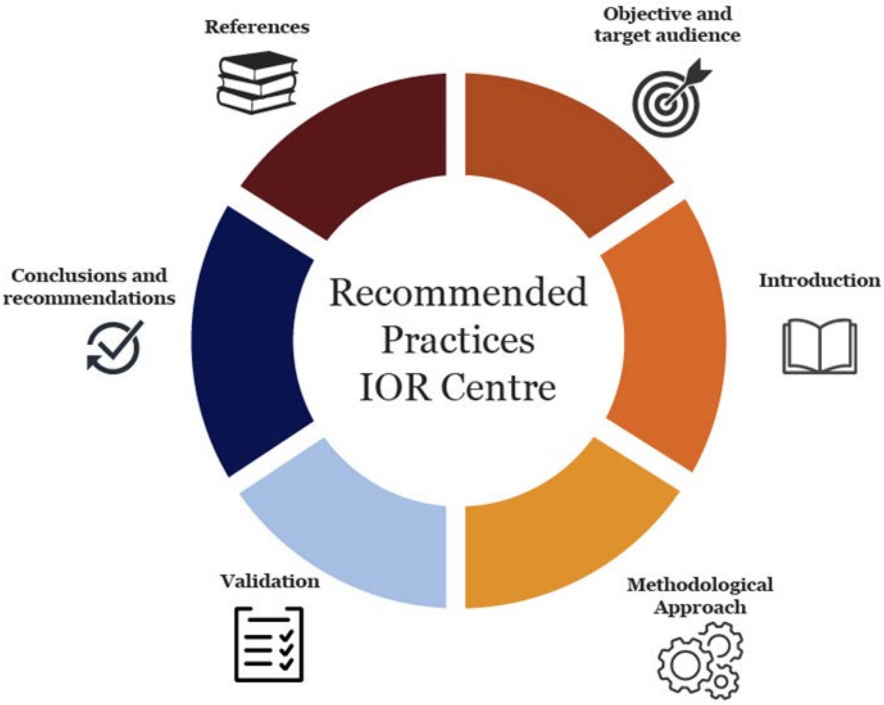
Olav Aursjø, Aksel Hiorth, Alexey Khrulenko, Oddbjørn Nødland

Polymer flooding

Simulation Upscaling Workflow

REPORT NO. 115, UNIVERSITY OF STAVANGER
30th NOVEMBER 2021


Recommended Practices IOR Centre
Objective and target audience
Introduction
Methodological Approach
Validation
Conclusions and recommendations
References

Reports from UiS

Polymer flooding - Simulation Upscaling Workflow

Olav Aursjø[1], Aksel Hiorth[2,1], Alexey Khrulenko[1] and Oddbjørn Nødland[2]

Rapport no.	115
Publisher	University of Stavanger
	www.uis.no
Project title	The National IOR Centre
ISBN	978-82-8439-099-4
ISSN (online)	2387-6662
ISSN (print)	0806-7031
DOI	https://doi.org/10.31265/usps.203

University of Stavanger

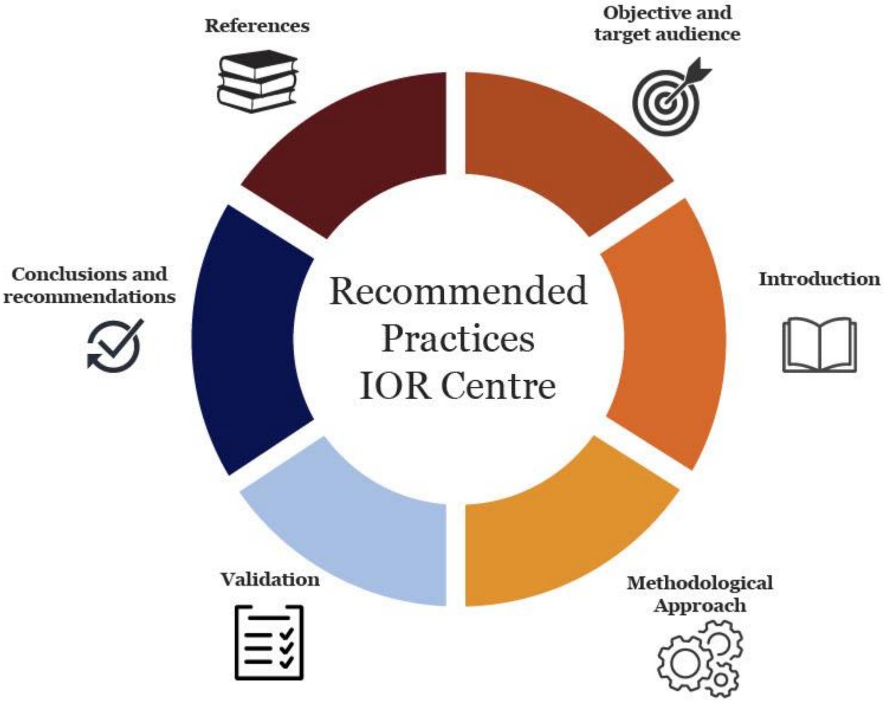
[1] NORCE Research, Norway

[2] University of Stavanger, Norway

The National IOR Centre of Norway

Polymer flooding

Simulation Upscaling Workflow


References
Objective and target audience
Conclusions and recommendations
Recommended Practices IOR Centre
Introduction
Validation
Methodological Approach

Olav Aursjø (NORCE), Aksel Hiorth (UiS, NORCE), Alexey Khrulenko (NORCE), Oddbjørn Nødland (UiS)

30th November 2021

Acknowledgement

The authors of recommended practices acknowledge the Research Council of Norway and the industry partners, ConocoPhillips Skandinavia AS, Aker BP ASA, Vår Energi AS, Equinor Energy AS, Neptune Energy Norge AS, Lundin Energy Norway AS, Halliburton AS, Schlumberger Norge AS, and Wintershall Dea Norge AS, of The National IOR Centre of Norway for support.

References
Objective and target audience
Conclusions and recommendations
Recommended Practices IOR Centre
Introduction
Validation
Methodological Approach

Contents

Objective and target audience

Several numerical models have been developed within the National IOR Centre of Norway (NIORC) to describe polymer flow in porous media at different length scales. To predict polymer behavior at the pore- and core scales, models must account for the essential polymer physics; in particular, the IORCoreSim simulator includes a state-of-the-art model for laboratory experiments (Lohne, Nødland, Stavland, & Hiorth, 2017) (Lohne A. , Stavland, Åsen, Aursjø, & Hiorth, 2021) (Stavland, Åsen, Lohne, Aursjø, & Hiorth, 2021).

The mechanisms observed at smaller scales may also be important at the reservoir scale. However, when constructing effective field scale models, it is not computationally feasible to carry over all features from the pore- and core-scale models.

This report addresses the following question: how can core scale simulation models, only found in IORCoreSim, be translated to the field scale? We present several, independent strategies for how to achieve this. The closing section, **Conclusions and recommendations,** sums up advantages and disadvantages with the proposed workflows.

The target audience is anyone who wants to reliably incorporate advanced polymer rheology into their field scale simulations. A prerequisite for using the workflows is having access to specific numerical simulators, several of which are included in the final deliveries of the NIORC (for details, see the **Introduction**).

Introduction

There are many issues to consider when implementing polymer flooding offshore. On the practical side one must handle large volumes of polymer in a cost-efficient manner, and it is crucial that the injected polymer solutions maintain their desired rheological properties during transit from surface facilities and into the reservoir. On the other hand, to predict polymer flow in the reservoir, one must conduct simulations to find out which of the mechanisms observed at the pore and core scales are important for field behavior.

This report focuses on theoretical aspects relevant for upscaling of polymer flooding. To this end, several numerical tools have been developed. In principle, the range of length scales covered by these tools is extremely wide: from the nm (10^{-9} m) to the mm (10^{-3} m) range, all the way up to the m and km range. However, practical limitations require the use of other tools as well, as described in the following paragraphs.

The simulator BADChIMP is a pore-scale computational fluid dynamics (CFD) solver based on the Lattice Boltzmann method. At the pore scale, fluid flow is described by classical laws of nature. To a large extent, pore scale simulations can therefore be viewed as numerical experiments, and they have great potential to foster understanding of the detailed physics of polymer flooding. While valid across length scales, pore scale models require a high numerical resolution, and, subsequently, large computational resources.

To model laboratory experiments, the NIORC has, through project 1.1.1 DOUCS, developed IORCoreSim. This simulator includes a comprehensive model for polymer rheological behavior (Lohne A. , Stavland, Åsen, Aursjø, & Hiorth, 2021). The model is valid at all continuum scales; however, the simulator implementation is not able to handle very large field cases, only smaller sector scale systems. To capture polymer behavior at the full field scale, simulators designed for that specific purpose must be used.

One practical problem is therefore: How can we utilize the state-of-the-art polymer model, only found in IORCoreSim, as a tool to decrease the uncertainty in full field forecasts? To address this question, we suggest several strategies for how to combine different numerical tools. In the **Methodological Approach** section, we briefly discuss the more general issue of linking different scales and simulators. In the **Validation** section, we present two case studies demonstrating the proposed strategies and workflows.

Methodological approach

Polymer flow may be simulated at different scales: the pore scale, the core scale, or the reservoir scale. Figure 1 illustrates how the different numerical simulators referred to in this report relate to each other with regards to length scales and level of physical details.

As stated in the introduction, the pore scale approach is the most fundamental: fluid flow is here governed by momentum and mass conservation equations derived directly from Newton's laws of motion, while the presence of the porous medium is accounted for by a set of boundary conditions. On the other hand, there is considerable overlap between core- and reservoir scale simulators. While these also rely on mass and momentum conservation equations, they are based on a *continuum* field description of multiphase fluid flow in a porous medium. In most cases, the extended Darcy law (Muskat & Meres, 1936) is used to link macroscopic quantities such as flow rates and pressure via the concept of fluid mobility. In addition, non-Newtonian polymer effects are typically introduced by making the aqueous phase viscosity a non-linear function of the velocity (or pressure gradient).

At the field scale, a large degree of uncertainty arises from geological uncertainty, i.e., in the choice of absolute permeability field and relative permeability curves, the latter which is required as input to capture multiphase flow behavior. Ultimately, all physical and chemical behavior observed at larger scales have their fundamental origin at the pore scale. To truly explain what is going on, one must therefore understand the dynamics of complex, cooperative pore scale mechanisms. However, at the present time it is only feasible to simulate dynamic multiphase flow systems of dimensions ~1 mm^3. Core flooding experiments, on the other hand, are typically conducted on ~7 cm long cores with a diameter of ~3 cm. This discrepancy in length scales means, for instance, that boundary conditions applied in laboratory experiments are not directly transferable to the pore scale. It is therefore imprudent to use results from pore scale simulations as direct input to continuum scale simulators. At least for the time being, the best use of pore scale models is to provide qualitative understanding of polymer behavior, which in turn can be used to improve the conceptual models implemented in continuum models.

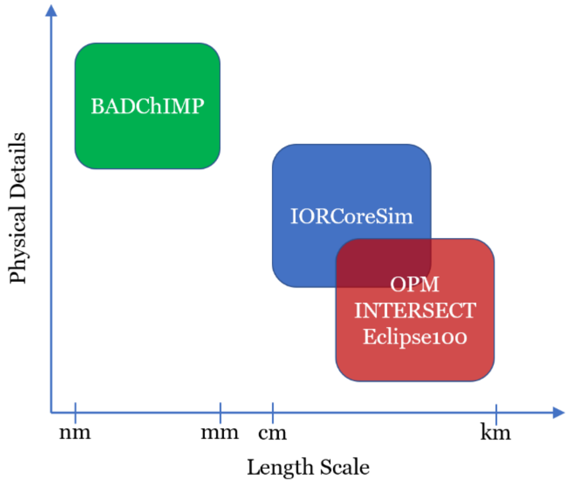


Figure 1 Schematic representation presenting the levels of physical details in the numerical simulators and at what length scales the simulators are applied.

The simulator IORCoreSim was originally developed as a tool to simulate laboratory experiments. The polymer models implemented in this simulator is based on simplified pore scale considerations; in particular, on the bundle of tubes concept. The model has been matched to experimental data for partially hydrolyzed polyacrylamide (HPAM) over a wide range of experimental conditions (Lohne A. , Stavland, Åsen, Aursjø, & Hiorth, 2021). It is this close agreement with experimental data, in combination with a physics-based modelling approach (as opposed to only using correlations), that makes us confident that IORCoreSim captures the essential polymer behavior for EOR applications. In the rest of this report, this conclusion will be taken as a premise.

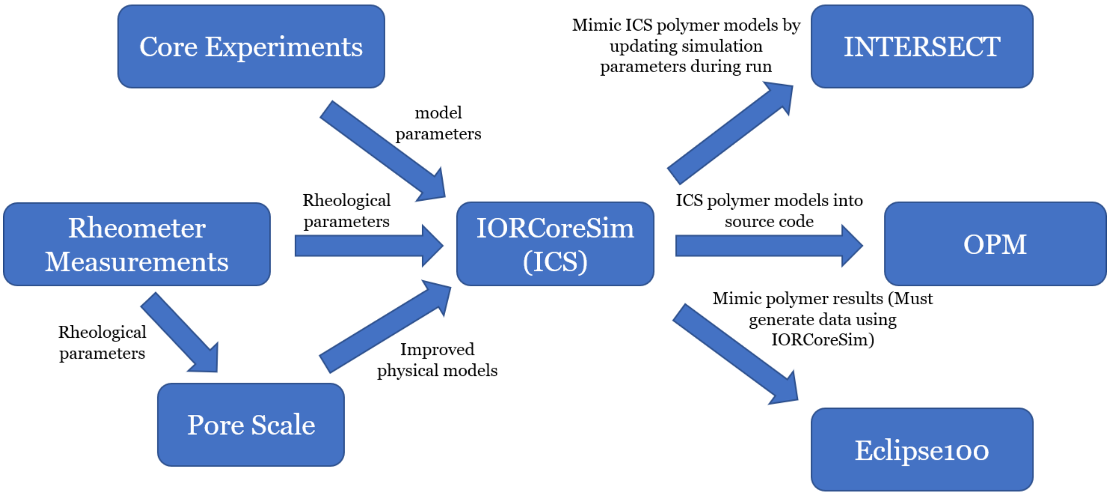


Figure 2 Representation of how the proposed polymer simulation workflows utilize experiments and the different numerical simulators.

Currently, IORCoresim has a limited ability to simulate large field scale cases. As a workaround, we have developed several workflows for how to approximate IORCoreSim polymer model behavior with the reservoir simulators OPM, INTERSECT, and E100. Figure 2 shows a schematical representation of how these simulators may be linked.

The choice of strategy depends heavily on available resources and know-how. If the source code is accessible, as is the case for OPM, and the required time, funding, and programming expertise exist, implementation of new polymer models into the code base is the preferred approach. On the other hand, if one chooses to use a commercial tool, constraints are obviously placed by the functionality present in the software. For some simulators, the only possible approach is to create appropriate input keywords and tables to the reservoir simulator before starting a simulation. In other cases, it is possible to modify selected reservoir grid block properties from outside the simulator *during* a run, for instance by writing Python scripts.

Regardless of the chosen approach, one should always account for numerical discretization errors when doing field scale simulations. This is because such simulations require the use of large grid blocks, which lowers the accuracy of the obtained results. One well-known issue is numerical dispersion, which is a consequence of truncation errors introduced as the governing partial differential equations are replaced by their finite difference or finite volume counterparts (Lantz, 1971). Even in the absence of non-Newtonian flow effects, numerical dispersion leads to an incorrect resolution of polymer fronts; this affects the calculation of viscosity as a function of polymer concentration. Numerical dispersion can in theory be removed by choosing small time steps, but for performance reasons this is not practical. For polymer flooding, a common alternative is to employ the Todd-Langstaff mixing model (Todd & Longstaff, 1972), which attempts to correct for dispersion at the front end of a polymer slug, and for viscous fingering at the rear end.

Special consideration should be given to near-well flow behavior. This is because fluid velocities near wells are underestimated by most simulators; again, due to poor numerical resolution. Since non-Newtonian flow effects depend strongly on velocity, forecasts of well injectivity made using coarse grids can be very different from predictions made with high resolution radial grids. This is especially important to consider when modelling synthetic polymers that exhibit shear-thickening and are prone to mechanical degradation. In fact, simulated reservoir shear rates are often so low that no mechanical degradation is predicted, even in scenarios where finely gridded models predict strong levels of degradation; to solve the IORCoreSim degradation equation accurately in radial flow, mm-sized grid blocks are required (Nødland, Lohne, Stavland, & Hiorth, 2019).

Clearly, it is not practical to use mm-sized grid blocks at the field scale, thus approximate models should be investigated. In IORCoreSim, a special well model partially accounts for the numerical error associated with coarse gridding (Nødland O. , 2021). Numerical integration is then used to compute an effective viscosity for each well block, as well as an approximate molecular weight to be transported into neighboring grid blocks. Elsewhere in the reservoir, shear rates are low and the impact of neglecting shear thickening or mechanical degradation is minimal.

Implementing new polymer models in the reservoir simulation source code

OPM

OPM comes with an implementation of the tabled-based E100 polymer model. In addition, the code base includes an experimental, and largely undocumented, feature in which an extra polymer component is introduced to track the polymer molecular weight. The molecular weight component is transported through the reservoir in the same way as the polymer mass concentration. This approach, also taken by IORCoreSim, makes it possible to compute the apparent viscosity of the polymer solution a function of the local rock and fluid properties.

In the current release version of OPM, the Huggins equation (Huggins, 1942) is used to capture effects of polymer concentration and molecular weight on the apparent viscosity. However, the model does not account for non-Newtonian flow. To improve the situation, some effort has been spent in translating the analytical shear thinning models from IORCoreSim to OPM, i.e., Meter's model (Meter & Bird, 1964) and the Carreau-Yasuda model (Yasuda, Armstrong, & Cohen, 1981). However, the shear thinning models have thus far only been applied to 1D cases, and the code is included in a *developmental* branch of OPM. More implementation and testing need to be done before it can become part of an official version (Nødland O. , 2021).

Incorporating IORCoreSim results in a reservoir simulator without access to source code

Reproducing IORCoreSim results with Eclipse100

IORCoreSim has a keyword-based input format similar to E100 (OPM). Some polymer keywords have direct equivalents in E100, but not all, which makes it hard to automatize the generation of input tables.

An alternative approach is to "history-match" E100 to IORCoreSim. A possible workflow is:

1. Apply IORCoreSim on a one-phase core flood, to produce reference simulation results that agree with experimental data.
2. Create an E100 model that matches the IORCoreSim lab-model. The input file created in step 1 must be translated, "keyword-by-keyword", into an analogous file for E100.
3. Extend the model to two phases. Simulations including an oil phase must be run both with both simulators, and the results should again match.
4. Set up sector- and field-scale models based on results from step 3.

A demonstration of this procedure is presented in the **Validation** section where results are compared with similar IORCoreSim results. A more thorough description is found in a separate NIORC project report (Khrulenko, 2021).

There is no direct method of modelling mechanical degradation in E100, but it is reasonable to assume that it only takes place within the well blocks (Nødland, Lohne, Stavland, & Hiorth, 2019). With the help of IORCoreSim, the following workflow is proposed:

1. Approximate the near wellbore region by a finely gridded IORCoreSim model in radial geometry.
2. Run the model and track the polymer properties in a specific grid block, located at a representative distance from the well.
3. Tune the injected polymer concentration in E100 to match the polymer solution viscosity in the grid block from step 2 (i.e., the alteration of polymer properties due to mechanical degradation is mimicked by an "effective" injected polymer concentration)
4. Match bottomhole pressures in E100 to IORCoreSim by means of skin-factor or productivity multipliers.

It must be noted that the resulting E100 model is valid only for a given set of well properties and operating constraints (e.g., injection rate and length of perforated well). It must be separately checked for other wells and injection rates and, if necessary, updated by following the same workflow.

Reproducing IORCoreSim behavior with INTERSECT 2020.1

The polymer model in INTERSECT is different from the one in E100: Meter's model (Meter & Bird, 1964) is used to calculate apparent viscosity. This model is also available in IORCoreSim, but in a more general form since the model coefficients do not have to be constant but can be made to depend on the local reservoir conditions, e.g., polymer molecular weight and polymer concentration.

As with the other simulators, it is important to be aware of how in-situ shear rate is computed in the reservoir simulator. The formula used by INTERSECT 2020.1 is very similar, but not identical, to the formula implemented in IORCoreSim. For instance, there are several variations in how velocity- and permeability-terms are averaged; for certain (rare) well configurations, the choice of formula greatly affects the simulation results, e.g., by incorrectly predicting the polymer to be in the Newtonian flow regime when it is not (Nødland O. , 2021).

To account for non-Newtonian flow effects near wells, INTERSECT comes with a special well block option: the user can input, for each well-cell connection, an effective radius from the well at which to calculate the well block shear rate. For shear thinning polymers, it is recommended to use this option. A workflow for comparing results between simulators could then be:

1. Select a set of reference conditions (molecular weight, temperature etc.).

2. Fit Meter's model with constant coefficients to the (more advanced) IORCoreSim model at the same conditions. Remember to account for differences in concentration units.

3. Select effective well block radii so that the corresponding well block viscosities matches results from the IORCoreSim well model.

Note that the effective radius option presupposes a constant flow rate at the well, but it may still give acceptable results when flow rates are changing in time, e.g., for constant bottom-hole pressure injectors. To check the sensitivity of near-well viscosity to flow rate variations, the INTERSECT model could be compared with an analogous IORCoreSim (sector) model.

Since Meter's model only captures the Newtonian and shear thinning flow regimes, it is currently difficult to account for shear thickening. Moreover, it is not possible to track local changes in polymer molecular weight in the reservoir, which makes it hard to account for (mechanical) degradation inside the reservoir. While INTERSECT 2020.1 requires the user to input a molecular weight for the polymer, it is kept constant after initialization and only used to convert between concentration units. Another issue is that the input molecular weight cannot exceed 10 MDa, which is lower than the molecular weight of commonly used EOR polymers.

An interesting possibility is raised by the fact that users can write custom Python scripts to extend the built-in simulator functionality: Is it possible to account for near-wellbore effects by modifying the well blocks from outside the simulator during a run? Unfortunately, the aqueous phase viscosity of a grid block cannot be modified after initialization in INTERSECT 2020.1. If this is changed in future versions, it would be worthwhile to try. As a possible workaround, attempts were made to capture the mobility reduction caused by polymer injection by lowering the relative permeability to water (i.e., instead of increasing viscosity). This approach worked reasonably well for cases with 100 % water saturation, but it was not successful for more relevant cases involving both water and oil (Nødland O. , 2021). In any event, the approach is not ideal because it could potentially affect other simulator options that are used in combination with polymer, e.g., smart water models that depend on changing relative permeability.

If degradation is important, we therefore suggest that the input polymer properties are selected from the start to represent those of the degraded polymer: The input polymer viscosity model can be selected based on results from sector-scale (radial) simulations in IORCoreSim. The resulting INTERSECT model will not capture the shear thickening effect, but at least it will reduce errors caused by assuming a too large polymer resistance factor inside the reservoir. As with the other proposed workflows, this method is expected to work best when well operating constraints are mostly the same throughout the period of polymer injection.

Validation

We present two case studies that can be used to validate some of the presented workflows:

1. We demonstrate the previously described E100 workflow. The aim of this study was to capture polymer flow behavior away from the near-well region, where mechanical degradation is negligible. For a full description, see (Khrulenko, 2021).
2. We exemplify how different model assumptions affect predictions made with IORCoreSim.

Reproducing IORCoreSim results with E100

First, a 1D IORCoreSim simulation model was set up, to be used as a reference for the “true” polymer properties. The model parameters were selected to match a single-phase core flood conducted in an outcrop sandstone with permeability ~2 Darcy and 23.5% porosity. This experiment was chosen since the permeability was in the 1 – 10 Darcy range, representative of the Statfjord Group of the Johan Sverdrup field, which is considered for piloting of polymer flooding (Eikje, et al., 2020). In the core flood, polymer solutions were injected at a wide range of flow rates, and the simulation model was able to reproduce all observed flow regimes, such as shear thinning and shear thickening (Lohne, Nødland, Stavland, & Hiorth, 2017).

The non-Newtonian rheology model in IORCoreSim was approximated in E100 with a PLYSHLOG table. Polymer adsorption and permeability reduction could be more directly accounted for, using the ADSORP keyword (assuming the Langmuir adsorption model). However, an extra set of simulations were conducted to verify the implementation of adsorption and inaccessible pore volume (IPV) effects. Moreover, input files to both simulators were modified to describe the injection of a polymer slug, at moderate injection rate, followed by its displacement by water (Figure 3a).

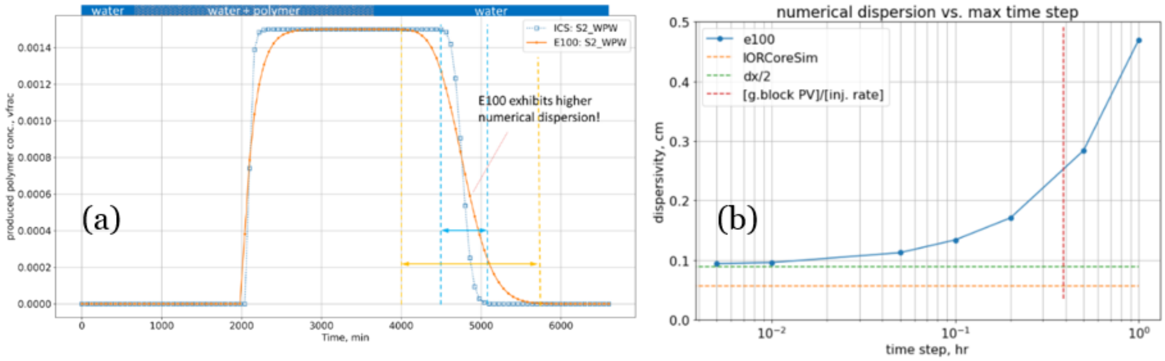


Figure 3(a) Simulation results showing the time evolution of produced polymer concentration after injecting a polymer slug in the form of a rectangular concentration pulse. The graphs show results from IORCoreSim (ICS) and Eclipse100 (E100) in blue and in orange, respectively. (b) Numerical dispersion coefficient as function of the maximally allowed time step in E100.

Physical dispersion was not included in these 1D simulations. Hence, any observed dispersion was solely due to numerical errors. Figure 3 shows that there was more numerical dispersion in E100 than in IORCoreSim. The level of numerical dispersion was quantified by an *equivalent dispersivity* (Brigham, 1974)[1]. As is readily apparent from Figure 3b, the amount of dispersion increased with the size of the input maximal time step, and a significant time step reduction was required to approach the theoretical dispersion of half a block length (Lantz, 1971). It is recommended to pay attention to this peculiarity of E100 when matching laboratory experiments.

[1] An effective (longitudinal) dispersion coefficient was fitted to the polymer effluent reported by the simulator, i.e., as if the smearing was caused by physical dispersion.

The 1D model was extended to two phases. Keyword values related to the IPV and the shear rate constant were tuned to ensure a good match between E100 and IORCoreSim. However, it must be remarked that there is a lack of relevant experimental data involving oil, and that IORCoreSim has not been directly calibrated to data for two-phase flow. Instead, the presence of oil is accounted for by making a few simple replacements in the equations developed for one-phase flow: porosity ϕ is replaced by $\phi \cdot S_\mathrm{w}$, while the absolute permeability k is replaced with the effective permeability to water, $k_\mathrm{w} = \mathrm{k} \cdot k_\mathrm{rw}$. All input polymer properties are assumed to be the same as before.

Next, the two-phase model was used to build a 1D sector scale model with a producer-injector pair at each end. A very good match between E100 and the twin IORCoreSim model was achieved for a 1 pore volume polymer slug injection (Khrulenko, 2021).

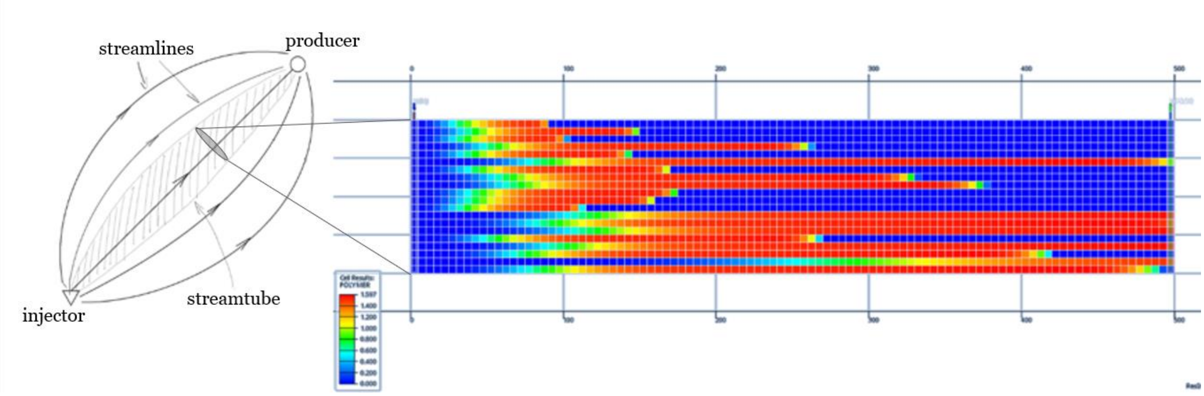


Figure 4 Schematic representation of the cross-sectional model concept

Finally, to test the workflow on a more realistic case, the 1D model was used to construct a set of 2D sector-scale models. Each model consisted of a cross-sectional layer-cake with no vertical communication (Figure 4). In effect, this is a mechanistic model of stream tubes between the producer-injector pair with 100% areal sweep efficiency and vertical heterogeneity. Several sets of model realizations with different degree of permeability variations, described through the Dykstra-Parsons coefficient V_dp, were generated. The models were run with both IORCoreSim and E100. It was observed that, while homogeneous and moderately heterogenous realizations matched relatively well, the mismatch between IORCoreSim and E100 increased with degree of heterogeneity (see Figure 5).

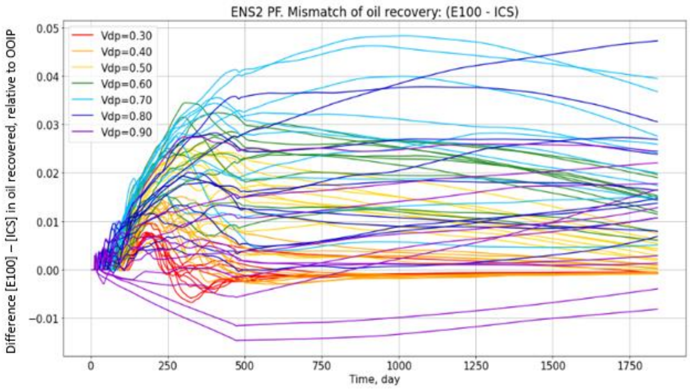


Figure 5 Plots showing the mismatch in oil recovery between E100 and IORCoreSim ([E100] - [ICS]) for an ensemble of model realizations.

These results may be understood as follows: fluid flow in heterogeneous reservoirs is dominated by high permeable layers, whereas low permeable layers tend to be plugged by the polymer. The strongly heterogenous flow systems deviate significantly from the homogenous 2-Darcy model that was used to construct the E100 model. Thus, using this 2-Darcy model for all layers leads to large discrepancies compared to IORCoreSim. A solution to this is to adjust each realization separately by specifying polymer properties for a range of permeabilities using the same workflow. This will help adjust how the different layers contribute to the flow due to their different permeabilities.

To summarize, it is possible to reproduce IORCoreSim behavior in E100 by using the standard polymer model. However, the approach has its limitations. It works relatively well for low and medium levels of heterogeneity, but for very heterogeneous reservoirs the match between the two simulators can be rather mediocre. Also, E100 core-scale models exhibit more numerical dispersion than IORCoreSim. The numerical dispersion depends on the maximum time step. It is recommended to limit the time step by 0.1 x [typical grid block PV]/[injection rate].

IORCoreSim case study in a channeled reservoir

For performance reasons we choose a relatively small field case: the first permeability realization of the so-called Egg model geological ensemble (Jansen, et al., 2014). The well placement is shown in Figure 6. The eight injectors and four producers are all vertical, and they are perforated throughout the entire formation thickness (28 m). For ease of interpretation, porosity is assumed uniform, φ = 0.2, and a constant initial water saturation is used in all active grid cells, $S_{wi} = 0.25$.

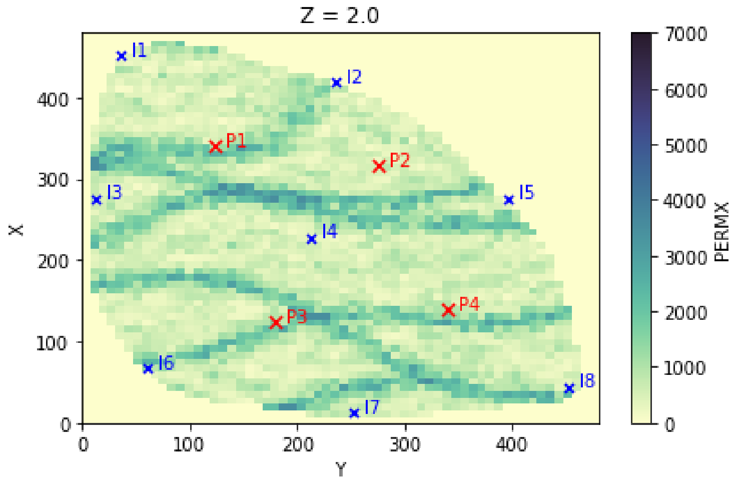


Figure 6 Horizontal permeability in top layer of the Egg model. There are 60x60x7 cells, of which 18553 are active. Grid block sizes are 8x8x4 meter. The seven layers have a strong vertical correlation, such that the permeability field is almost 2-dimensional (82 mD-7 D, arithmetic average 1.12 D). The ratio of horizontal to vertical permeability is ~10.

We consider injection of synthetic sea water (SSW) into a reservoir filled with oil and, for simplicity, SSW. The temperature is assumed constant, T=20°C. The water viscosity is assumed to be 1.07 mPas, while the oil viscosity is 20.0 mPas. The oil density is 900 kg/m^3. Capillary pressure is ignored in the simulations. A set of generic relative permeability curves, representative of mixed-wet formations, is used.

The total simulated time is 10 years. Water is injected at constant rate 79.5 m3/day with an upper bottom-hole pressure limit of 271 bar, which is close to the estimated fracturing pressure when assuming a fracture pressure gradient of ~0.14 bar/m. All producers are set to operate at a constant bottom-hole pressure of 195 bar. For more details on the simulation setup, see (Nødland O. , 2021).

We first simulate a baseline waterflood, followed by a “screening run” in which the effects of polymer are approximated by injecting a Newtonian fluid of viscosity 10 mPas, about 10 times as viscous as water. For the screening run, as well as for simulations with actual polymer, injection of viscous brine is commenced after 5 years (1827 days) and terminated after 6 years (2192 days); a slug of approximately ¼ pore volumes. The injected polymer concentration is always 1500 ppm. We consider two different HPAM polymers: FLOPAAM 3530S, with an initial molecular weight 15 MDa, and FLOPAAM 3630S, with molecular weight 20 MDa. The input polymer properties are taken from (Lohne, Nødland, Stavland, & Hiorth, 2017), with some notable exceptions; see discussion below.

For waterflooding with SSW, the absolute oil recovery is 47.9 % OOIP after 10 years. This number increases to 51.8 % when injecting the 10 mPas Newtonian fluid, to 51.1 % when injecting the 3530S polymer, and to 52.4 % when injecting 3630S. The left plot of Figure 7 shows the *relative increase* of oil production as a function of time. For both polymers, most of the incremental oil is produced after commencing the second waterflood. For 3530S, the Newtonian model gives the same recovery as the polymer model during the first year of post-polymer water injection. In contrast, the response is much quicker for the 3630S polymer, which is no surprise considering the larger molecular weight of this polymer.

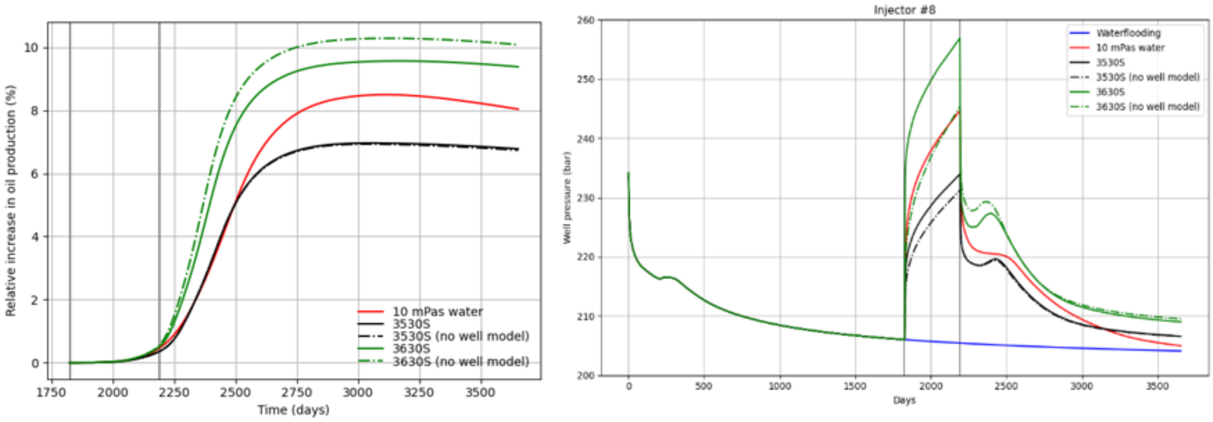


Figure 7 Left plot: Simulated (total) oil production as a function of time for two HPAM polymers and a Newtonian fluid. The y-axis shows the relative increase in oil recovery compared to the baseline waterflood. The grey vertical bars show the timing of the polymer slug. The dashed lines show results in which the IORCoreSim well model was not used. Right plot: Bottom-hole pressure at one of the injection wells.

Obviously, the additional flow resistance of 3630S compared to 3530S necessitates higher injection pressures to keep a prescribed injection rate. Before injecting HPAM polymer solutions, one should check whether shear thickening could cause problems; if this is done by conducting simulations with IORCoreSim, it is important to use the well model which corrects for non-Newtonian flow and mechanical degradation close to the wellbore. The right plot of Figure 7 indicates that injectivity could become an issue for the 3630S polymer, especially if no fracturing of the formation is allowed.

The oil recovery curves suggest that the polymer is not severely degraded in the scenario considered here. Indeed, separate calculations in radial geometry indicate that the 3530S polymer will experience very little, if any, degradation. This number is based on analytical calculations for steady-state, single-phase flow without any inaccessible pore volume, depletion layers, or permeability reduction effects (Nødland, Lohne, Stavland, & Hiorth, 2019). Even if we account for these effects, simulations in finely gridded radial geometry indicate that the conclusion still holds. On the other hand, the 3630S polymer is expected to undergo some degradation upon entering the reservoir, but no more than ~20 % in the least permeable injection zones. Since most of the injected polymer is allocated to high permeability layers, this number is likely pessimistic, especially considering that degradation levels are very low around most of the injectors, and that polymer from adjacent layers mix due to crossflow. To summarize, shear thickening and degradation is expected to influence the results to some extent for 3630S, and much less so for 3530S. Of course, this conclusion might no longer be true if the well operating constraints are changed.

The right plot of Figure 7 further highlights the fact that for non-Newtonian fluids, calculated well pressures are affected by the coarse gridding. This error, which occurs for both shear thinning and shear thickening polymers, is usually exacerbated when increasing the grid block dimensions (Li & Delshad, 2014), as one would do in a realistic offshore field setting.

If we disregard near-well behavior, the polymer solutions behave as either near-Newtonian or shear thinning fluids. However, simulations of polymer flooding do not just depend on viscosity behavior; predictions are also sensitive to assumptions made about polymer adsorption and depletion layers; a form of inaccessible pore volume. In some respects, these are competing effects, but there is one crucial difference: adsorption reduces the permeability of water. Potentially, this could have a large positive impact on reservoir sweep in the post-polymer period.

In the simulations presented so far, the maximum level of adsorption was set ~5 times lower than values matched to laboratory experiments in water-wet outcrop Berea and Bentheimer cores (Lohne, Nødland, Stavland, & Hiorth, 2017). Slight adjustments were also made to the depletion layer model, and the tortuosity factor was changed to depend on porosity and water saturation via Archie's law (Archie, 1942). If we instead use the originally reported input values for the adsorption and depletion layer models, the polymer is delayed at the producers, produced polymer concentrations are lower, but the predicted EOR response is greater (Figure 8).

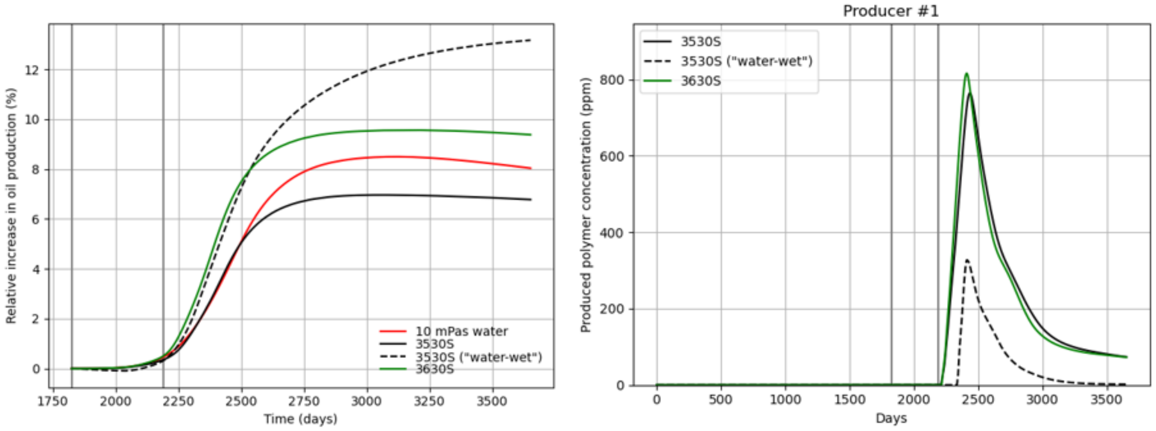


Figure 8 Left plot: Oil recovery results from Figure 7 compared with a new simulation for 3530S with higher levels of adsorption and permeability reduction, plus the original depletion layer model from (Lohne, Nødland, Stavland, & Hiorth, 2017). Right plot: Corresponding polymer production curves at one of the producers.

This is because the "water-wet" model predicts significantly higher permeability reduction than the "mixed-wet" model, and more variability inside the reservoir. The polymer front moves more slowly, but the level of mobility reduction is increased. Moreover, since oil permeability is assumed unaffected by polymer adsorption, reservoir sweep is improved in the second waterflood. Once again, higher resistance factors imply lower injectivity; for this particular simulation, some of the injectors had to switch from being rate-controlled to being pressure-controlled during polymer injection (not shown).

A very high level of adsorption may not be realistic for reservoirs that are candidates for polymer flooding, but the simulation results demonstrate that permeability reduction is an important effect to consider in heterogeneous reservoirs. Here, it has a positive impact on oil recovery, but in other circumstances it is going to be detrimental.

Conclusions and recommendations

During the initial screening phase of a polymer project, a simple Newtonian fluid can be used as a proxy for the polymer. The Newtonian model gives a baseline expectation for whether polymer injection can improve oil recovery, under the assumption that the dominating mechanism is an increase in the viscosity of the aqueous phase. However, other effects may also be important, e.g., non-Newtonian flow, permeability reduction, and inaccessible pore volume. The relative importance of each mechanism is field-specific, and depends on factors such as reservoir heterogeneity, temperature, salinity, oil viscosity, well placement, and well operating constraints. After the screening phase, we therefore recommend that all polymer mechanisms observed at the core scale be evaluated with sector scale simulations. To this end, IORCoreSim is a very useful tool.

To constrain IORCoreSim model input, two-phase core scale experiments should be performed at representative wettability conditions. Polymer adsorption is important to characterize. In the simulation case study presented in this report, adsorption was assumed instantaneous and irreversible. More work should be carried out to evaluate effects of reversible adsorption, especially regarding the performance of post-polymer water injection. Furthermore, the impact of numerical dispersion and viscous fingering should be evaluated, as well as effects of varying temperature and salinity. There are models to account for salinity and temperature effects in IORCoreSim, but they should be subjected to testing, and potentially developed further (Lohne A. , Stavland, Åsen, Aursjø, & Hiorth, 2021).

Since IORCoreSim cannot run very large field cases, two commercial simulators (E100 and INTERSECT) and one open-source tool (OPM) were evaluated for their potential to simulate polymer flooding. It was not straightforward to account for shear thickening and mechanical degradation when using these tools, but several workarounds were discussed. For the case of INTERSECT, attempts were made to capture shear thickening by modifying grid block properties from outside the simulator with Python scripts. The Python-scripting approach did not work well in realistic field settings, but it might be useful in future versions of the simulator; provided the user is allowed to change the aqueous phase viscosity during a simulation run (this is not possible in the 2020.1 version). As for mechanical degradation, the best alternative, for now, seems to be to inject a polymer having the properties of a degraded polymer.

Another limitation with INTERSECT 2020.1 is that only isothermal flow is supported when doing polymer flooding simulations. If this is not a concern, the simulator seems to be a good choice for modelling polymers that are in either the Newtonian or shear thinning flow regime, e.g., biopolymers. Arguably, the analytical shear thinning option in INTERSECT is easier to use than the table-based model of E100 / OPM. The simulator also has a special option to handle shear thinning effects near wells, which can reduce numerical errors in well injectivity calculations.

The polymer model in E100 does not presume isothermal flow, which is an advantage. Another reason for preferring E100 is that it is an “industry-standard tool”, and therefore available to many. When using E100 to reproduce experimental core flooding data for HPAM, multiple IORCoreSim simulations must be conducted to generate the necessary input tables. Moreover, matching E100 to a single permeability could lead to a non-negligible discrepancy between E100 and benchmarking results from IORCoreSim. For heterogeneous reservoirs, it may therefore be necessary to generate many input tables, covering a range of permeabilities, which is both time-consuming and more error prone than using a simple analytical model.

The robustness of the E100 model conversion workflow should be investigated by applying it to more cases. In principle, the same workflow can be used in OPM, however it is our experience that using E100 leads to fewer convergence problems when using the table based

PLYSHLOG model. The main rationale for using OPM is that users have access to the source code, and thus full control over the models that are implemented. With sufficient coding and testing, there would be no need for any workarounds or "hacks".

Finally, we remark that this report has focused exclusively on HPAM polymers. Some work has previously been done to model associative polymers with IORCoreSim (Lohne, Stavland, & Reichenbach-Klinke, Modeling of Associative Polymer Flow in Porous Medium, Vol. 2019, No. 1). To numerically incorporate effects of hydrophobic chemical groups on the polymer backbone, an extra polymer component was added to the model, and the total resistance factor was made to depend on both polymer components. Typically, open source or commercial tools do not handle the mixing of several polymer types (e.g., neither OPM nor INTERSECT allows this). It is therefore anticipated that associative polymers will be harder to model accurately with such tools.

References

Archie, G. E. (1942). The electrical resistivity log as an aid in determining some reservoir characteristics. *Transactions of the AIME, 146(01)*, 54-62.

Brigham, W. (1974). Mixing Equations in Short Laboratory Cores. *SPE J. 14 (01)*, 91–99. doi:https://doi.org/10.2118/4256-PA

Eikje, E., Nedrelid, T., Bratli, E., Kulkarni, R., Fylling, A. E., Lyse, O. A., & Aarrestad, H. D. (2020). Towards A Recovery Ambition Of "More Than 70%" For The Johan Sverdrup Field. *paper presented at the Offshore Technology Conference, Houston, Texas, USA, May 2020*. doi:https://doi.org/10.4043/30525-MS

Huggins, M. L. (1942). The viscosity of dilute solutions of long-chain molecules. IV. Dependence on concentration. *Journal of the American Chemical Society, 64(11)*, 2716-2718.

Jansen, J. D., Fonseca, R. M., Kahrobaei, S., Siraj, M. M., Van Essen, G. M., & Van den Hof, P. M. (2014). The Egg model – A geological ensemble for reservoir simulation. *Geoscience Data Journal, 1(2)*, 192-195.

Khrulenko, A. (2021). *Translation of IORCoreSim polymer model into Eclipse. Report part of Project 2.7.1.* Stavanger, Norway: The National IOR Centre of Norway.

Lantz, R. B. (1971). Quantitative evaluation of numerical diffusion (truncation error). *Society of Petroleum Engineers Journal, 11(03)*, 315-320.

Li, Z., & Delshad, M. (2014). Development of an analytical injectivity model for non-Newtonian polymer solutions. *SPE Journal, 19(03)*, 381-389.

Lohne, A. (2020). *User's manual for IORCoreSim - combined EOR and SCAL simulator (Version 1.311)*. NORCE.

Lohne, A., Nødland, O., Stavland, A., & Hiorth, A. (2017). A model for non-Newtonian flow in porous media at different flow regimes. *Comput Geosci 21*, 1289–1312. doi:https://doi.org/10.1007/s10596-017-9692-6

Lohne, A., Stavland, A., & Reichenbach-Klinke, R. (Vol. 2019, No. 1). Modeling of Associative Polymer Flow in Porous Medium. *IOR 2019–20th European Symposium on Improved Oil Recovery* (pp. 1-28). European Association of Geoscientists & Engineers.

Lohne, A., Stavland, A., Åsen, S. M., Aursjø, O., & Hiorth, A. (2021). *Recommended polymer workflow - Interpretation and parameter identification.* Stavanger, Norway: The National IOR Centre of Norway.

Meter, D. M., & Bird, R. B. (1964). Tube flow of non-Newtonian polymer solutions: PART I. Laminar flow and rheological models. *AIChE Journal, 10*, 878-881.

Muskat, M., & Meres, M. W. (1936, 7(9)). The flow of heterogeneous fluids through porous media. *Physics, 7(9)*, 346-363.

Nødland, O. (2021). *Project report 1.1.15*. Stavanger, Norway: The National IOR Centre of Norway.

Nødland, O., Lohne, A., Stavland, A., & Hiorth, A. (2019). An investigation of polymer mechanical degradation in radial well geometry. *Transport in Porous Media, 128(1)*, 1-27.

Schlumberger. (2020a). Eclipse Version 2020.2. Reference Manual.

Schlumberger. (2020b). Eclipse Version 2020.2. Technical description.

Stavland, A., Åsen, S. M., Lohne, A., Aursjø, O., & Hiorth, A. (2021). *Recommended polymer workflow - Lab (cm and m scale).* Stavanger, Norway: The National IOR Centre of Norway.

Todd, M. R., & Longstaff, W. J. (1972). The development, testing, and application of a numerical simulator for predicting miscible flood performance. *Journal of Petroleum Technology, 24(07)*, 874-882.

Yasuda, K. Y., Armstrong, R. C., & Cohen, R. E. (1981). Shear flow properties of concentrated solutions of linear and star branched polystyrenes. *Rheologica Acta, 20*, 163-178.