Upstream

Stimulating collaborative research through **open-source sharing of methodology**

Reservoir modelling is an essential tool in subsurface engineering activities ranging from hydrocarbon recovery to underground storage of carbon dioxide (CO_2) and hydrogen. It relies on inputs which, in addition to parameters such as porosity and permeability, include flow functions for multiphase flow, such as relative permeability and capillary pressure. These are typically obtained from core flooding experiments in special core analysis (SCAL), which consists of laboratory-based flow experiments to measure flow rate and pressure drop, and a subsequent interpretation step involving inverse modelling. However, robust first-principles inverse modelling tools are not easily accessible, and therefore Shell's partners do not always follow best-practice methodology for interpretation. To provide access to appropriate interpretation methodology that follows fundamental principles, an internally developed set of Python code, which was already available within Shell, has been made open source.

Open-source sharing of code enables Shell partners to interpret experimental results using best-practice methodology, for accurate and consistent results. (Image created with Microsoft Copilot.)

Interpretation of core flooding experiments by inverse modelling

Over the past 30 years, Shell has developed a rigorous first-principles methodology, based on numerical interpretation by the inverse modelling of core flooding experiments, which provides accurate and consistent results. However, not all Shell's industrial and academic partners have access to the relevant tools or full awareness of the shortcomings of more simplified interpretation methods, such as closed-form analytical expressions. There is a general lack of awareness of exactly how many and what types of data are required for an adequate interpretation of such experiments, that is, what data are required to properly constrain an inverse model.

This is illustrated in the example shown in Figure 1, which reflects a situation sometimes encountered in the literature or from external parties, where both relative permeability and capillary pressure are determined from a single unsteady-state core flooding experiment. This is not recommended, according to Shell SCAL best practices. Instead, steady-state core flooding experiments and the independent measurement of capillary pressure by, for example, the centrifuge method would be recommended [Ref 1, 2]. However, external parties may not follow Shell recommendations, because of either a lack of instrumentation, for example, an ultracentrifuge to independently measure capillary pressure, or other experimental considerations, such as, for CO_2 sequestration, obtaining a consistent data set for relative permeability and capillary pressure measured with, for example, supercritical CO_2, which is impossible or extremely difficult to do in a centrifuge.

The example in Figure 1, which is described in more detail in Reference 3, illustrates that such a workflow can result in an acceptable interpretation with an acceptable uncertainty range only under very specific conditions, and only if the inverse model is properly constrained by adequate experimental data. If only pressure drop and a production curve obtained at a single flow rate are used, the interpretation-based uncertainty (shaded region) in the so-obtained relative permeability is unacceptably high and significantly larger than the experimental error, indicated by error bars. Only when adding in-situ-measured saturation profiles along the core, and conducting the experiment at multiple flow rates, does the interpretation-related uncertainty arrive at a similar range to the experimental error [Ref 3].

Although the flow code is equivalent to best practice at Shell, the assisted inverse modelling workflow, which includes the flow code and optimisation methods, goes beyond the best-practice standard workflow recommended in special core analysis at Shell. In the standard workflow, relative permeability is only interpreted in the saturation range after breakthrough of the displaced phase. This is a physical constraint for

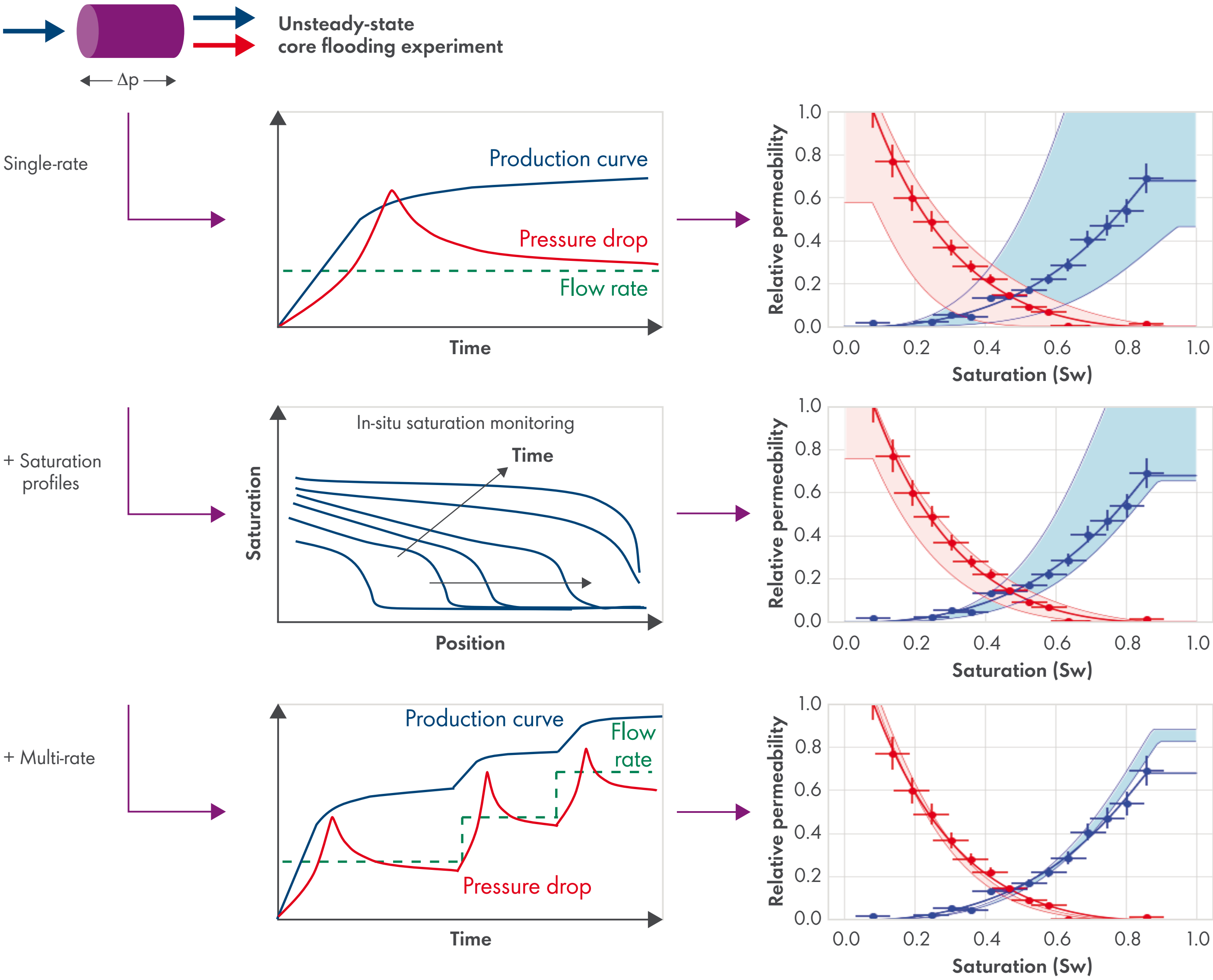

FIGURE 1
Advanced interpretation of an unsteady-state experiment aimed at obtaining relative permeability and capillary pressure functions simultaneously.

cases where the displacement front is very sharp, that is, in the viscous limit (Buckley–Leverett shock front), but for situations at lower capillary numbers the displacement front is more elongated and controlled by both relative permeability and capillary pressure, with a saturation range accessible by in-situ saturation monitoring.

Using the saturation profiles in the inverse modelling workflow also provides access to the saturation range before the Buckley–Leverett breakthrough. The main benefit of the methodology is that, by using analytical functions in the inverse model, the uncertainty range of the fit parameters is also obtained. The example in Figure 1 demonstrates that the more data are used, the better the inverse model is constrained and the smaller the uncertainty envelope. It may not be entirely obvious why a bump flood – traditionally used to extend the saturation range closer to the residual saturation – also reduces the uncertainty in the saturation range below that. This is because, via the capillary end-effect – which is mapped out better in the bump floods – the data constrain the capillary pressure-saturation function better.

The high degree of coupling and nonlinearity of the multiphase flow equations also enable the reduction of uncertainty in saturation ranges that depend on both capillary pressure and relative permeability, such as the profile of the displacement front before breakthrough; therefore, the uncertainty of relative permeability is reduced.

Additionally, the interpretation methodology is flexible in terms of exactly which measured data are available. If, for instance, as Shell best practice recommends, the relative permeability at the saturation endpoint is measured experimentally, then these data can be used as a constraint and the respective endpoint saturation and endpoint relative permeability are fixed. If, however, this data point is not measured, it can be allowed to vary in the inverse modelling workflow, which – while not recommended – still allows an interpretation of the data, although with the consequence of increased uncertainty.

Going open source

Because of a lack of awareness of how large the actual uncertainty range of an improperly ▸▸▸

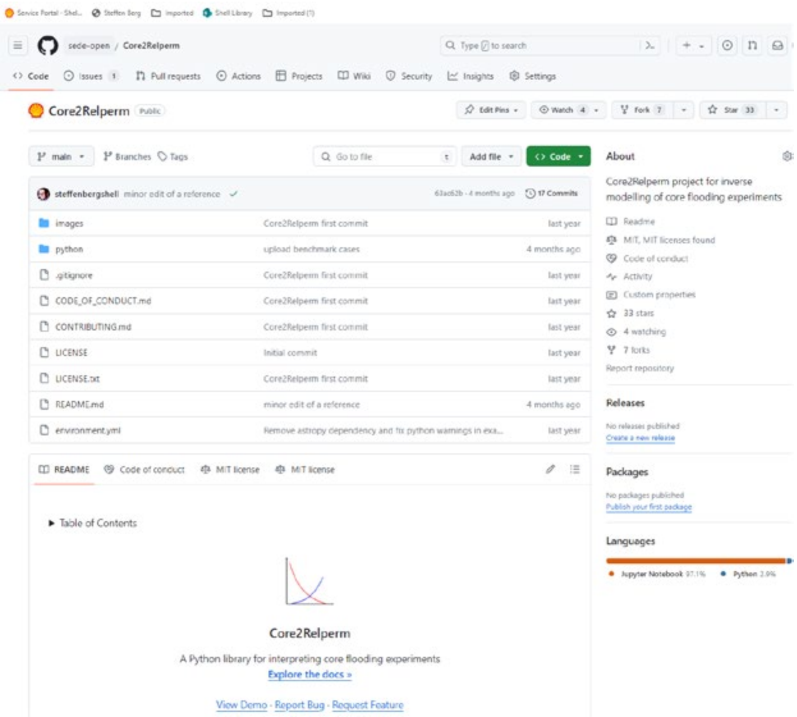

FIGURE 2
Screenshot of the open-source Core2Relperm project on sede-open.

constrained inversion workflow can be, and a lack of access to appropriate interpretation tools, external parties and partners may use inadequate methodology, which may lead to an incorrect interpretation and underestimation of uncertainties. This can become a problem in joint projects where, for instance, partners do not follow Shell best practices, which makes the resulting data more difficult to use within Shell. Therefore, it is in Shell's interest to make reliable methods, which are first-principles-based and have already been published for many years, readily accessible.

The Python code for inverse modelling of core flooding experiments – which had already been published in a set of publications [Ref 1, 2, 4], presented in a previous *TechXplorer Digest* article [Ref 5] and was accessible as the Core2Relperm project in Shell's innerSource GitHub sede-x – has now been made accessible as an open-source project in the publicly accessible sede-open platform, following Shell's OpenSource release process. This code is available at github.com/sede-open/Core2Relperm, and a screenshot of the location is shown in Figure 2.

The open-source release is accompanied by a publication in a peer-reviewed academic journal [Ref 3] that explains the function of the code and provides details of the example shown in Figure 1. Respective Jupyter Notebooks reproduce two key figures from the journal publication.

Reaction of the community

After the initial release of the code, an announcement on LinkedIn received more than 10,000 impressions within a week and reached 20,000 impressions a few weeks later. The project has been forked multiple times and the code is already actively used in PhD projects. The community began interacting with the code; 33 users saved it as a "starred" repository, in addition to creating several forks and reporting issues, which the team successively addressed. Some revealed issues and subtle details in the underlying Python libraries. In the meantime, the release was extended by adding more benchmarks for the flow solver.

Summary and conclusions

The authors have provided a useful tool which has been well received in the community. The open-source format addresses a specific aim, from the Shell side, for partners to provide results consistent with the Shell recommended interpretation workflows. They have also successfully demonstrated the open-source release route, according to the Shell process. It is expected that, in the future, this will become relevant more often, as more and more journals adopt the requirement to make code used in journal articles publicly available.

Next steps

As a next step, the team plans to provide a more parallelised version of the Markov Chain Monte Carlo (MCMC) optimiser. The current version uses a gradient-based optimiser, which runs in seconds, whereas a full MCMC with 10,000 iterations would require several hours.

This project is a great example of an open-source release from Shell adhering to the following principles:

1. **Transparency:** whether developing software or solving a business problem, everyone has access to the information and materials necessary for doing their best work. When these materials are accessible, people can build upon each other's ideas and discoveries, to make better decisions and understand their effects.
2. **Collaboration:** when people are free to participate, they enhance each other's work in unanticipated ways. When they can modify what others have shared, they unlock new possibilities. By initiating new projects together, solutions are possible to problems that no one can solve alone. And, when open standards are implemented, it enables others to contribute in the future.
3. **Inclusive meritocracy:** good ideas can come from anywhere, and the best ideas should win. Only by including diverse perspectives in conversations can decision-makers be certain they've identified the best ideas.
4. **Release early and often:** rapid prototypes can lead to rapid discoveries. An iterative approach – "learning by doing" – leads to better solutions faster. When people are free to experiment, they can look at problems in new ways and seek answers in new places.
5. **Community:** communities form when different people unite around a common purpose. Shared values guide decision-making, and community goals supersede individual interests and agendas.

While this interpretation methodology is strictly one-dimensional (1D), that is, it does not take into account unstable displacement and/or channelling in heterogeneity – which should be excluded by the design of the experiment and upfront inspection of the sample – it can, in principle, be extended to a 3D interpretation. This is subject to ongoing research and not yet at a stage where an automated workflow could be considered.

Acknowledgements
Luud Heck, Bryce Bartmann, Sherin Mirza, Rene-Edouard Plessix and Paul Gelderblom are acknowledged gratefully for their support with the GitHub release. The authors would also like to thank Harm Dijk for development of the code and the flow solver and for providing in-depth support with solving various computations and coding issues. Evren Unsal is gratefully acknowledged for contributions across the whole project, from conceptualisation through to publications and, in particular, opening options during the Shell open-source process. Bochao Zhao, Ronny Hofmann and Yingxue Wang are gratefully acknowledged for improving the code, sharpening its application and reviewing the application to Special Core Analysis. Justin Freeman and Miranda Mooijer are gratefully acknowledged for support in multiple dimensions. ■

Review:

Yingxue Wang, Principal Technical Expert, Special Core Analysis (SCAL)
Bryce Bartmann, Shell Chief Digital Technology Advisor

This content follows Shell's net carbon footprint and net-zero emissions target principles as documented here: www.shell.com/investors/disclaimer-and-cautionary-note.html.

References

[Ref 1] Berg, S. and Dijk, H.: "Uncertainty quantification of relative permeability: Assisted history matching of special core analysis experiments," Shell Report SR. 20.00994 (2020)

[Ref 2] Berg, S., Unsal, E. and Dijk, H.: "Non-uniqueness and uncertainty quantification of relative permeability measurements by inverse modelling," *Computers and Geotechnics* (2021) 132, 103964

[Ref 3] Berg, S., Dijk, H., Unsal, E., et al.: "Simultaneous determination of relative permeability and capillary pressure from an unsteady-state core flooding experiment?" *Computers and Geotechnics* (2024) 168, 106091

[Ref 4] Berg, S., Unsal, E. and Dijk, H.: "Sensitivity and uncertainty analysis for parameterization of multiphase flow models," *Transport in Porous Media* (2021) 140(1), 27–57

[Ref 5] Berg, S., Unsal, E. and Dijk, H.: "The certainty in uncertainty: Quantifying coreflood data errors," TechXplorer Digest (2021), 20–26

Authors

Steffen Berg, Principal Science Expert in the Special Core Analysis team in Amsterdam, is a subject matter expert in SCAL. He holds a master's degree in materials science from Saarland University and a PhD in physics from the University of Mainz/Max Planck Institute for Polymer Research at Mainz, Germany. Following postdoctoral research at Princeton University, he joined Shell in 2005 as a research scientist. He is currently also a visiting reader at Imperial College London.

Vishal Ahuja, Researcher in the Scientific Image Analysis and Simulations team at Shell Technology Centre Bangalore, holds a master's degree in chemical engineering from the Indian Institute of Technology – Bombay and a PhD in computational chemical physics from the University of Twente, the Netherlands. He joined Shell in 2018 as a researcher in the Computational Science department and has developed several image processing solutions for the business and worked extensively in the area of multiphase flow simulations through porous media.

Aarthi Thyagarajan has worked for Shell since 2007 and is a Digital Commercial and Open Source Manager at Shell Technology Centre Bangalore. She is currently pursuing an Executive MBA at IMD Switzerland. Her team's collaborative efforts have established an open-source programme and standard at Shell, which reflect a dedication to transparent and responsible digital technology deployment. She has 20 years' experience in techno-commercial delivery and new business development, with a strong technical/chemical engineering background from Indian Institute of Science.

Evren Unsal is the academic partnerships manager based in Amsterdam, the Netherlands. She oversees Shell's research partnerships with academia, industry and governmental institutes in continental Europe. Evren is also, since 2020, the executive editor of *Shell TechXplorer*. She has a PhD in polymer and fibre engineering from Auburn University, USA.