector applies pressure support, the understanding of the communication between an injector and a producer, what region is affected by this communication and more [19]. For a fault bounded reservoir, we have established the flow field paths and pattern between arrays of wells separated by two intersecting faults and establish their well solutions as a function of time [20]. To improve and study computational speed and accuracy, we scaled up and presented a coarse model using the same approach to compare their efficiencies and validates.

It is therefore recommended that a simulation driven design and analysis of reservoir performance be fully adopted in assessing the viability of oil and gas business processes and projects, and recommend machine learning algorithms in data analytics for oil and gas projects and operations.

## Acknowledgment

The authors wish to acknowledge the effort made by the Department of Petroleum Engineering, Abubakar Tafawa Balewa University Bauchi, in Providing enabling environment for the conduct of this research.

## References

[1] S. Krogstad, K. A. Lie, O. Moyner, H. M. Nilsen, X. Raynaud & B. Skaestad, "MRST-AD an open-source framework for rapid prototyping and evaluation of reservoir simulation problems", In SPE Reservoir Simulation Symposium, 23-25 February, Houston, Texas, 2015. https://doi.org/10.2118/173317-MS

[2] H. Gross, M. Honarkhah, & Y. Chen, "Uncertainty modelling of a diverse portfolio of history-matched models with distance-based methods: application to an offshore gas condensate reservoir", In Proceedings of the SPE Asia Pacific Oil and Gas Conference & Exhibition, Jakarta, Indonesia, 2017, pp. 20-22. https://doi.org/10.1007/s11242-005-0619-7

[3] O. Andersen, H. M. Nilsen & X. Raynaud, "Coupled geomechanics and flow simulation on corner-point and polyhedral grids", SPE Reservoir Simulation Conference, Montgomery, Texas, USA, February 2017. https://doi.org/10.2118/182690-MS

[4] O. Moyner, S. Krogstad & K. A. Lie, "The application of flow diagnostics for reservoir management", SPE J. **20** (2014) 306. https://doi.org/10.2118/171557-PA

[5] J. E. Aarnes, S. Krogstad & K. A. Lie, "Multiscale mixed/mimetic methods on corner-point grids", Comput. Geosci. **12** (2008) 297. ISSN 1420-0597. https://doi.org/10.1007/s10596-007-9072-8

[6] M. H. Hui, R. Karimi-Fard, M. Mallison & L. J. Durlofsky, "A general modeling framework for simulating complex recovery processes in fractured reservoirs at different resolutions", SPE J. **22** (2018) 20. https://doi.org/10.2118/182621-MS

[7] M. Pal, S. Lamine, K. A. Lie & S. Krogstad, "Validation of the multiscale mixed finite-element method", Int. J. Numer. Meth. Fluids. **77** (2015) 206. https://doi.org/10.1002/fld.3978

[8] A. F. Rasmussen & K. A. Lie, "Discretization of flow diagnostics on stratigraphic and unstructured grids", In ECMOR XIV 14th European Conference on the Mathematics of Oil Recovery, Catania, Sicily, Italy, 8-11 September 2014, EAGE, 2014. https://doi.org/10.3997/2214-4609.20141844

[9] O. Andersen, H. M. Nilsen & X. Raynaud, "Virtual element method for geomechanical simulations of reservoir models", Comput. Geosci. **21** (2017) 877. https://doi.org/10.1007/s10596-017-9636-1

[10] Tomlab Optimization Inc. Matlab Automatic Differentiation (MAD). https://matlabad.com/ [Online; accessed July 11, 2018]

[11] P. Samier & R. Masson, "Implementation of a vertex-centered method inside an industrial reservoir simulator: Practical issues and comprehensive comparison with corner-point grids and perpendicular-bisector-grid models on a field case", SPE J. **22** (2017) 660. https://doi.org/10.2118/173309-PA

[12] J. M. Nordbotten, "Convergence of a cell-centered finite volume discretization for linear elasticity", SIAM J. Numer. Anal. **53** (2015) 2605. https://doi.org/10.1137/140972792

[13] J. M. Nordbotten, "Stable cell-centered finite volume discretization for biot equations", SIAM J. Numer. Anal. **54** (2016) 942. https://doi.org/10.1137/15M1014280

[14] H. M. Nilsen, K. A. Lie & J. R. Natvig, "Accurate modelling of faults by multipoint, mimetic, and mixed methods", SPE J. **17** (2012) 568. https://doi.org/10.2118/149690-PA

[15] H. M. Nilsen, J. M. Nordbotten & X. Raynaud, "Comparison between cell-centered and nodal based discretization schemes for linear elasticity", Comput. Geosci. **22** (2018) 233. https://doi.org/10.1007/s10596-017-9687-3

[16] O. Møyner & H. A. Tchelepi, "A multiscale restriction-smoothed basis method for compositional models", In SPE Reservoir Simulation Conference, 2017. https://doi.org/10.2118/182679-MS

[17] O. Møyner, "Nonlinear solver for three-phase transport problems based on approximate trust regions", Comput. Geosci. **21** (2017) 999. https://doi.org/10.1007/s10596-017-9660-1

[18] S. Krogstad, K. A. Lie, H. M. Nilsen, C. F. Berg & V. Kippe, "Efficient flow diagnostics proxies for polymer flooding", in Comput. Geosci. **21** (2017) 1203. https://doi.org/10.1007/s10596-017-9681-9

[19] Ø. S. Klemetsdal, R. L. Berge, K. A. Lie, H. M. Nilsen & O. Møyner, "Unstructured gridding and consistent discretizations for reservoirs with faults and complex wells", in SPE Reservoir Simulation Conference, 2017. https://doi.org/10.2118/182679-MS

[20] M. Karimi-Fard & L. J. Durlofsky, "A general gridding, discretization, and coarsening methodology for modeling flow in porous formations with discrete geological features", Adv. Water Resour. **96** (2016) 354. https://doi.org/10.1016/j.advwatres.2016.07.019

## APPENDIX A.

Some important codes from Matlab reservoir Simulation Toolbox Used in the Running of the Project

```
%% Reservoir geometry and petrophysical properties
% Define grid
grdecl = makeModel3([50, 50, 5], [1000, 1000, 5]*meter);
G = processGRDECL(grdecl);
G = computeGeometry(G);
% Set up permeability based on K-indices
[I, J, K] = gridLogicalIndices(G);
top = K < G.cartDims(3)/3;
lower = K > 2*G.cartDims(3)/3;
middle = ~ (lower | top);
px = ones (G.cells.num, 1);
px(lower) = 400*milli*darcy;
px(middle) = 200*milli*darcy;
px(top) = 800*milli*darcy;
% Introduce anisotropy by setting K_x = 10*K_z.
perm = [px, px, 0.1*px];
rock = makeRock(G, perm, 0.3);
% Plot horizontal permeability and wells
figure(1); clf
plotCellData(G, rock.perm(:, 1)/(milli*darcy))
view(50, 50), axis off
colorbar
title('K_x [mD]')
%% Define wells and simulation schedule
% Producers
pv= poreVolume(G, rock);
injRate = 1*sum(pv)/simTime; offset = 10;
W = verticalWell([], G, rock, offset, offset, [],...
'Name', 'P1', 'comp_i', [1 0], 'Val', 250*barsa, 'Type', 'bhp');
W = verticalWell(W, G, rock, offset,
floor(G.cartDims(1)/2)+3, [],
'Name', 'P2', 'comp_i', [1 0], 'Val', 250*barsa, 'Type', 'bhp');
W = verticalWell(W, G, rock, offset, G.cartDims(2) - offset/2, [], ...
'Name', 'P3', 'comp_i', [1 0], 'Val', 250*barsa, 'Type', 'bhp');
% Injectors
W = verticalWell(W, G, rock, G.cartDims(1)-5, offset, 1,...
'Name', 'I1', 'comp_i', [1 0], 'Val', injRate, 'Type', 'rate');
% Plot the wells
plotWell(G, W)
axis tight
% Three-phase template model
fluid = initSimpleADIFluid('mu', [1, 5, 0]*centi*poise, ...
'rho', [700, 1000, 0]*kilogram/meter^3, ...
'n', [2, 2, 0]);
% Constant oil compressibility c = 0.001/barsa;
p_ref = 300*barsa;
fluid.bO = @(p) exp((p - p_ref)*c);
clf
p0 = (100:10:500)*barsa;
plot(p0/barsa, fluid.bO(p0))
xlabel('Pressure (bar)')
ylabel('Ratio')
title('Reciprocal formation volume factor for oil (bO)')
% Construct reservoir model
```

```
gravity reset on
model = TwoPhaseOilWaterModel(G, rock, fluid);
```