Unifob AS – Avdeling for Petroleumsforskning
CIPR – Centre for Integrated Petroleum Research
University of Bergen
Allégaten 41 * N-5007 Bergen, Norway
Tel: 47 55 58 00 00 * Telefax 47 55 58 82 65 * www.cipr.uib.no

REPORT

REPORT NO:	ACCESSIBILITY:
UP 05/2007	**Open**
REPORT TITLE: An improved compaction model for reservoir simulation and coupled flow – stress simulation	
AUTHORS: Øystein Pettersen	

DATE:	NO. OF PAGES:	NO. OF APPENDICES:	E-MAIL TO CONTACT PERSON:
13.02.2007	**40**	**2**	**oystein.pettersen@cipr.uib.no**
CLIENT: BP / Valhall			

SUMMARY:
Based on exact strain calculations from a simplified coupled flow – stress simulation run, the reservoir is subdivided into a number of "pseudo soil regions" such that in each sub-region compaction is a function of fluid pressure only, while still honouring the original soil properties. This revised compaction model is tailored for the flow simulator framework, and when used in that setting the flow simulator computes a compaction state which is as good as identical to the "exact" state computed from strain, in (almost) every grid cell.
The construction process is always possible, and an error tolerance can be set such that coupled simulations can be guaranteed to run in explicit mode (no pore volume iterations needed) without loss of accuracy.
The overall gain is a flow simulator computed compaction field which is accurate at all times (not only at stress steps), and which can be computed with significantly less computer effort than with the standard approach.

KEY WORDS:	
Compaction	**Reservoir simulation**
Coupled simulation	**Computational mechanics**

Executive Summary

The conventional compaction model used in reservoir simulators defines compaction as a function of fluid pressure, whereas in reality it is a function of effective stress. The industry standard procedure to compute accurate compaction is by performing coupled flow simulation and rock mechanics simulations. However, to achieve the desired accuracy, the coupled simulation must be carried out with *pore volume iterations*. The pore volume iteration scheme always converges, but may be, and often is time consuming.

In this paper we first demonstrate that the non-uniform distribution of the compaction state within the reservoir can mainly be explained by the (rock mechanics) boundary conditions and the production process (well positions and rates). Based on this result, we attempt to subdivide the original soil regions into sub-regions, such that *in each sub-region, compaction is a function of fluid pressure only*, and hence the new (pseudo) soil description on the sub-regions is suitable for use in a flow simulator.

This novel way of altering the soil description to fit into the flow simulator framework implies that an accurate compaction solution will be computed already by the flow simulator, a vast improvement over the traditional flow simulation compaction modelling. More important perhaps is that the flow simulator compaction state provides an interface to the stress simulator by which the "exact" rock state can be computed without need for pore volume iterations. (Or exceptionally with a few iterations.)

Description of work flow

1. Construction is based on results from a simplified coupled flow simulator – stress simulator run, denoted the *tuning run.* The tuning run should be representative of the actual run, but only a few (e.g. three) stress steps are needed, and the simulation can be run in explicit mode.
2. Based on exact compaction calculations from the tuning run, each original material region is subdivided into a number of sub-regions, such that in each sub-region, compaction is (approximately) a function of fluid pressure only.
3. For each sub-region defined above, optimal compaction vs. fluid pressure relations are constructed
4. Sub-regions (from bullet 2) and associated compaction functions (tables) (from bullet 3) are used to define the flow simulator compaction model. The actual coupled simulation can then be run, as explicit coupling, and with larger stress steps.

The workflow has been automated by a computer program, `mech2sim`, so very little user interaction or expert knowledge is needed.

In the paper we also demonstrate that the construction process is always possible, and that the sub-regions can be chosen such that the rock mechanics simulator *will* converge to the correct compaction at each stress step, without pore volume iterations. (Disregarding some rarely occurring exceptional cases.) Factors influencing the overall computing time and convergence process are discussed.

The overall gain is greater accuracy in the flow simulator results (compaction, fluid pressure and saturations), which are valid at all time steps – not only at the stress steps as in the traditional iterative scheme, and a significant reduction of computer processing time, without sacrificing any accuracy.

1. Introduction

A producing reservoir will be influenced by soil compaction in several ways, e.g. in a depletion process the total available compaction energy will govern reservoir pressure development and thereby production rates and totals.

Conventionally, compaction in a reservoir simulator is modelled as a grid cell pore volume multiplier vs. fluid pressure. This is a simplified model that is used because fluid pressure is the only available parameter for compaction computations in the simulator, and is only partly based on physics, since it does not take account of the reservoir rock behaviour, which may be nonlinear poro-elasto-plastic, depending on stress path, temperature, and possibly water content (Longuemare *et al.* 2002).

Volumetric compaction is a function of effective stress σ, defined as

$$\sigma = \sigma_T - \alpha p_f \tag{1}$$

where σ_T is total stress, p_f fluid pressure, and α Biot's constant, which is a measure for the relative significance of grain compressibility versus bulk compressibility (Wood 1990). For incompressible grains $\alpha = 1$; in general $0 < \alpha \leq 1$ (Biot 1941; Terzhagi 1943).

Of special relevance for compaction calculations is the mean effective stress p (the average of the diagonal elements in the stress tensor),

$$p = \left(\sigma_{xx} + \sigma_{yy} + \sigma_{zz}\right)/3 \tag{2}$$

and the volumetric strain ε_p (the sum of the diagonal elements in the strain tensor),

$$\varepsilon_p = \varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz} \tag{3}$$

Also of interest is the deviatoric stress q, which simplified is the difference between radial and axial stress. (For exact definition, see Wood 1990 or Pettersen 2007).

During the last decades there has been a growing awareness that the dynamic stress state in the reservoir often has a significant impact on petrophysics and fluid production, and that this interaction can only be understood by performing coupled rock mechanics and reservoir simulator studies (Koutsabeloulis, Heffer, and Wong, 1994; Settari and Mourits, 1994; Gutierrez and Lewis, 1998; Koutsabeloulis and Hope, 1998; Settari and Mourits, 1998; Mainguy and Longuemare, 2002; Longuemare *et al.*, 2002; Thomas *et al*., 2003). Extending reservoir simulations to also take account of stress state computations will normally increase run time by at least an order of magnitude. Even acknowledging that this is necessary to gain the needed knowledge, there is undoubtedly a need to investigate methods which can reduce overall computing time (e. g. Settari and Walters, 1999).

The ideal manner to simulate the soil–fluid interaction is to solve the full coupled system of stress and fluid flow equations (Settari and Walters, 1999; Gutierrez, Lewis, Masters, 2001; Longuemare *et al.*, 2002; Lewis, Makurat, Pao, 2003). This is, however, complex and time consuming. In addition, currently no fully coupled simulator exists which includes all options provided by the market leading stress simulators or reservoir simulators, although development of such software is in progress (Koutsabeloulis and Hope, 1998; Stone *et al.*, 2003; Liu *et al.*, 2004) Hence, it is interesting to look at the alternative

approach of *partial coupling*, where stress state and reservoir fluid dynamics are computed by dedicated software with data exchange at chosen time steps, called *stress steps* (Gutierrez and Lewis, 1998; Longuemare *et al.*, 2002; Mainguy and Longuemare, 2002; Thomas *et al.*, 2003; Dean *et al.*, 2003). In *explicit* coupling the data exchange is one-way only. First the flow simulator is run a time interval ending with the stress step. Then the simulated reservoir state (fluid pressure and saturations) is used to initialise the rock mechanics simulator, which computes the stress state at the time. This computed stress state is further used to update the flow simulator data (typically porosity or / and permeability) where after the flow simulator progresses the solution in time to the next stress step (Heffer *et al.*, 1992). The explicit scheme provides a qualitatively correct stress (and compaction) distribution. However, the stress or compaction *level* need not be correct, due to the inter-dependency between true compaction and the flow simulator computed compaction used to initialise the rock mechanics simulator. Therefore the explicit scheme has been improved by iteratively updating the flow simulator cell pore volumes by the values calculated by the rock mechanics simulator at the stress step until convergence, *iterative* coupling (Settari and Walters, 1999, Chin *et al.*, 2002, Onaisi *et al.*, 2002; Tran, Settari, Nghiem, 2004). As noted by Settari and Walters (1999), iterative coupling is as accurate as full coupling if taken to full convergence, but can be very costly in terms of computing time.

An alternative approach suggested in this paper, is to redefine the soil description used by the flow simulator, such that it computes near-accurate compaction in all grid cells. The “pseudo soil model” is constructed from results obtained by a simplified coupled simulation (explicit with few stress steps), which can be done with affordable computing time. In that manner a very accurate compaction field can be obtained already in the flow simulation, and all subsequent coupled runs can be done as explicit (or in exceptional cases with substantially fewer pore volume iterations than by the standard scheme).

The procedure / workflow will be presented step by step, including background material and details. The overlying philosophy was described in the executive summary, and should be kept in mind.

2. Preliminaries

We will investigate the relationship between the true compaction values computed from strain and those computed from fluid pressure, and demonstrate how the flow simulator pore volume compressibility concept can be modified such that stress simulation can be performed by the explicit scheme, or if an iterative scheme is needed, the number of iterations can be greatly reduced, hence reducing overall processor time considerably.

Some of the referred coupling schemes are based on code modification in the flow simulator or / and the stress simulator, and some are restricted in the choice of poro-elasto-plastic model (e.g. linear elastic). The scheme we suggest is valid for a general poro-elasto-plastic model, and is based purely on data exchange between commercial simulators (no code modification is needed).

The simulators that have been used in the study are the finite element rock mechanics simulator VISAGE ™ *) from V.I.P.S. Ltd, (VIPS, 2003) coupled to the market leading finite difference reservoir simulator ECLIPSE™ **) from Schlumberger (Schlumberger 2005).

2.1 Notation

2.1.1 Pore volume multipliers, *m*

We will use *m,* the ratio of current to initial cell pore volumes (denoted *pore volume multiplier*) as a measure for compaction, hence

$$m = \frac{PV(t)}{PV_0} \tag{4}$$

where, in each grid cell, PV_0 is the initial pore volume and $PV(t)$ the pore volume at time *t*.

The flow simulator computes compaction from functions (normally tables) of pore volume multipliers vs. fluid pressure, while in reality *m* is a function of mean effective stress *p*. Further, for a deforming rock, the compaction of a control volume can be computed from volumetric strain.

To distinguish between the different computational sources for the pore volume multipliers used in the paper, we define the following notation,

m_{pf} Function of fluid pressure used by the flow simulator.

m_p Function of mean effective stress

m_ε Computed from volumetric strain, $m_\varepsilon = \exp(\varepsilon_p^0 - \varepsilon_p(t))$ (5)

where ε_p^0 and $\varepsilon_p(t)$ are volumetric strains initially (no load) and at time *t* (compressive strain positive). Provided calculated strains are correct, this *m*-function represents the true compaction to which other values will be compared.

The notation m_{pf} will be used both to denote the input function (table) used for flow simulation, and for grid cell values of *m* derived from this function (similar for m_p).

The function $m_p(p)$ can be the result of laboratory experiments, but can also be derived from the poro-elasto-plastic model. As shown in Pettersen 2007, Critical State theory (CST) is an appropriate model for sands or sandstone (or in practice a special case of CST, the Cam Clay model (Wood 1990)). For consistency reasons, the $m_p(p)$ we have used in the simulations were derived directly from the Cam Clay model (Pettersen 2007), or from an extension of CST to chalk.

In order to be consistent with the poro-elasto-plastic model and applicable in the proposed procedure, the flow simulator compressibility function m_{pf} must fulfil some minimum requirements

i) It must be possible to differentiate the behaviour of m_{pf} during loading and unloading, as e.g. by reversible or irreversible compaction, or partly irreversible (hysteresis)

ii) The simulator must allow for multiple functions m_{pf}, valid for different material types defined on separate material regions.

Typically, also the flow simulator will put some restrictions on the input m_{pf}, as e.g. ECLIPSE requires that the tables are monotonic non-increasing with load.

Note: *Loading* is to be understood as increasing mean effective stress, which amounts to decreasing fluid pressure, i.e. a loading process is what is commonly denoted depletion in the reservoir engineer context.

2.1.2 Energy level and distribution

When discussing compaction energy (p_f or p) or compaction we will use the term *level* to denote an absolute value of the magnitude (as e.g. the average cell value) and the term *distribution* to denote (in a non-strict fashion) how the parameter varies in the reservoir. Obviously the reservoir energy or compaction state is correct if and only if both level and distribution are correct.

2.1.3 Reservoir state

The vector of all contributing dynamic variables is denoted the *reservoir state* Σ, which is comprised of two parts, the *flow state* Σ_F and the *rock state* Σ_R. In essence, the flow state is comprised of those variables computed by the flow simulator, and the rock state of the variables computed by the stress simulator,

$$\begin{aligned} &\Sigma = (\Sigma_F, \Sigma_R), \text{ with} \\ &\Sigma_F = \mathbf{u}, p_f, S_l, \ldots \\ &\Sigma_R = \boldsymbol{\sigma}, \boldsymbol{\varepsilon}, \boldsymbol{\varsigma}, \ldots \end{aligned} \qquad (6)$$

where $\mathbf{u}$ is Darcy velocity, p_f is fluid pressure, S_l are fluid saturations *(l = oil, water, gas)*, $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ are the stress and strain tensors, and $\boldsymbol{\varsigma}$ is the displacement vector.

2.2 Rock Mechanics Boundary Conditions and Flow Simulation

In a coupled simulation model large volumes of over-, under-, and sideburdens are normally included, with rock mechanics boundary conditions that constrain the model edges (far from the reservoir). Thereby the interaction between porous and non porous rock is honoured, as e.g. the influence of the surrounding soil on reservoir deformation. It is well known that basing compaction calculations on fluid pressure as in a flow simulator where this interaction is missing, will be inaccurate.

Figure 1. Fluid pressure contours

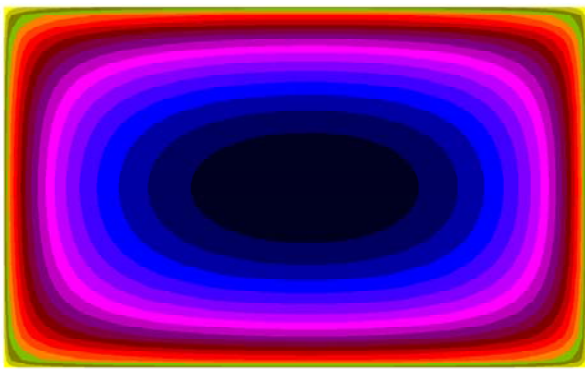


Figure 2. Corresponding compaction distribution

A simple illustrative example is shown in Figures 1 and 2, depicting contours of fluid pressure and the corresponding true compaction, computed from strain, for a depletion process in a box-shaped reservoir with a row of injectors along the western edge and a row of producers along the eastern. In this case the vertical displacement at the top of the reservoir is non-uniform (bowl-shaped – “arching effect”) causing a corresponding compaction field which cannot be reproduced by the fluid simulator by a conventional

compaction model. (Note that contours of compaction computed from fluid pressure by the standard flow simulation model would be as good as identical to Figure 1.)

Omitting the rock mechanics influence will also often result in an inaccurately simulated flow pattern, due to permeability reduction in compacted volumes (see e.g. Pettersen 2007). For the same example as in Figures 1 and 2, using the compaction field calculated by the flow simulator (m_{pf}) results in a piston-like fluid front movement (Figure 3, left), while using the accurate compaction field (m_{ε}) the simulated fluid front shape is as depicted in Figure 3, right.

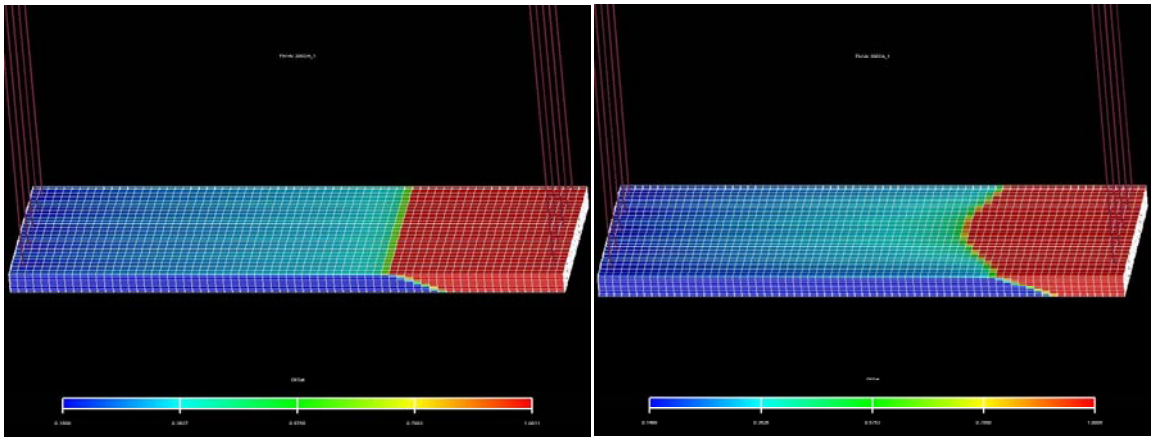


Figure 3. Contours of water saturation.
Left: Using standard flow simulation compaction model.
Right: Using accurate compaction computed from strain.

Neglecting true rock boundary conditions is an error source in flow simulator compaction computations, as there is no way to account for the geometry dependent compaction distribution resulting from these boundary conditions in a flow simulation using the standard compaction model.

2.3 Iterative coupling – Description and theory

In a coupled scheme, the stress simulations are performed at selected time steps, called *stress steps,* such that the flow simulator is run between the stress steps to compute the appropriate flow states, and the stress simulator is run at the stress steps to compute the rock state at the selected times. The procedure is shown schematically in Figure 4.

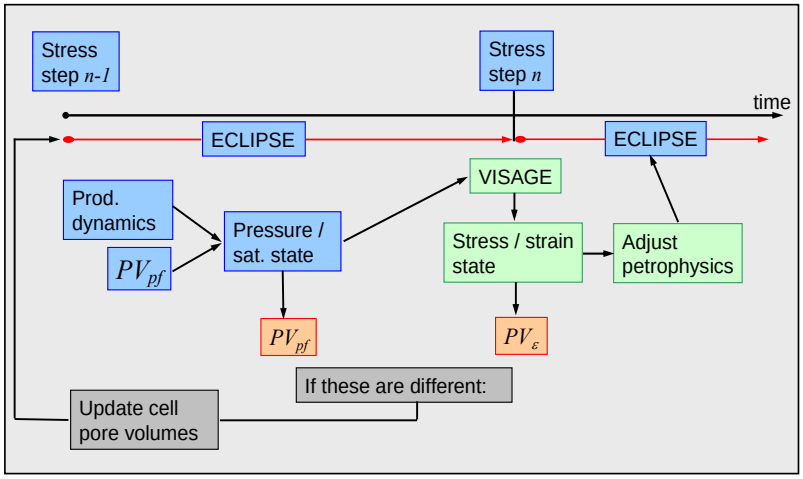


Figure 4. Schematic of coupled flow and rock mechanics simulation, with optional pore volume iterations

2.3.1 "Standard" iteration

To solve the problem *f(x) = x* by iteration, set $x^0 = c_0$ (the iteration *initialiser*), then update *x* by

$$x^k = f(x^{k-1}),\ k = 1,2,\ldots,\ \text{until}\ \left\|x^k - x^{k-1}\right\| < \varepsilon_{\text{tol}} \qquad (7)$$

where ε_{tol} is some predetermined error tolerance.

A number of methods exists to accelerate the convergence, but irrespective of the algorithm used, intuitively the number of iterations needed to converge will be fewer if the initialiser is closer to the sought solution (which also reduces the risk of converging to an erroneous solution).

2.3.2 Iterative coupling – Technical Description

At stress step *n-1* (denoted t_S^{n-1}) the reservoir state has been computed to $\Sigma^{n-1} = (\Sigma_F^{n-1}, \Sigma_R^{n-1})$.

The flow simulator is now run from t_S^{n-1} to the next stress step t_S^n. The computed state Σ_F^n is stored and the flow simulator stopped. The rock state at t_S^n is then found by the stress simulator by solving the rock mechanics system of equations

$$\mathbf{F}(\Sigma_R) = \mathbf{0} \qquad (8)$$

subject to the rock boundary conditions and Σ_F^n obtained from the flow simulation.

Note: The system (8) is solved in the stress simulator by an iterative scheme. To avoid confusion we will always refer to the iteration scheme used *internally* by the stress simulator to obtain a single stress solution as *solver iterations*.

The initialiser for the solver iterations is $\Sigma_0^n = (\Sigma_F^n, \Sigma_R^{n-1})$ (9)

From Σ_F^n we can calculate the flow simulator computed pore volumes in all grid cells

$$PV_{pf}^n(c_i) = PV(m_{pf}(c_i), \Sigma_F^n) \qquad (10)$$

where c_i is a cell-index that runs over all (active) cells.

(**Note:** Throughout the paper, an expression as *PV(...)* is to be read as "*PV* is a function of …")

Correspondingly, the pore volumes computed from the strain state are

$$PV_{\varepsilon}^n(c_i) = PV(\varepsilon_p(c_i), \Sigma_R^n) \qquad (11)$$

where $\varepsilon_p(c_i)$ is to be understood as the volumetric strain at the nodes of cell c_i (ref. Equation (5)).

The solution Σ^n is accepted if

$$\left\|PV_{pf}^n - PV_{\varepsilon}^n\right\| < \varepsilon_{\text{PVtol}} \qquad (12)$$

where $\varepsilon_{\text{PVtol}}$ is a predefined tolerance. ($\|\cdot\|$ is some appropriate norm, e.g. maximum or L_2).

If the criterion is not fulfilled, the flow state Σ_F^{n-1} is modified by setting $PV(c_i) = PV_{\varepsilon}^n(c_i)$ in all cells c_i. Then the flow simulator is re-run from t_S^{n-1} to t_S^n (using the modified Σ_F^{n-1}), and the rock mechanics simulation repeated with the updated initialiser Σ_F^n.

The cycle {[flow simulation] → [stress simulation] → [pore volume update]} is repeated until pore volume error tolerance is acceptable.

It is this loop (external to the simulators) that is denoted *pore volume iterations*.

As noted by Mainguy and Longuemare (2002) (and others), "the pore volume compressibility in conventional reservoir simulation is a parameter determined by the reservoir engineer, which can be considered a numerical parameter, since whatever the value supplied by the reservoir engineer, the rock mechanics simulator will provide the exact porosity." They also demonstrate that the number of iterations needed in an iterative coupling scheme may be very sensitive to this parameter. Apart from this observation there seems to be no attempt to utilise the "compressibility parameter" to increase the efficiency of coupled simulations in the referred papers.

When Σ_F^{n-1} is changed by a pore volume update, the pressure development and hence pore volumes will change at all time steps between t_S^{n-1} and t_S^n, hence changing the initialiser for the stress simulator, and thereby computed strains. In addition, solver classification may change, as e.g. some grid cells could be in the elastic region on one pore volume iteration and in the plastic region on another. The pore volume iteration scheme does therefore not necessarily guarantee rapid convergence; on the contrary very slow convergence has often been experienced in practice.

The object is hence to,

1. Accelerate the iterative scheme by, in lieu of updating pore volumes, construct "rules" by which pore volumes and fluid pressures are updated simultaneously
2. construct the "pseudo soil model" such that an "optimal" flow state is computed already by the flow simulator, for use as initialiser by the stress simulator.

2.3.3 The role of m_{pf} – relations between p and p_f

The solver iterations initialiser Σ_F^n is dependent on the flow simulator computed compaction,

$$\Sigma_F^n = (p_f, m_{pf}, S_l, \ldots)^n \tag{13}$$

As noted earlier m is a function of mean effective stress, $m = m(p)$. However, the flow simulator uses a fluid pressure dependent compaction, $m = m_{pf}(p_f)$.

The functional dependency of the different parameters can be schematically set up as,

$$p = p\,(BC, p_f, \boldsymbol{\sigma}, \ldots) \tag{14}$$

$$p_f = p_f(m_{pf}, W_{\text{pos}}, W_{\text{rate}}, \ldots) \tag{15}$$

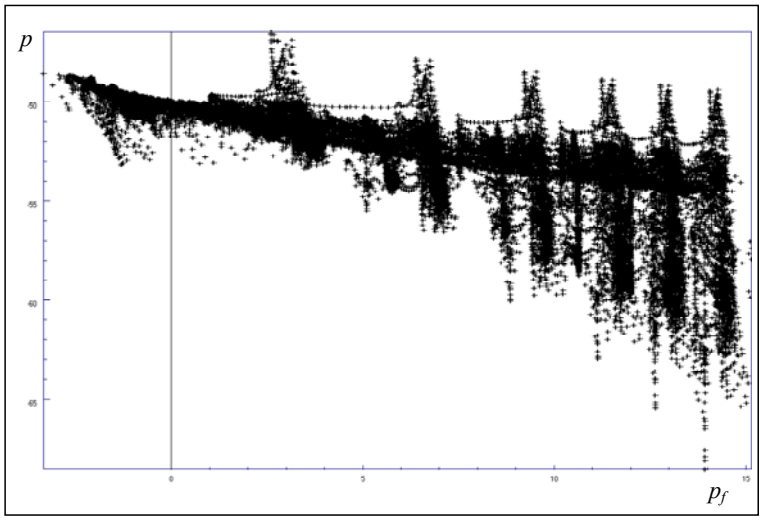
$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(BC, R_N(\mathbf{x}), R_{\text{prop}}(R_N), \ldots) \tag{16}$$

where BC = boundary conditions, W_{pos} and W_{rate} are well positions and rates, $R_N(\mathbf{x})$ is the soil distribution, and $R_{\text{prop}}(M)$ is soil properties for material M.

Hence,

$$p = p(BC, p_f, W_{\text{pos}}, W_{\text{rate}}, R_N(\mathbf{x}), R_{\text{prop}}(R_N)) \tag{17}$$

From this equation we clearly see that in general no simple relationship between p and p_f exists. An example demonstrating this is shown in Figure 5, where a (p vs. p_f) correlation is shown for all grid cells in a material region for six stress steps.



Note on representation:
In a plot like Figure 5, which is a kind of correlation plot, each point (x, y) on the plot is the value-pair (x, y) recorded from one grid cell. ((x, y) in this case would be (p_f, p)).

Figure 5. Correlation mean eff. stress vs. fluid pressure in one material region, six stress steps

On the other hand, Figure 2 is an indication that the distribution of m_{pf} may be dominated by boundary effects, and that hence p vs. p_f relations may exist *locally.* This topic will be studied in section 3.

It is also of interest to examine to what extent $p(p_f)$, or rather $m_{pf}(p)$ depends on the production process, i.e. if an established relationship can be expected to be robust with respect to changing production conditions, a question which will be addressed in section 5.1.

Static and dynamic parameters

Equation (17) can alternatively be formulated as a relation for m_{pf},

$$m_{pf} = m_{pf}(p_f;\ BC,\ p,\ W_{\text{pos}},\ W_{\text{rate}},\ R_N(\mathbf{x}),\ R_{\text{prop}}(R_N)) \qquad (18)$$

We wish to split this equation into a static and a dynamic part.
The static parameters are; BC, W_{pos}, $R_N(\mathbf{x})$, and the initial $R_{\text{prop}}(R_N)$,
while the dynamic ones are p, W_{rate}, and $R_{\text{prop}}(R_N)$

Note: Many of the qualitative features of the compaction distribution are primarily determined by the *static* soil properties (the poro-elasto-plastic model, geometry, and soil behaviour in the surrounding non-porous rock), defining a quasi-static compaction *distribution*. On the other hand, the compaction *level* is primarily determined by the dynamic process, which obviously also influences the distribution, but often this influence is smaller than the static contribution. E.g., altering m_{pf} will change the flow simulator fluid pressure level, which then influences the stress simulator computed strain and compaction level. Hence the choice of m_{pf} is directly tied to the final compaction level in an explicit scheme.

The function m_{pf} can always be defined as

$$
\begin{aligned}
&m_{pf} = m^S_{pf} + m^D_{pf}, \text{where} \\
&m^S_{pf} = m^S_{pf}(p_f; BC, W_{\text{pos}}, R_N(x), R^0_{\text{prop}}(R_N)) \\
&m^D_{pf} = m^D_{pf}(p_f; p, W_{\text{rate}}, R_{\text{prop}}(R_N))
\end{aligned}
\qquad (19)
$$

The idea is now to establish a procedure for determining the static m^S_{pf} which can be used to compute high quality cell pore volumes already by the flow simulator. The dependency on boundary conditions implies that this pseudo m^S_{pf} necessarily must be position dependent, $m^S_{pf} = m^S_{pf}(p_f; \mathbf{x})$. When used in the flow simulator, the computed flow state will be closer to the true flow state, and hence fewer pore volume iterations will be needed. The dynamic part of Equation (19) is only implicitly used, as it is accounted for by the pore volume iteration process. Experience from a number of simulations is a strong indicator that m^D_{pf} generally *is* small compared to m^S_{pf}, and that few if any pore volume iterations will therefore be needed (ref. sections 4.3.1 and 7)

Note: Exact one-to-one correspondence between flow simulator and stress simulator computed compaction would be hard or impossible to achieve, and is not necessary. As long as the flow simulator computed pore volumes approximate the actual values *on average locally*, the exact distribution will eventually be found by the stress simulator (explicit run). This will be further discussed in section 7.

Discontinuous pore volume changes

One obvious disadvantage of the classic iterative coupling scheme is that correct pore volumes are computed and updated only at stress steps. Hence, if the family of input m_{pf} tables are not sufficiently close to the optimal ones, a false compaction trend will be computed between stress steps, also affecting the fluid pressure. Then at the stress step the compaction field will be corrected, leaving the resulting pore volumes to be inconsistent with the computed fluid pressure. Thereby the flow simulator when restarted will be faced with an unphysical pore volume – fluid pressure combination, and at its next time step will be forced to "reinitialize" the pressure. This may cause convergence problems (due to an unphysical starting point) and the end result is also a pressure field that is

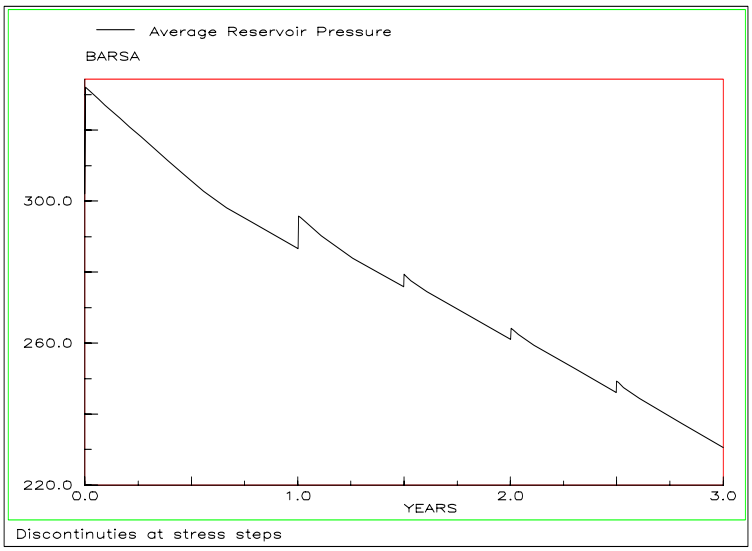


Figure 6. Fluid pressure vs. time, showing discontinuities

valid only at the stress steps, and typically will have discontinuities there. An example of such behaviour is shown in Figure 6. The same argument applies to permeability updates when a stress dependent permeability is modelled. Hence, the advantage of improved compaction calculations in the flow simulator goes further than only reduced computing times, e.g. providing a flow state which is valid at all time steps.

2.4 Case studies description

The case study examples will be used both to exemplify and describe the construction procedure, and to present results from runs using the methodology.

Two different models have been used. The simple one has been valuable for establishing and discovering fundamental relationships, but lacks the realism which is required to develop required robustness, while the complex model aims at covering "all" configurations that can develop during a coupled run, and is considered sufficiently realistic to be used as a test of robustness.

2.4.1 Single Material Sandstone (Case SMS)

The grid is regular cartesian box-shaped and is comprised of 78 x 33 x 17 cells, where the outer four cell columns are used for over-, under-, and sideburdens. I.e. the reservoir proper consists of 70 x 25 x 9 active cells. The reservoir soil is a homogeneous single material moderately weak high permeability sandstone, which has been modelled as Cam Clay with initially vanishing ellipse axes, and horizontal unloading-reloading lines. Hence the soil enters the plastic region in *p*-*q*-space immediately when loading is commenced (Pettersen, 2007), and deformation is permanent, i.e. no pore volume recovery on unloading, corresponding to *irreversible rock tables* in the flow simulator compaction functions (ref. section 4.4.1).

In the base case a row of injectors is placed along the western edge of the reservoir, and a row of producers along the eastern edge. The injection and production rates are controlled by reservoir voidage, providing a slow depletion case, i.e. on average pure loading. The over-, side-, and underburdens are modelled as Mohr-Coulomb soils, using "standard" North Sea conditions.

This model is sufficiently simple to allow for isolating of features of interest, yet with some realism.

2.4.2 Multi Material Chalk (Case MMC)

In this case we attempt to model a reservoir with realistic and complex soil properties, yet confined to a manageable size.

The modelled reservoir is a fractured chalk, where some fractures have been modelled explicitly (as 1 m wide high-permeability cells), and some are modelled by a "transition zone", with implicit fractures. The reservoir grid is comprised of 57 x 115 x 8 cells, with four additional columns for over-, under-, and sideburdens (total 65 x 123 x 16 cells).

The soil is heterogeneous and anisotropic, with smooth transitions between the different material types, and has been grouped into 11 different material regions, of which six are reservoir soils.

The matrix and transition zones have been modelled by NGI's chalk model for Valhall, which is similar to Critical State, with an extension which includes water weakening / swelling and a more realistic dilation model. The fractures (both explicit and implicit) have been modelled with stress dependent

fracture closure (permeability reduction), and with differing parameters for north-south and east-west fractures.

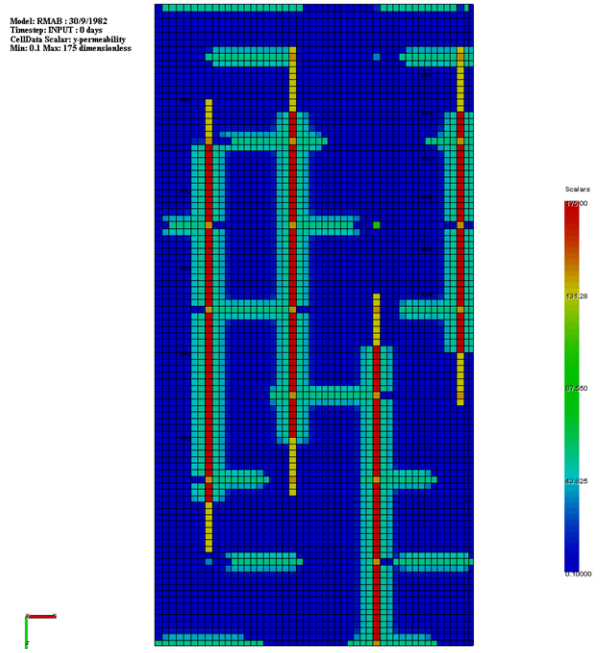


Figure 7. MMC model, North-South permeability

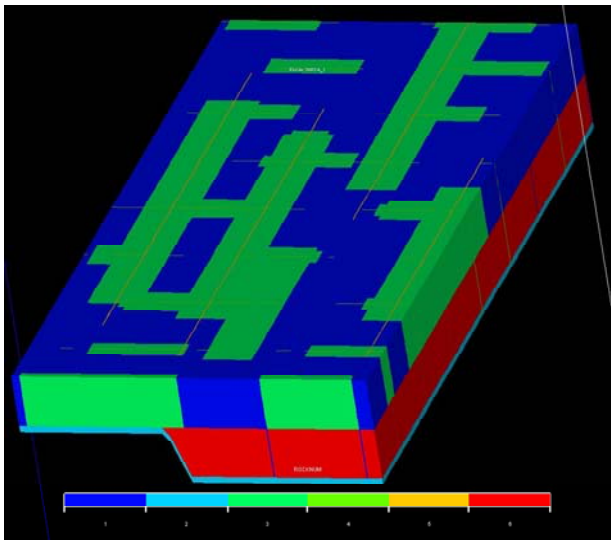


Figure 8. MMC model, Geometry with the six original material regions colour-coded

The model also includes a chalk pinchout zone and a hardground layer modelled as Mohr-Coulomb. Over-, side-, and underburdens have been modelled as Mohr-Coulomb soil using Valhall data. The model, including the soil models used, is fully documented in Standnes and Pettersen, 2006.

Figure 7 shows north-south permeability K_y in the top reservoir layer, while Figure 8 shows the reservoir geometry and the six soil regions used.

The base model is essentially a water drive from east towards west, with rates tuned for a slow loading process. However, unloading occurs locally to a much larger extent than in the SMS case, due to the large contrasts between the different soil types.

3. Physical motivation – Local behaviour

The fluid pressure field from the base case SMS run is not surprisingly regular, decreasing from the injectors to the producers, as shown in Figure 1. Even though this field has no symmetry at all, the corresponding compaction distribution (m_ε) obtained from the stress simulator is quite symmetric (Figure 2), indicating that the strain distribution is much more influenced by the rock boundary conditions than by the flow process. A simple test to study the role of the boundary conditions versus the flow process is to run the same model with differing process data.

First injectors and producers were swapped, without affecting the strain field noticeably.

The next case was an unsymmetrical 5-spot run, with fluid pressure distribution as shown in Figure 9. Still the strain was not noticeably different from the base case strain. Lastly, the load rate was tested, increasing the pressure difference between the injector row and producer row. At high pressure

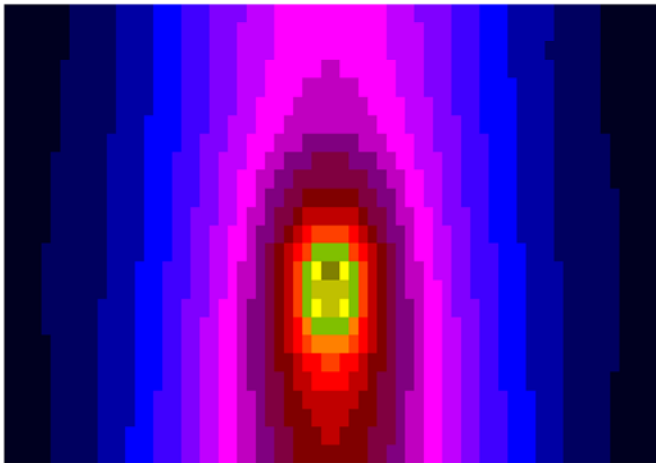


Figure 9. Contours of fluid pressure from the unsymmetric 5-spot run.

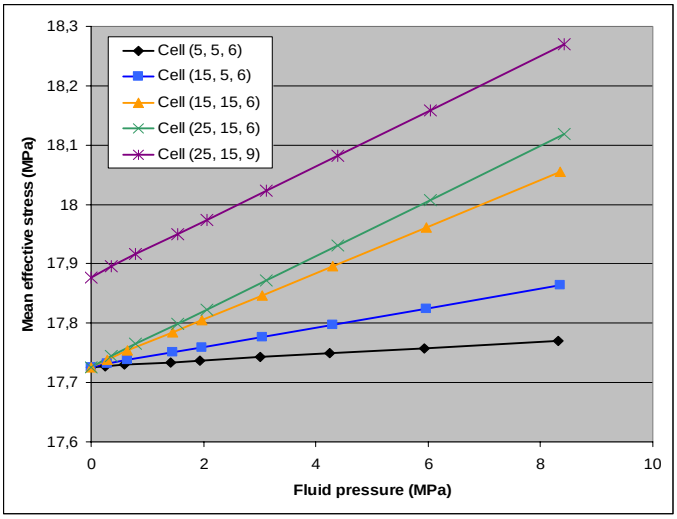


Figure 10. Correlation p vs. p_f in some grid cells, SMS case

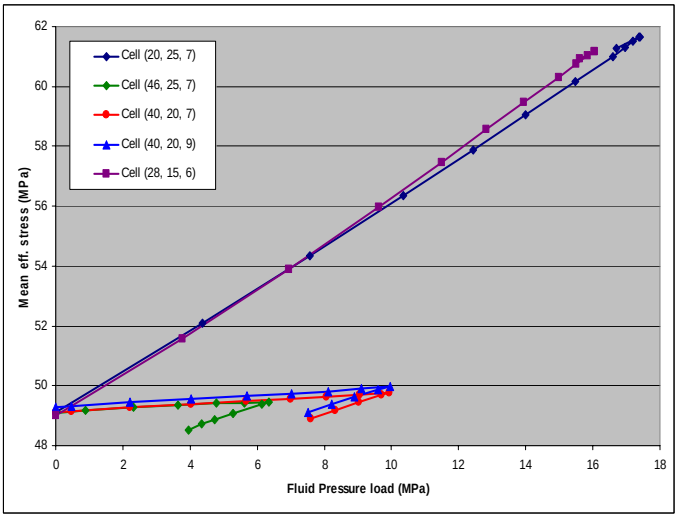


Figure 11. Correlation p vs. p_f in some grid cells, MMC case

differences, the minimum of the m_ε-bowl was shifted towards the producers, but this became noticeable only at unrealistic production conditions.

For the SMS case the tests hence indicate that *boundary conditions have a larger influence on the strain field than he flow process*.

This would signify that the lack of correlation between fluid pressure and mean effective stress (Figure 5) is primarily due to *geometrical differences*. I.e. in a *(p vs p_f)* plot we should expect large variation between different positions in the reservoir, but a more regular relation if measured at the same point. To that extent we have plotted the correlation *(p vs p_f)* in a single cell, for a number of different cells. Some of the results are shown in Figures 10 (for the SMS case) and 11 (for the MMC case).

The SMS case depicts an almost perfect correlation, and the MMC also has a well defined relationship, except for the deviating points which clearly are due to local unloading.

Hence, although a general *(p vs p_f)* plot appears rather chaotic (as in Figure 5), *locally* a functional relationship seems to exist. (Locally to mean single cells or a group of related cells). This will be the basis for the construction of local (or pseudo) compaction functions.

The fundamental assumptions for construction of pseudo compaction functions m^*_{pf} are then,

i. Pore volume multipliers for each original material can be defined as a function of fluid pressure and position, $m_{pf}(R_N) = m_{pf}(R_N; p_f, \mathbf{x})$, where R_N is a material type identifier (corresponding to ROCKNUM in ECLIPSE)

ii. Each original material region R_N can be subdivided into (pseudo) sub-regions such that in each sub-region m^*_{pf} is (approximately) a function of fluid pressure only, $m^*_{pf}(R^j_N) = m^*_{pf}(R^j_N; p_f)$, where R^j_N denotes sub region j of (original) material region N.

Obviously, if anything is to be gained by the procedure, the pseudo regions and compaction functions must be possible to construct with small effort.

4. Procedure description

4.1 Tuning Run – (p_f, m_ε)-point clouds and envelopes

The pseudo material regions and associated m-functions are constructed from the results of a *tuning run*. The tuning run is related to the actual coupled simulation, but smaller, as it can be run in explicit mode, and only a few stress steps are needed, three steps being a typical number. The production scheme should be defined such that the entire load range of interest is covered by the tuning run stress steps, and a complex scenario is probably best simplified. The input m_{pf} tables are not critical as they are not used in the construction process, but they should not be too far from reality. As an example, we have often used experimental $m_p(p)$ functions and replaced p with p_f.

From the tuning run, in each grid cell the "exact" pore volume multiplier m_ε is computed from strain (Equation (5)). The point cloud comprised of all cell values m_ε vs. p_f is the basis for further construction. An example of such a point cloud (from the SMS case) is shown in Figure 12 (For illustrative purposes a point cloud constructed from six stress steps is shown, i.e. somewhat more than was used in the tuning run.)

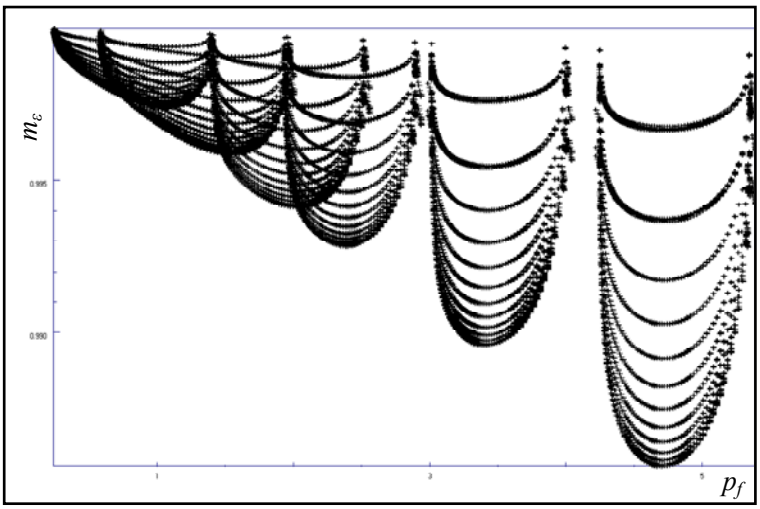


Figure 12. Pore Volume Multipliers from strain vs. fluid pressure, SMS case, six stress steps

Note: In practice unloading-reloading points are discarded from the point cloud, i.e. in each grid cell a new point is discarded if a smaller p_f-value has been recorded earlier.

By an explicit run the stress simulator will compute a strain level and distribution based on the input flow state from the flow simulator (which

depended on the input m_{pf}). If the input flow state is wrong, as it normally will be on this stage, the strain *level* will also be wrong, but the distribution is correct relative to the utilized initialiser (flow state), hence the reservoir state is a *valid solution*, admittedly not the solution we are seeking, but still a physically possible solution in the reservoir and soil description we are studying.

Each point in the point cloud is then a valid (observed) combination of m_ε and p_f. This observed value is probably not related to the correct pressure or compaction in the cell it was taken from at the simulated observation time at all, if compared to the final (correct) reservoir state (which at this stage is unknown). But the only important fact in this context is that all the points in the point cloud represent valid combinations of m_ε and p_f.

The area between the upper and lower envelopes of the point cloud hence defines the region of *permitted combinations* of m_ε and p_f in the (p_f, m_ε)-plane, and this region is only dependent on the boundary conditions and reservoir and soil properties, not on the input m_{pf} tables, which are hence only indirectly used in the construction, to define the cell pressures. (Although if the flow simulator compaction tables are too far off, the simulated load range can be inappropriate.)

For a structured case like SMS, with only a single material in the reservoir, *the lower envelope can easily be seen to represent compaction far from boundaries*.

For compound soils, a similar physical interpretation is not always obvious, as the point cloud for each original material will depend both on the outer boundaries as above, but also on internal soil-to-soil interactions. But the conclusion above, that the point cloud envelopes are independent of the m_{pf} tables used in the flow simulator still remains valid.

Hence, we state the **Proposition** (which is very significant for the region construction procedure),

> The upper and lower envelopes of the (p_f, m_ε)-point cloud depend only on the reservoir and soil distribution and properties, not on the flow simulator compaction model.

The point clouds for the six different soils in the MMC case are shown in Appendix A.

4.2 Construction of envelopes

In cases that are as well-structured as the SMS case the envelope construction is straightforward, with both the lower and upper envelopes being well defined. Note that it is a natural requirement that both envelopes pass through the point (0, 1), i.e. at vanishing load the pore volume multiplier is unity, corresponding to no compaction.

> **Note:** Although "no load" is uniquely defined as a function of p, the concept is not always well defined in the flow simulator. Since the "standard" flow simulator uses tables of m versus p_f, and initial p_f increases with depth, vanishing mean effective stress corresponds to a depth dependent pressure, a fact which is normally not taken account of in flow simulation. As we will see later, the pseudo material regions concept will also to some extent fix this inconsistency.
>
> In the general case, as exemplified by the point clouds shown in Appendix A, the envelope construction is not that straightforward, although based on the same principles.

In short, the construction has been based on the following guidelines,

1. Capture actual trends by removal of outliers
2. Disregard non-flow-simulator features, as e.g. dilation
3. Maximum smoothness, i.e. use as few points as possible to define the envelopes, without sacrificing existent features
4. Give larger weight to densely populated regions of the point clouds than sparsely populated
5. Disregard unloading – reloading intervals if they disagree with general trend
6. Attempt to force curve to pass through (0, 1), but not if inconsistent with point cloud
7. Envelopes must be non-increasing with load

It should be mentioned that, given "any" point cloud, the most physically appropriate envelopes are easily identified by "hand drawing". However, converting the intuitively obvious to a robust computer algorithm is not straightforward.

4.3 Construction of pseudo material regions

Once the envelopes are available, the construction of the pseudo material regions is straightforward. Normally, the span in m_ε at a given load (i.e. the distance between upper and lower envelope) increases with load. We wish to use maximum span for a sub-region as a means of error control, and since the total span increases with load, we have chosen to use the span at a given load (10 MPa / 100 bars depletion) as criterion. The number of sub-regions associated with an original region is based on this maximum permitted span (error) for each sub-region. Hence if the total span in m_ε at 10 MPa is $\Delta_{10}m_\varepsilon$ and maximum permitted span in any sub-region is $\Delta_{sub}m_\varepsilon$ the number of sub-regions would be $N = \Delta_{10}m_\varepsilon / \Delta_{sub}m_\varepsilon$.

To determine to which sub-region a cell c_i belongs, the distance between the upper and lower envelopes at the cell fluid pressure value can be imagined subdivided into N intervals, and which of these intervals the cell m_ε falls into determines the pseudo sub-region:

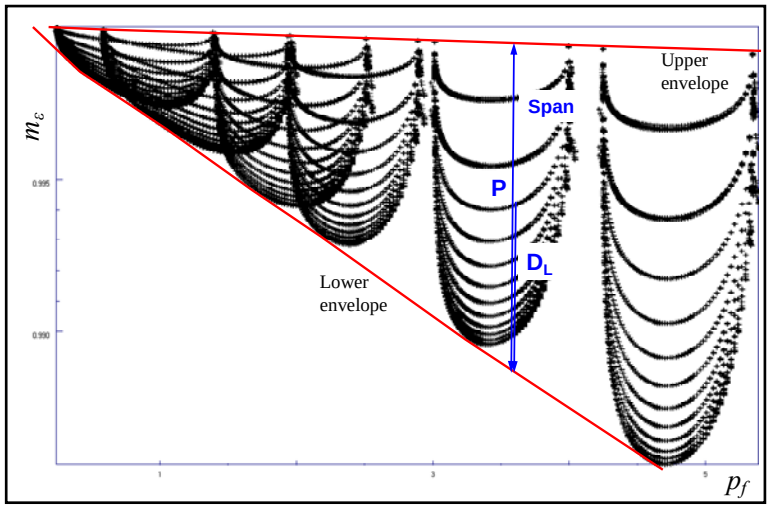


Figure 13. Upper and lower envelopes for the point cloud shown in Figure 12. Example of sub-region assignment

$$R_N^j(c_i) = N \cdot \frac{m_\varepsilon - m_L^e}{m_U^e - m_L^e} \qquad (20)$$

where subscripts U and L denote upper and lower envelopes (superscript e) respectively, and all values in the fraction are evaluated at the fluid pressure of cell c_i.

(In Figure 13, with the symbols used, Equation (20) would read,

$$R_N^j(c_i) = N \cdot \frac{D_L}{\text{Span}})$$

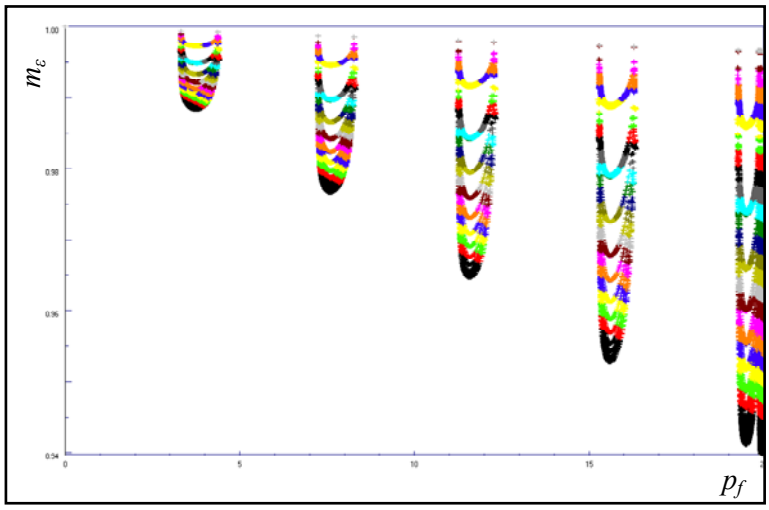


Figure 14. Assigning cells to sub-regions (colour-coded) by the decribed procedure, SMS case as in Figures 12-13

Figure 13 shows the envelopes and construction for the case shown in Figure 12, and in Figure 14 the classification of cells into sub-regions has been colour-coded such that cells of the same colour belong to the same sub-region.

Figures 15 and 16 show examples of the pseudo materials distribution in the reservoir, Figure 15 for the SMS case, which agrees very well with the compaction distribution shown in Figure 2, and Figure 16 for the MMC model,

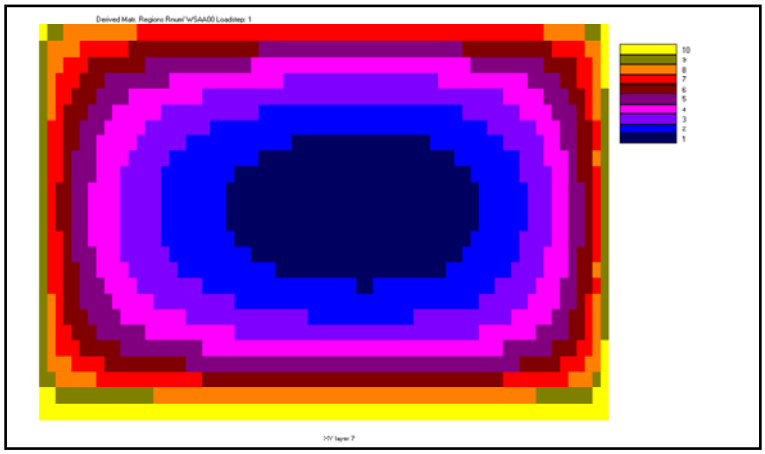


Figure 15. Contours of resulting material sub-regions, SMS case, reservoir layer 3.

where the original regions can be identified, but which also contains areas of high irregularity (the finer details can be difficult to observe due to minute colour differences between the regions).

Note that an important consequence of the proposition in section 4.1 is that the constructed sub-regions are independent of the flow simulator compaction functions used.

Figure 16. As prev. figure, MMC case, reservoir layer 3.

4.3.1 Time-dependence of Sub-regions Construction

In the description above we have suppressed the fact that the construction does not need to be unique. From the tuning run we have several different points in the point cloud associated with the same cell, namely one point for each stress step used. Obviously, in general we cannot expect the sub-region

assignment by Equation (20) to be the same when different stress steps are used. The difference in sub-region assignment for a given cell at different stress steps is a measure for how much the sub-region m_{pf} functions vary with time, or equivalently, if the dynamic part m_{pf}^{D} of Equation (19) really is small compared to the static part.

Firstly, to honour results from all stress steps in the tuning run, the procedure described above is altered slightly. When assigning sub-regions by Equation (20), we use a load-weighted average of the results from all stress steps to determine the sub-region. Load-weighting was chosen, since errors in region assignment are less critical for small loads, where the difference in compaction between different sub-regions is smaller.

To test this variability, for each cell, the maximum difference $D_{max}(c_i)$ between any sub-region computed by Equation (20) and the utilized average is recorded. If the sub-region assignment for a cell c_i is equal for all stress steps, $D_{max}(c_i)$ will obviously be zero, and larger values are a clear indication of inconsistency in the procedure, i.e. that static m_{pf} functions cannot be expected.

A measure for total variability for the model is then, for each possible value of D_{max}, to count the number of cells with such a D-value. If m_{pf} was perfectly static in the material region in question, then all cells would have $D = 0$, and a concentration of D-values near zero would signify a static dominated process.

Figure 17 shows the relevant plot for the SMS case. A great majority of the cells do have a D-value of zero, and the remaining have $D = 1$, signifying only a small discrepancy in sub-region assignment.

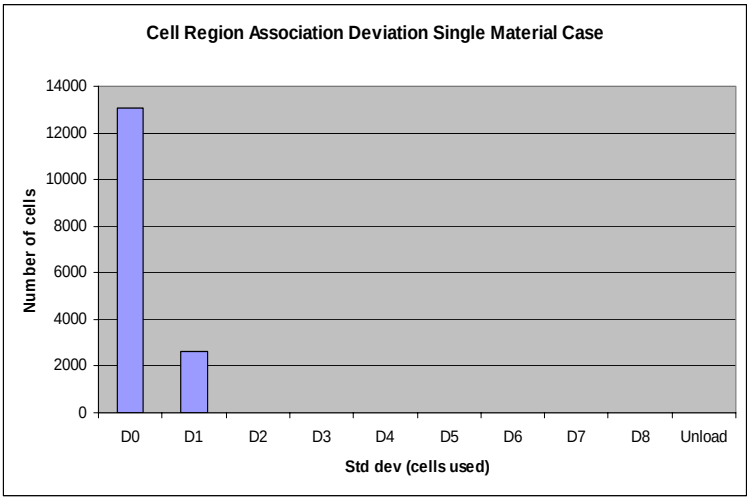


Figure 17. Number of cells with max difference between utilized sub-region and other candidate sub-regions. SMS case.

(Actually, almost all the cells with $D = 1$ are close to a sub-region boundary, so it appears more or less coincidental which side of the boundary they were assigned to). In conclusion, the sub-region assignment for the SMS case is as good as perfectly static.

For the MMC case the results not surprisingly are somewhat more spread. The corresponding plot is shown in Figure 18. Although not as clear as the SMS case, we would still claim that this process is dominated by static behaviour.

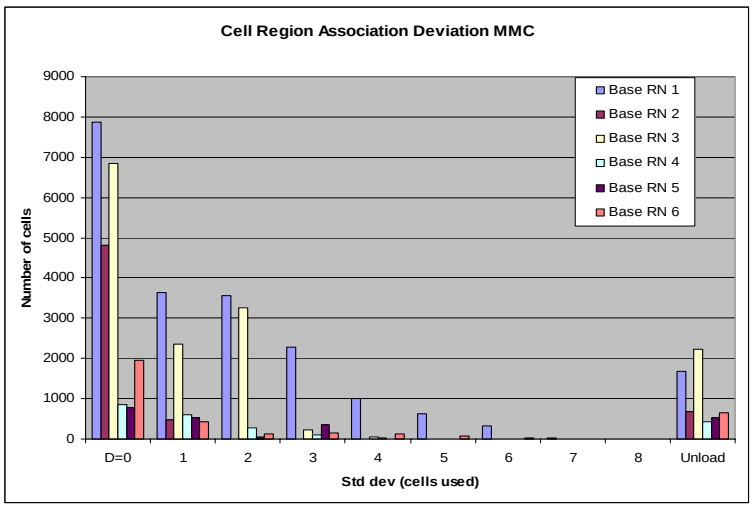


Figure 18. As previous Figure, MMC case

(The "unload" label on the plots correspond to cells which are in the unloading region on all tuning run stress steps, and therefore sub-region assignment is more uncertain.)

4.4 Construction of (local) pseudo compaction functions

Once the pseudo material regions have been defined, the determination of the pore volume multiplier function to use in each region is straightforward. Calculation of $m^*_{pf}(p_f; R^j_N)$, the pseudo *m*-function for sub-region *j* of original material *N*, is done on the basis of a reduced point cloud, comprised of the points $(p_f\ ,\ m_\varepsilon)$ for all cells in R^j_N. The construction is done on a best-fit principle, using piecewise linear regression. The same guiding principles as described in section 4.2 apply in this case. Each regression interval should be chosen such that the number of points and the length of the interval is sufficiently large to identify real trends, and at the same time small enough that non-linear trends are not missed.

Figures 19 and 20 are examples from the MMC case. A typical construction is shown in Figure 19, where the $m_\varepsilon(p_f)$-points from the tuning run are shown in red (tuning stress step no. 1), green (step 2), and yellow (step 3), while the constructed $m_{pf}(p_f)$ is shown in blue.

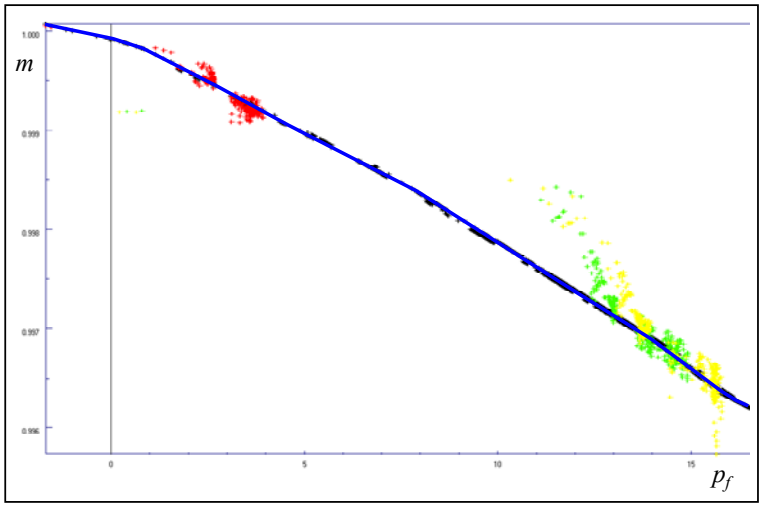


Figure 19. Example of construction of pore volume multiplier curve as a function of fluid pressure, for one sub-region. MMC case (blue curve). Red, green, and yellow point clouds identify the three steps of the tuning run.

All the 25 constructed (pseudo) $m_{pf}(p_f)$ for original material 1 (the matrix) are shown in Figure 20.

Many other examples of the construction procedure, with comments on situations that require special attention are given in the detail results section, Appendix B.

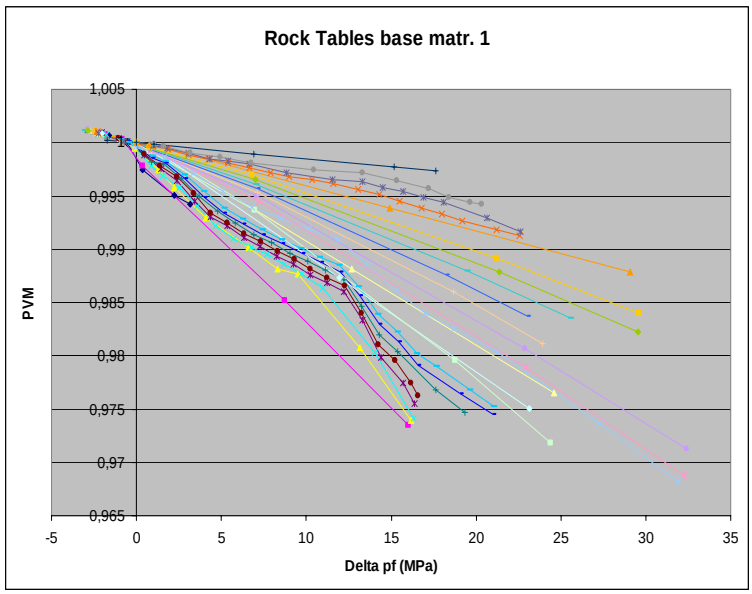


Figure 20. Examples of constructed pore volume multiplier curves. MMC case, matrix

4.4.1 Elasticity – plasticity – flow simulator options

The pore volume multiplier tables used by ECLIPSE ("ROCKTAB") have some options,

- REVERSIBLE curves: The same *m*-curve is used during loading and unloading, corresponding to an elastic soil
- IRREVERSIBLE curves: The *m*-curve is used during loading, and minimum encountered fluid pressure is recorded for each cell, such that for all cell pressures above the minimum, the minimum pressure compaction value is used. This corresponds to permanent deformation without recovery, e.g. Critical State with horizontal unloading–reloading lines
- HYSTERESIS: Each Rock Table is comprised of several sub-families of *m*-functions, one primary used for primary loading, and the rest for unloading–reloading.

During construction of the (pseudo) *m*-curves, both the REVERSIBLE and IRREVERSIBLE cases are directly accounted for, by disregarding cells in the unloading–reloading region in the point clouds. The appropriate flag is then set in the ECLIPSE data deck, and should be compatible with the soil failure model used.

The HYSTERESIS option could in principle be handled by using unloading–reloading data to construct the secondary families of *m*-functions. In practice we have experienced that most runs contain insufficient data for the required construction. By careful construction of the tuning run this hurdle could probably have been passed, but we suggest an alternative and more straightforward route, namely to define the secondary curves with qualitative features depending on the primary curves (e.g. percentage volume recovery), and hence construct the secondary curves on basis of the primaries once these have been defined.

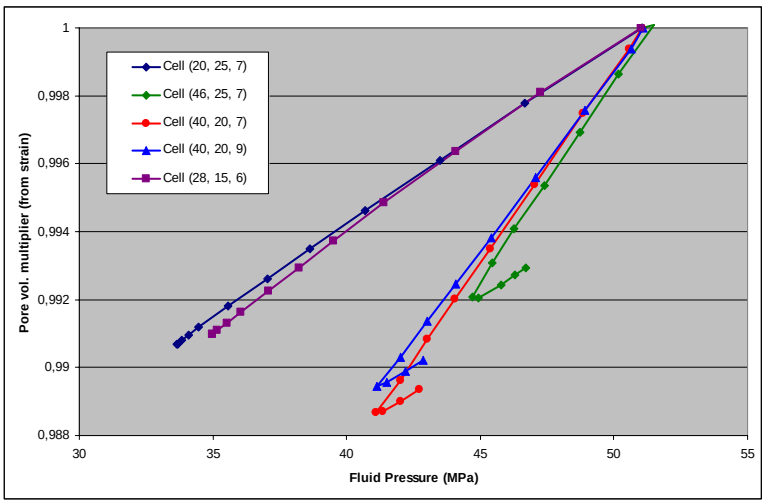


Figure 21. Correlation m_ε vs. p_f in some grid cells, MMC case

Examples of m_ε (p_f) functions for some single cells (the accompanying plot to Figure 11) are shown in figure 21, and clearly demonstrates that the m-curves should preferably be modelled with the HYSTERESIS option in this case. The relationship was taken from the MMC case, which was run with the REVERSIBLE multipliers option. Figure 21 is a strong indication that even better results could have been achieved for this case, if the HYSTERESIS option had been used.

4.4.2 Permeability modification

The ECLIPSE Rock Tables include an optional transmissibility multiplier that can be used to modify permeabilities as a function of fluid pressure. Whether we regard permeability as a function of mean effective stress, or as a function of porosity (pore volume), the presented procedure for converting stress-dependent pore volumes to pseudo soil regions and associated m*-tables handles permeability modification simultaneously, without any extra effort. If e.g. a permeability reduction vs. p is known, the algorithm already maps p to p_f locally, and the same mapping is used to define the appropriate transmissibility multipliers.

Note: Since ECLIPSE uses a single transmissibility multiplier in each cell, anisotropic development cannot be modelled. Since compaction often is dominated by vertical compression, it would then be appropriate to assume that the permeability reduction was anisotropic. Such behaviour is unfortunately not possible to model in the Rock Table context.

If such modelling is seen as important, an alternative route could be to write modified permeabilities to the data deck at every restart step. Such an approach is also consistent with the standard data exchange phase in coupled simulation

5. Case study results and discussion

In this section we look at some results from the case studies. The figures are mainly of three types,

i. Comparison of m_ε and m_{pf}, both as a function of p_f, for one (pseudo) material region. In these plots we regard the m_ε points as the exact solution, to which m_{pf} should be compared. Recall that m_ε is taken from the stress simulator in an explicit coupled run, and that m_{pf} has been taken directly from the flow simulation results.

ii. Correlation m_ε vs. m_{pf}. The plot can be based on one, or several (pseudo) material regions. For a perfect match $m_\varepsilon = m_{pf}$ in all cells, and the closer the shown point cloud approximates this straight

line, the better the match. In these figures, the (m_ε vs. m_{pf}) point cloud is shown in black, while the exact correlation line $m_\varepsilon = m_{pf}$ is shown in red for comparison.

iii. Contour plots, showing the variation of some parameter in the reservoir.

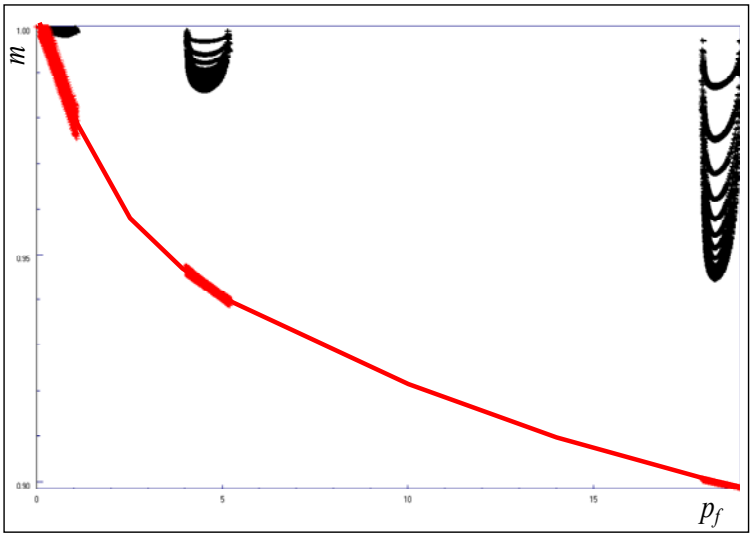


Figure 22. m_ε and m_{pf} vs. p_f, SMS tuning run

This figure shows the great difference between the compaction functions used in the flow simulation, and the compaction computed from strain by the explicit coupling. Note that this difference has no impact on the final upper and lower envelopes. Actually, the role of the Rock Tables (m_{pf}) was tested by re-running the tuning run with Rock Tables that were a closer fit to the final m_ε. In accordance with theory, the envelopes constructed by these two tuning runs were as good as equal.

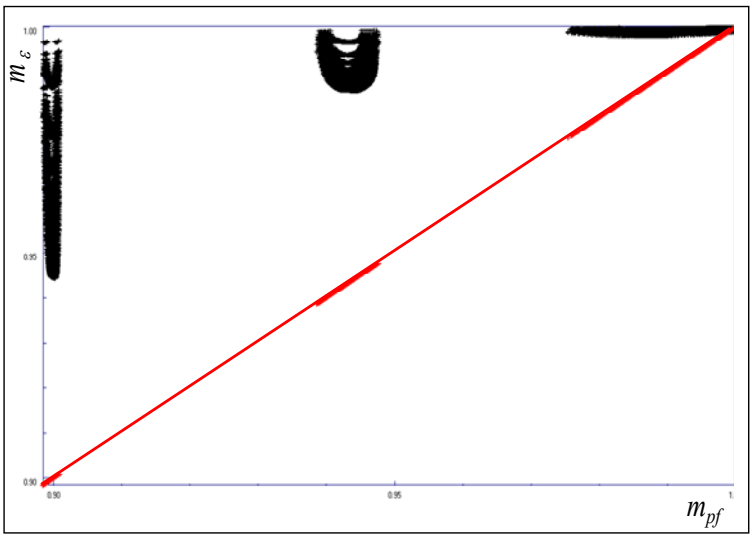
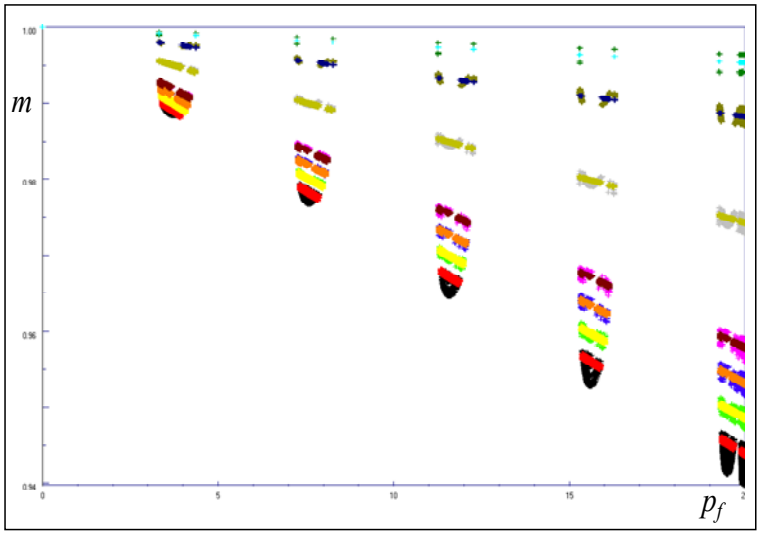


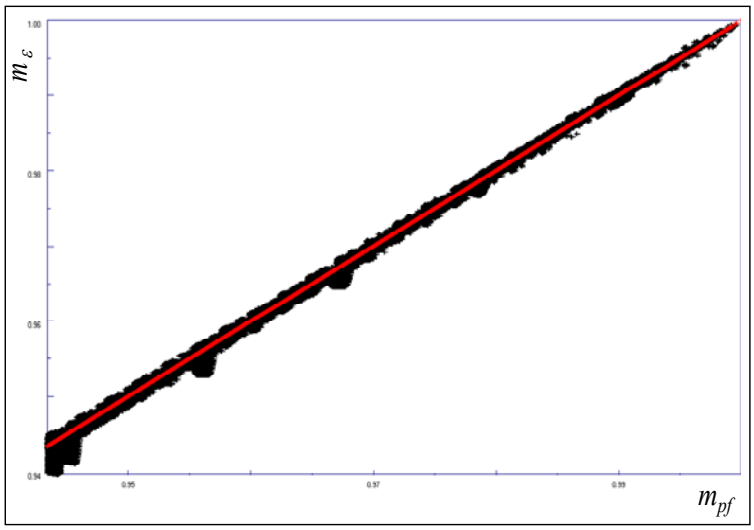
Figure 23. m_ε vs. m_{pf}, SMS tuning run

This is the accompanying figure to Figure 22, and is only included to show how bad the correlation is at the tuning run stage, before any pseudo pore volume multipliers have been constructed.



The match between strain-computed compaction and Rock Table compaction is evident, the largest difference being controlled by the user-defined maximum tolerance.

Figure 24. m_ε and m_{pf} vs. p_f, SMS case, showing the match in some sub-regions, 5 stress steps



The correlation is almost perfect, signifying that ECLIPSE-computed and Visage-computed compaction is as good as equal in all grid cells

.

Figure 25. m_ε vs. m_{pf}, SMS case. All 10 pseudo sub-regions, all 12 stress steps

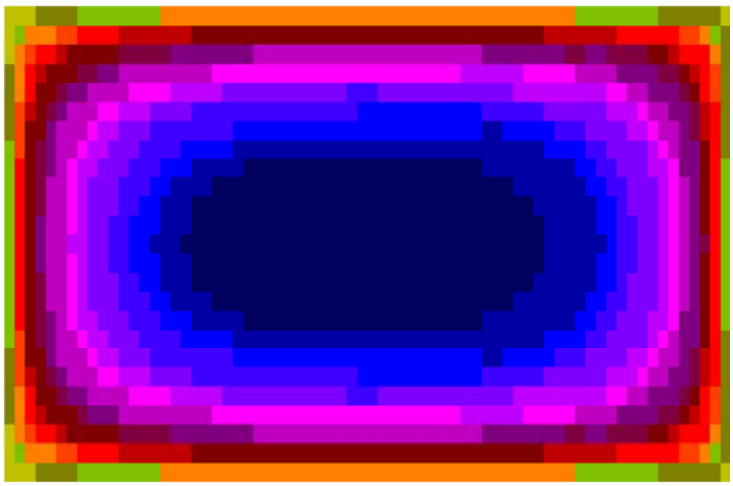
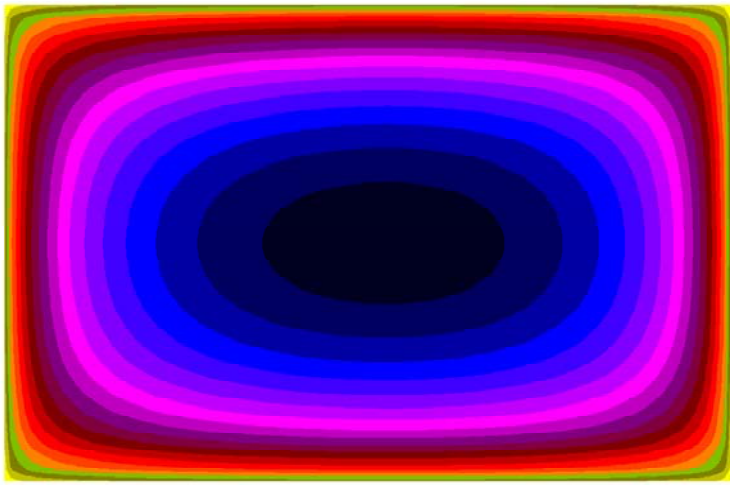
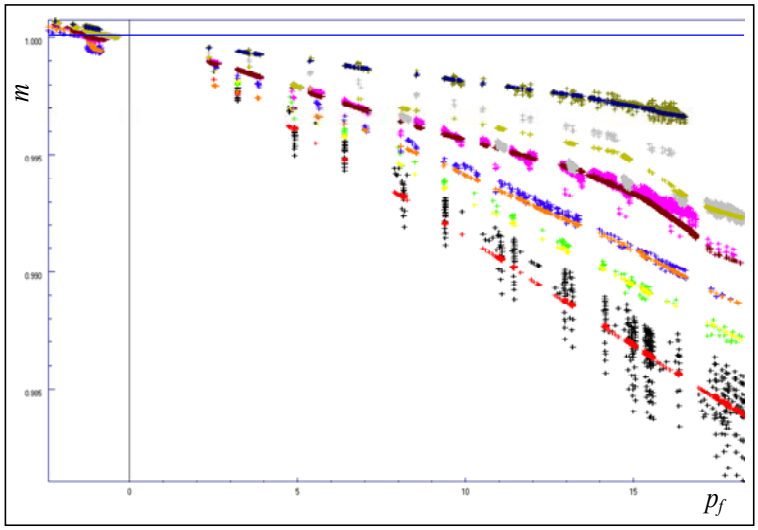


Figure 26. Contours of m_{pf} (from ECLIPSE), SMS case, stress step 3, reservoir layer 3



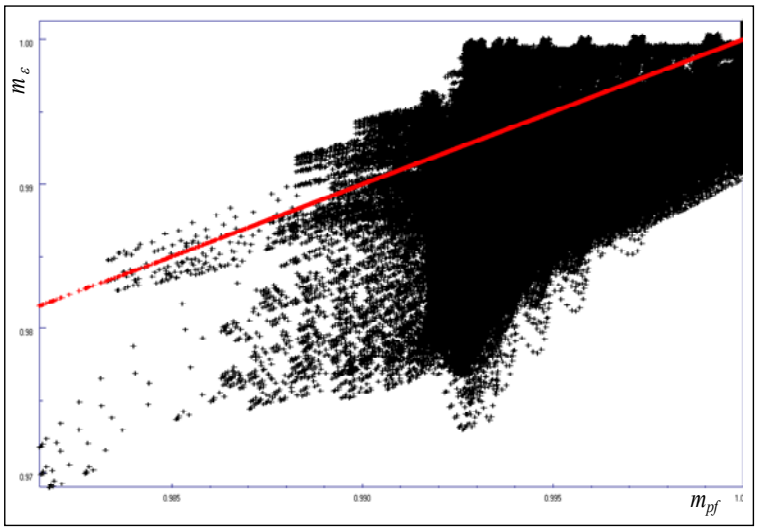
Figures 26 and 27 show an excellent match between the ECLIPSE computed and Visage computed compaction field. Both level and distribution have been successfully computed by the (pseudo) Rock Tables in ECLIPSE

Figure 27. Contours of m_ε (from Visage), SMS case, stress step 3, reservoir layer 3



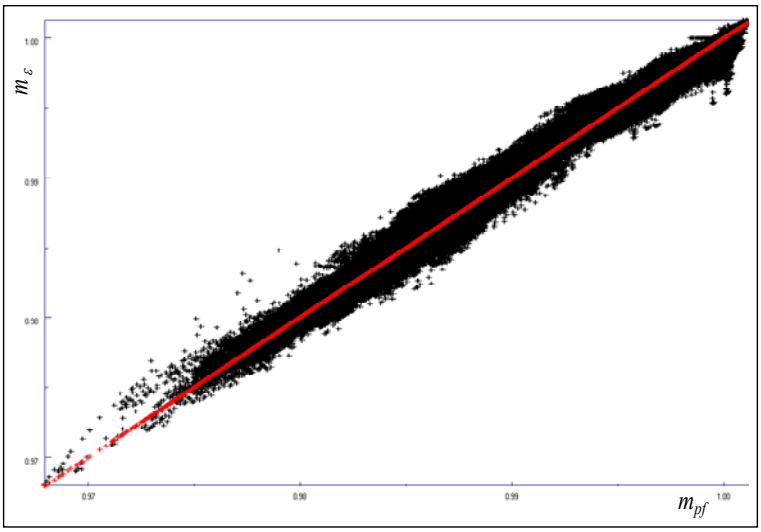
In general an excellent match. The exception is the boundary sub-region (black and red), where a somewhat larger spread is seen, due to outliers. Many more examples are shown in Appendix B.

Figure 28. m_ε and m_{pf} vs. p_f, MMC case, some sub-regions from N/S fractures



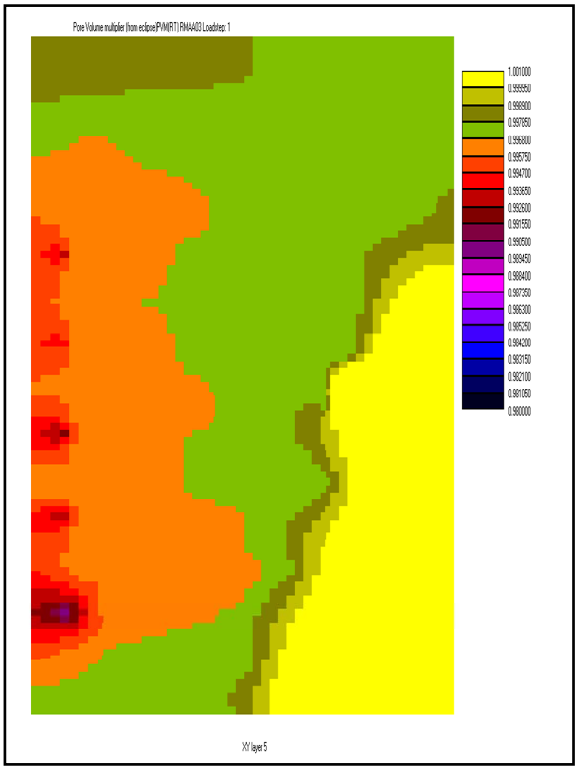
This figure is included for comparison. Similar to the tuning run, but including all 12 stress steps. The Rock Tables are of the traditional ECLIPSE type (one for each of 6 original soil types), but tweaked to match average "exact" tables as closely as possible. This is the best possible result with the "standard approach"

Figure 29. m_ε vs. m_{pf}, MMC case with "optimal standard Rock Tables", all materials, 12 stress steps.



Although not as good as the correlation in Figure 25, the match is still very good.

Figure 30. m_ε vs. m_{pf}, MMC case. All 135 pseudo sub-regions constructed from original 6 material regions, 12 stress steps.



”Standard” compaction field from flow simulator, using one (”optimal”) Rock Table for each of the six original soil regions. (Same run as Figure 29.)

Note: Range of contours is the same in Figures 31, 32, and 33, so compaction level is directly comparable by colour.

Figure 31. Contours of *m* from ECLIPSE, MMC case “standard Rock Table approach”, stress step 3, reservoir layer 2

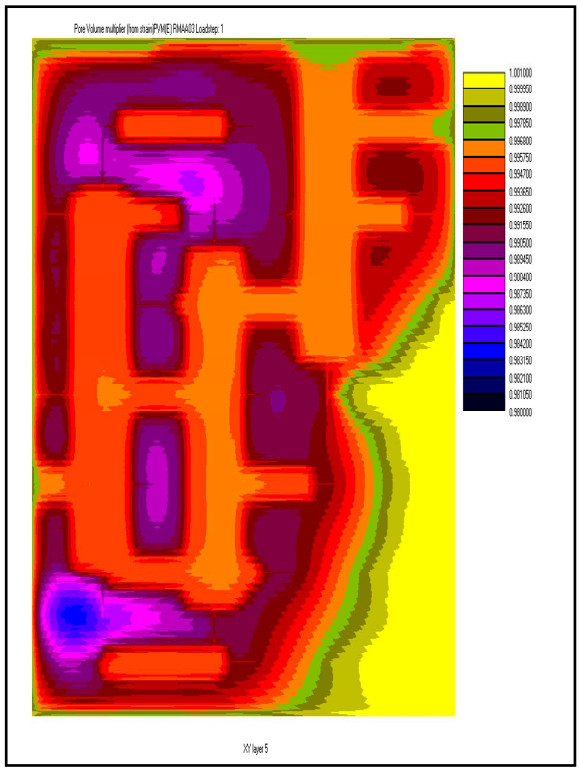


Figure 32. Contours of *m* from Visage strain, as Figure 31.

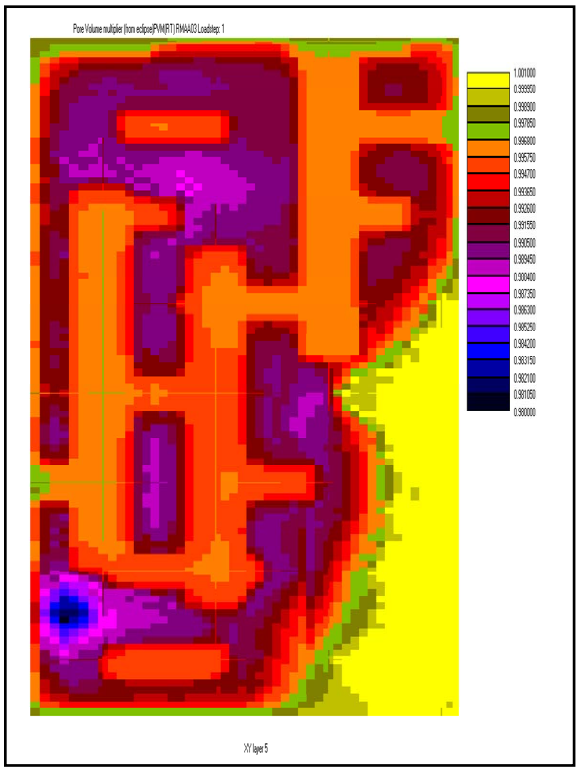


Figure 33. Contours of *m* from ECLIPSE, using 135 (pseudo) sub-regions and Rock Tables, as Figure 31.

By the proposed procedure, the compaction field computed by ECLIPSE matches the "exact" compaction field from Visage very accurately, while the traditional flow simulation compaction model (Figure 31) fails to capture any of the detail variation caused by soil-to-soil interaction and boundary effects.

5.1 Robustness tests

The constructed (pseudo) soil regions and pore volume multipliers depend heavily on the boundary conditions and soil distribution and properties, and the production process itself (sections 2.3.3 and 3).

If the production scheme is changed, the constructed m_{pf}-functions, and probably also the sub-regions become invalid, and must be re-constructed. Intuitively, altered well positions should have a large influence on the construction, whereas changing well rates according to the theory should have less effect.

Exactly how robust the constructed sub-regions / curves are, is difficult to quantify, so the general rule of thumb would be to do a re-construction whenever well configurations are significantly changed.

Firstly it should be noted that none of the test runs showed any dependency on rates, i.e. as long as the well positions and types (injector / producer) are left unchanged, the construction is valid for a large range of rates.

Secondly, to test robustness some cases were run with altered well positions and rates, but re-using the pseudo regions and m_{pf}-functions that were constructed in the base case (i.e. no re-construction in connection with the altered well configurations).

For the SMS case, we have already observed that boundary conditions were the dominating factor. This was confirmed by the sensitivity tests, which all showed correlation as good as equal to Figure 25. Even in

the 5-spot case, with pressure field as in Figure 9, when run with the pseudo sub-regions / rock tables constructed for the SMS base case, the match between ECLIPSE computed and Visage computed compaction was excellent (Figures 34 and 35)

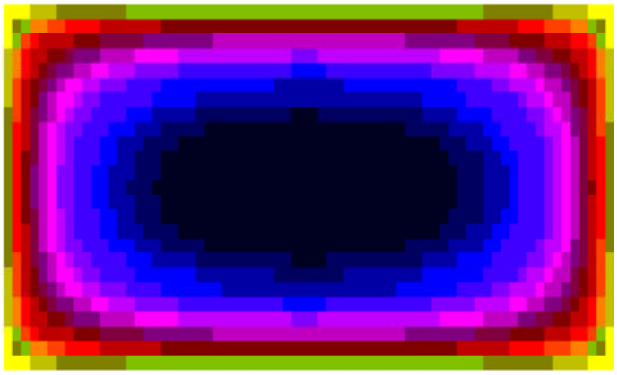


Figure 34. Contours of *m* (from ECLIPSE), SMS 5-spot.

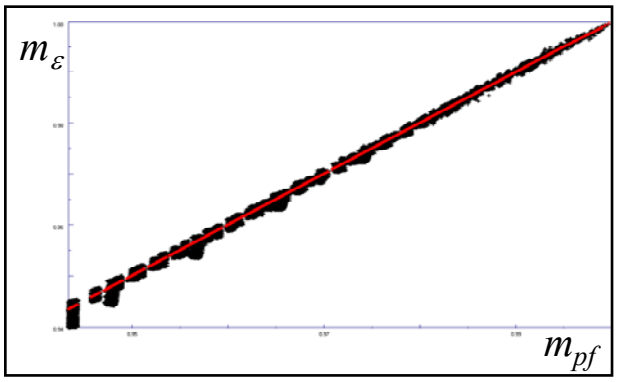


Figure 35. Correlation m_ε vs. m_{pf}, SMS 5-spot

For the MMC case it is only to be expected that the dependency on well positions is stronger than in the SMS case. While the MMC base case was a water drive from west to east, with five injectors and five producers, the robustness test had a more or less random positioning of the wells with 7 injectors and 7 producers, and no dominant flow direction. To test also load rate, the voidage was increased, resulting in a substantially more rapid pressure drop. The constructed regions and curves from the base case were used unchanged. Still, the results were surprisingly good, as seen in Figures 36, 37, and 38.

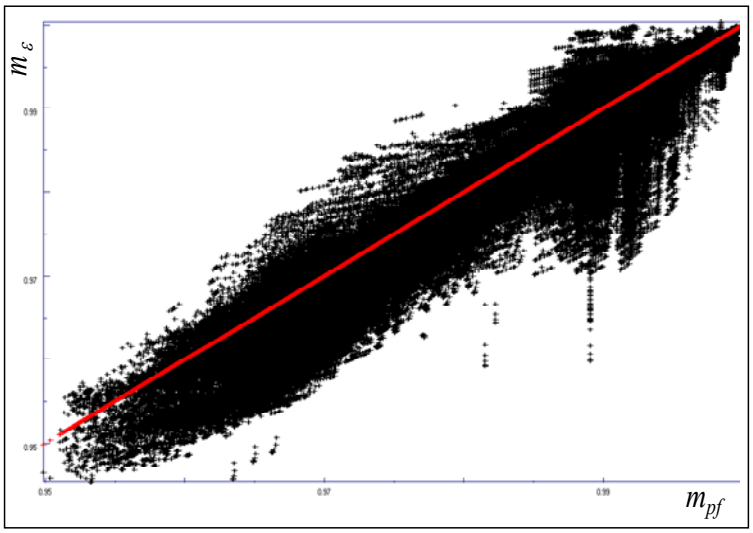


Figure 36. m_ε vs. m_{pf}, MMC robustness test. Pseudo sub-regions and rock tables re-used from base case. (All 135 regions, 12 stress steps)

Not as good a correlation as Figure 30, but on the other hand, definitely better than Figure 29.

Figure 37. Contours of *m* from Visage strain, MMC robustness test

Figure 38. Contours of *m* from ECLIPSE, using base case (pseudo) sub-regions and Rock Tables,

In conclusion, the constructed pseudo soil regions and associated $m_{pf}(p_f)$ tables seem to be more robust with respect to changing production scheme than one would intuitively assume. This may have some interesting consequences, as e.g. it could be possible to use the base sub-regions / rock tables in a geostatistical workflow, such that soil compaction is taken better care of than with a traditional approach.

6. Some practical considerations

The described procedure is completely automated. Once the data deck for the planned run has been completed, the tuning run must be defined by selecting some stress steps to use in this run, and setting up the appropriate run. The user also can control some parameters used in the analysis,

- Maximum *m*-span at load 10 MPa (equivalent to $\max \left\| m_{\varepsilon} - m_{pf} \right\|$ at 10 MPa in any sub-region)
- Percentage points to flag as outliers (and discard) in any point cloud
- Minimum and maximum number of sub-regions in any original material region

The parameters that are not user-defined are defaulted.

The software `mech2sim` writes one file containing the pseudo soil regions in ECLIPSE ROCKNUM format, and one file containing the pseudo *m*-tables in ECLIPSE ROCKTAB format.

The `mech2sim` log file will include useful information, as the mapping from original to new soil regions, and the maximum number of entries in any Rock Table (needed by ECLIPSE).

Excerpt of log file,

```
****************************************************************
**                                                            **
** Org. RockNum  1 subdivided into 25 materials, ROCKNUM   1 to  25 **
** Org. RockNum  2 subdivided into 10 materials, ROCKNUM  26 to  35 **
** Org. RockNum  3 subdivided into 25 materials, ROCKNUM  36 to  60 **
** Org. RockNum  4 subdivided into 25 materials, ROCKNUM  61 to  85 **
** Org. RockNum  5 subdivided into 25 materials, ROCKNUM  86 to 110 **
** Org. RockNum  6 subdivided into 25 materials, ROCKNUM 111 to 135 **
**                                                            **
** Max number of entries in a table:  23                      **
**                                                            **
****************************************************************
```

Next, the actual simulation can be performed, by defining the desired stress steps, and attaching the soil property files defined above in the ECLIPSE data deck.

7. Convergence

As referenced above, Settari and Walters (1999) showed that the pore volume iteration scheme will always converge. A corollary of this is that in any group of cells the compaction level will be correct by the end of the iterations, or stated slightly different; the pore volume iteration scheme will not change the total compaction energy in any group of cells, once total compaction energy in all such groups are correct. (Since this would move the current iterative solution away from the destination, which would be in conflict with the principle of iteration.)

The stress simulator will compute the correct distribution of compaction energy within the group of cells, based on materials properties, process,… (as described in section 2.3). In this process, the compaction energy (manifested by pore volume multipliers) will be distributed among the cells in the group, but without changing the total compaction energy in the region.

Even if total compaction energy has been preserved within a region, the difference between flow simulator computed pore volumes and (correct) strain-based pore volumes can be large in individual cells. If this is the case, the need for pore volume iterations is established.

By the construction procedure we have described above,

1. In any (pseudo) materials region, the total compaction energy is correct, by construction
2. The individual pore volume multiplier error in any cell does not exceed the defined maximum span. (Adjusted for actual load vs. 10 MPa)

Hence, if the (adjusted) maximum span is no greater than the iteration scheme pore volume tolerance (Equation (12)), no cell pore volume errors will be larger than the tolerance, and the pore volume iteration flag is never triggered.

Our experience agrees with this statement. For testing purposes many of the cases we have done were run in iterative coupled mode, but no pore volume iterations were ever performed in any of the runs. (Still we cannot completely rule out the possibility – outliers and dilation are e.g. handled such that pore volume errors could exceed the tolerance, but this would be exceptional cases.)

We state this important observation as a **proposition,**

When (pseudo) material regions are such defined that the *m*-function in each region is a function of fluid pressure only, within an approximation smaller than the pore volume tolerance, the flow state computed by the flow simulator is exact (within tolerance), and coupled simulation can be performed as explicit.

8. Computing times

Since the flow state initialiser is optimal, no pore volume iterations are needed, which obviously has a major impact on total computing time. In addition, the (pseudo) materials description provides a vastly improved compaction modelling in the flow simulator. Hence, if the object of the simulation is primarily the results from the flow simulator, fewer stress steps than by the standard model is needed. In some cases it could actually be possible to use the rock mechanics simulator for the tuning run only, although this approach is not generally recommended – a pure flow simulation has no mechanism of error control, and for this reason alone should be discouraged. In a large number of cases, the results from the stress simulator are needed for field understanding, and in those cases the stress steps should certainly be governed by the request for stress data, not by the flow simulator.

The stress simulator initialiser was, $\Sigma_0^n = (\Sigma_F^n, \Sigma_R^{n-1})$ (Equation (9)). We have already demonstrated that using an optimal value of Σ_F^n eliminated pore volume iterations (in general). We would also expect that this optimal flow state would influence the stress solver itself, reducing the number of solver iterations, and reducing computing time on each (explicit) stress step. However, such behaviour has not been observed. In fact, *the solver iterations seem to be completely independent of* Σ_F^n, and are only influenced by the rock state initialiser Σ_R^{n-1}, taken from the previous stress step. Typical comparisons are shown in Figures 39 and 40. In Figure 39, computing times for the tuning run case (but run with all stress steps for illustrative purposes) and the final run are compared. The tuning case uses

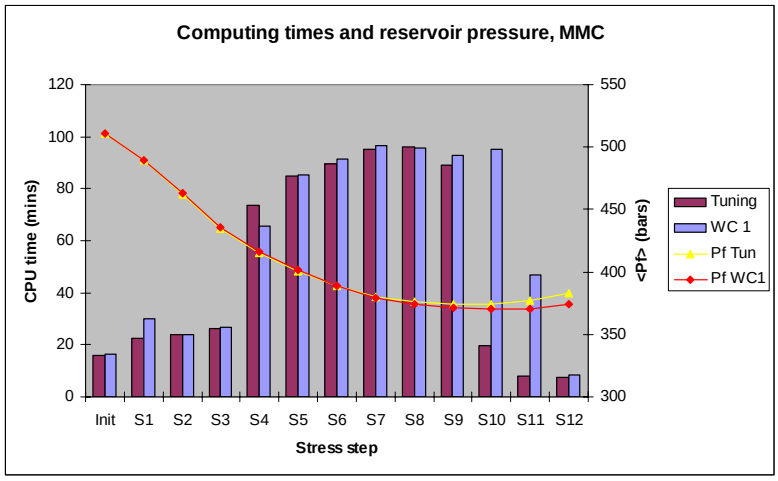


Figure 39. Computing time pr. stress step, equivalent tuning run (w. 12 stress steps) and MMC base (denoted WC1 here)

one *m*-function for each of the six material regions, and none of the *m*-functions are good approximations to reality, while the final run uses the optimal soil regions and *m*-functions. Except for the last stress steps, where the tuning run enters elastic mode, the computing times for each stress step are as good as identical.

Figure 40 shows a comparison of the number of solver iterations for the two cases. The *x*-axis is the nonlinear iteration counter, and the bars show the number of linear iterations for each nonlinear. The plot reveals that not only is the total computing time identical at each stress step, but the detail solver progress towards a solution is also identical (same behaviour was observed for all the stress steps), which demonstrates the independency of the flow state initialiser. Only the existent rock state seems to matter.

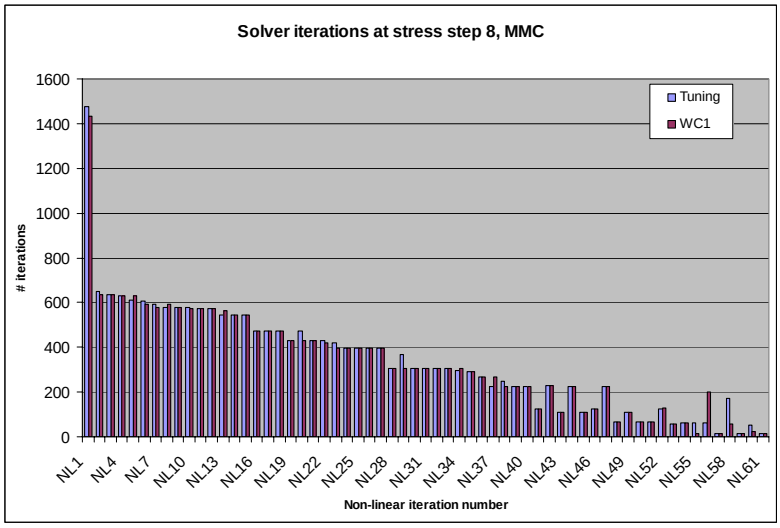


Figure 40. Number of linear iterations pr. nonlinear iteration, equivalent tuning run and MMC base

This can be explained by that the major work goes into computing of the correct qualitative stress distribution, independent of flow state (defining the level). Note that our results are based on Critical State and Chalk models with initially vanishing elastic regions. In a model with mixed elastic and plastic behaviour, changing the level could e.g. change deformation from elastic to plastic, which would certainly affect computing times.

In conclusion, we can reduce / eliminate pore volume iterations by using an optimal flow state initialiser, but we cannot speed up the rock simulator solver.

9. Conclusion

The compaction state computed by a flow simulator (based on fluid pressure, with a traditional approach) is very different from the actual compaction state calculated from strain, which in many cases will result in an erroneous fluid flow pattern if simulated by a flow simulator only. This will be especially relevant for reservoirs containing weak sands or many chalk reservoirs. Further, due to this large difference, the flow-simulator calculated compaction state is typically not a good initialiser for the stress calculations.

In the paper we have shown that in general it is possible to subdivide the reservoir model into a number of (pseudo) sub-regions, such that in each sub-region, compaction is a function of fluid pressure only. In each sub-region, optimal tables for compaction and permeability variation with fluid pressure can be

constructed with accuracy within defined tolerance. When these new-defined functions ("rock tables") are used in the flow simulator setting, a very accurate compaction state (defined by pore volume multipliers for all grid cells) is computed by the flow simulator, matching "exact" compaction computed from strain by the stress simulator almost perfectly. This compaction state is an obvious improvement in itself, being significantly better than what can be obtained by the standard flow simulator approach, and in contrast to coupled runs, the compaction state will be accurate at all times, not only at the stress steps.

Moreover, the improved state is an optimal initialiser for the stress simulator solver, such that coupled flow simulator and stress simulator runs can be done without pore volume iterations, and with larger stress steps if so desired. (Disregarding some exceptional cases).

Since the number of pore volume iterations in the traditional scheme typically exceeds 10, and may even be an order of magnitude larger than that, the presented scheme will reduce overall computing time with at least 90%, without sacrificing accuracy in the end result.

*) VISAGE is the mark of VIPS Ltd.

**) ECLIPSE is the mark of Schlumberger

References

- Biot, M. A., 1941. General Theory of three dimensional consolidation, *J. Appl. Phys.* **12,** 155-163
- Chin, L.Y., Thomas, L.K., Sylte, J.E., Pierson, R.G., 2002. Iterative Coupled Analysis of Geomechanics and Fluid Flow for Rock Compaction in Reservoir Simulation, *Oil & Gas Science and Technology – Rev. IFP,* **57**, 485-497
- Dean, R. H., Gai, X., Stone, C. M., Minkoff, S. E., 2003. A Comparison of Techniques for Coupling Porous Flow and Geomechanics. Paper SPE 79709, *SPE Reservoir Simulation Symposium,* Houston, Texas, Feb. 3-5, 2003
- Gutierrez, M. & Lewis, R. W., 1998. The Role of Geomechanics in Reservoir Simulation. Paper SPE/IRSM 47392, *Proc. Eurock 98, 2,* Trondheim, Norway, July 8-10, 439-448
- Gutierrez, M., Lewis, R. W., Masters, I., 2001. Petroleum Reservoir Simulation Coupling Fluid Flow and Geomechanics. *SPE Reservoir Evaluation & Engineering*, June 2001, 164-172
- Heffer, K. J., Last, N. C., Koutsabeloulis, N. C., Chan, M., 1992. The influence of natural fractures, faults and earth stresses on reservoir performance – analysis by numerical modelling, *Proc. 3rd International Conference on North Sea Oil and Gas Reservoirs,* Trondheim, Norway, Nov. 1992.
- Koutsabeloulis, N. C., Heffer, K. J., Wong, S., 1994. Numerical Geomechanics in Reservoir Engineering, *Comp. Methods and Advances in Geomech.,* Siriwardane & Zeman, eds., Balkema, Rotterdam, 2097-2104
- Koutsabeloulis, N. C. & Hope, S. A., 1998. "Coupled" Stress/Fluid/Thermal Multi-Phase Reservoir Simulation Studies Incorporating Rock Mechanics. Paper SPE/IRSM 47393, *Proc. Eurock 98, 2,* Trondheim, Norway, July 8-10, 449-454
- Liu, Q., Stone, T., Han, G., Marsden, R., Shaw, G., 2004. Coupled Stress and Fluid Flow Using a Finite Element Method in a Commercial Reservoir Simulator, Paper SPE 88616, *SPE Asia Pacific Oil and Gas Conference and Exhibition,* Perth, Australia, Oct. 18-20, 2004
- Longuemare, P., Mainguy, M., Lemonier, P., Onaisi, A., Gérard, Ch., Koutsabeloulis, N., 2002. Geomechanics in Reservoir Simulation: Overview of Coupling Methods and Field Case Study, *Oil & Gas Science and Technology – Rev. IFP,* **57**, 471-483
- Lewis, R. W., Makurat, A., Pao, W. K. S., 2003. Fully coupled modeling of seabed subsidence and reservoir compaction of North Sea oil fields, *Hydrogeology Journal* (**11**) 2003, 142-161

- Mainguy, M., Longuemare, P., 2002. Coupling Fluid Flow and Rock Mechanics: Formulations of the Partial Coupling between Reservoir and Geomechanical Simulators, *Oil & Gas Science and Technology – Rev. IFP,* **57**, 355-367
- Onaisi, A., Samier, P., Koutsabeloulis, N. C., Longuemare, P., 2002. Management of Stress Sensitive Reservoirs Using Two Coupled Stress-Reservoir Simulation Tools: ECL2VIS and ATH2VIS, SPE 78512, *10. Abu Dhabi Intl. Petroleum Exhibition and Conference,* Oct. 13-16 2002.
- Pettersen, Ø., 2007. Sandstone Compaction, Grain Packing and Critical State Theory, *Petroleum Geoscience,* **13**, 63-67
- Schlumberger 2005, ECLIPSE Reference Manual
- Settari, A. & Mourits, F. M., 1994. Coupling of geomechanics and reservoir simulation models, *Comp. Methods and Advances in Geomech.,* Siriwardane & Zeman, eds., Balkema, Rotterdam, 2151-2158
- Settari, A. & Mourits, F. M., 1998. A Coupled Reservoir and Geomechanical Modeling System, *SPEJ* Sept. 1998, 219-226
- Settari, A., & Walters, D. A., 1999. Advances in Coupled Geomechanical and Reservoir Modeling With Applications to Reservoir Compaction, Paper SPE 51927, *SPE Reservoir Simulation Symposium,* Houston, Texas, Feb. 14-17, 1999
- Standnes, D. C. and Pettersen, Ø., 2006. Simulating Flow in Fractures and Fracture Closure on a Generic Valhall Model, Report UP 03 / 2006, Unifob AS / CIPR. (Confidential)
- Stone, T.W., Xian, C., Fang, Z., Manalac, E., Marsden, R., Fuller, J., 2003. Coupled Geomechanical Simulation of Stress Dependent Reservoirs, Paper SPE 79697, *SPE Reservoir Simulation Symposium,* Houston, Texas, USA, Feb. 3-5, 2003
- Terzaghi, K., 1943. *Theoretical Soil Mechanics*, Wiley, New York
- Thomas, L. K., Chin, L. Y., Pierson, R. G., Sylte, J. E., 2003. Integrating Geomechanics in Full-Field 3-D Reservoir Simulation – Modeling Techniques and Field Applications, Paper SPE 77723. Presented at SPE Appl. Techn. Workshop, Corpus Christi, Texas, Jul. 31 – Aug. 1, 2003
- Tran, D., Settari, A., Nghiem, L., 2004. New Iterative Coupling Between a Reservoir Simulator and a Geomechanics Module, Paper SPE 88989, *SPEJ,* Sept. 2004, 362-369
- VIPS, 2003. The VISAGE™ System User's Guide, V.I.P.S. Ltd.
- Wood, D. M., 1990. Soil Behaviour and Critical State Soil Mechanics. Cambridge University Press

Appendix A. Point Clouds for the MMC tuning run

In all the figures, the point cloud for the first (tuning) stress step is shown in black, stress step 2 is in red, and stress step 3 in green. The rock table used in the flow simulation is shown by the yellow line.

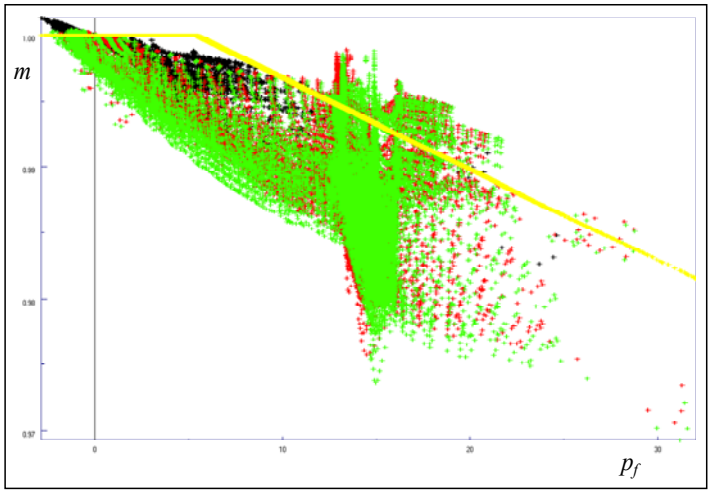


Figure A1. Original soil region 1 (Matrix)

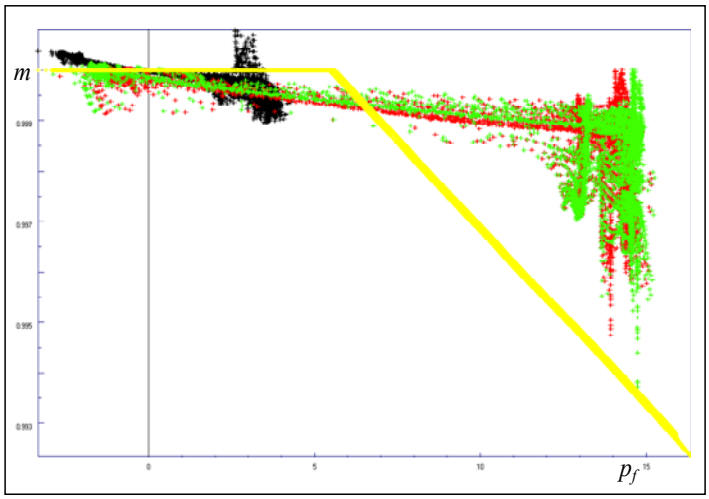


Figure A2. Original soil region 2 (Hardground)

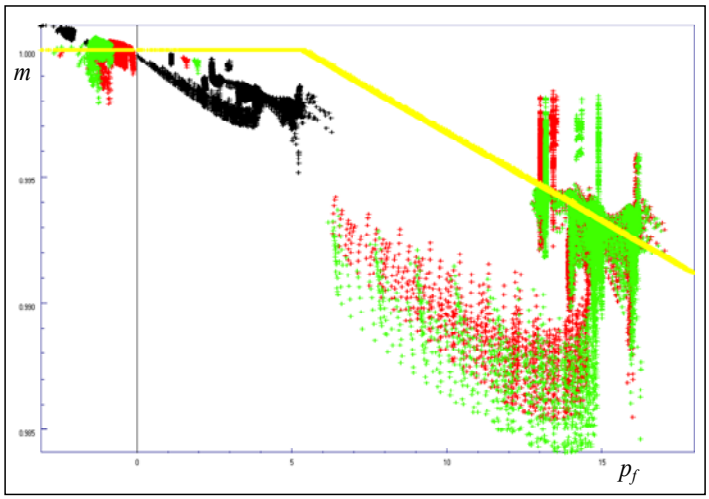


Figure A3. Original soil region 3 (Transition zone)

Note: The figure reveals two distinctly different trends – an indication that the transition zones would perhaps have been better defined as two zones, one for transition towards north-south fractures, and one towards east-west fractures.

See also comment to Figures B9, B10.

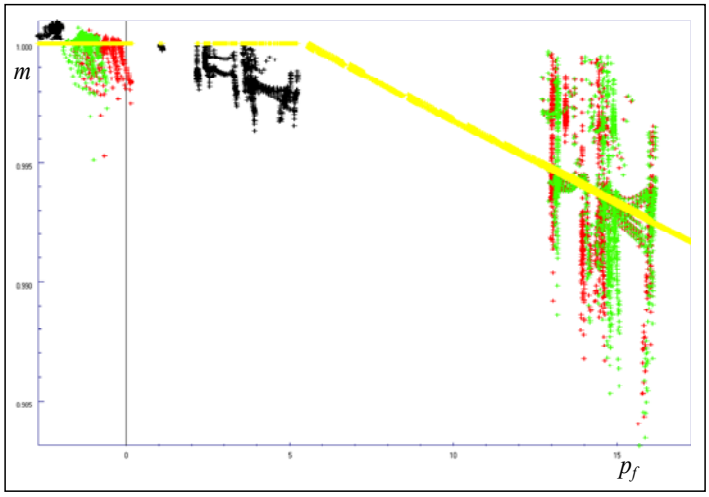


Figure A4. Original soil region 4 (North-South fractures)

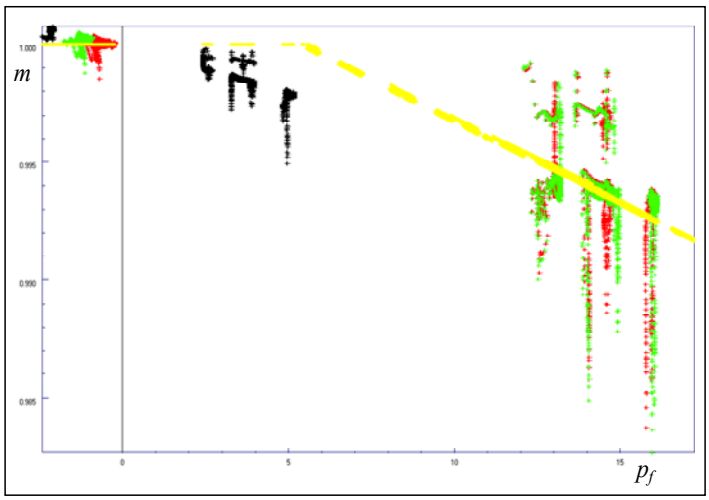


Figure A5. Original soil region 5 (East-West fractures)

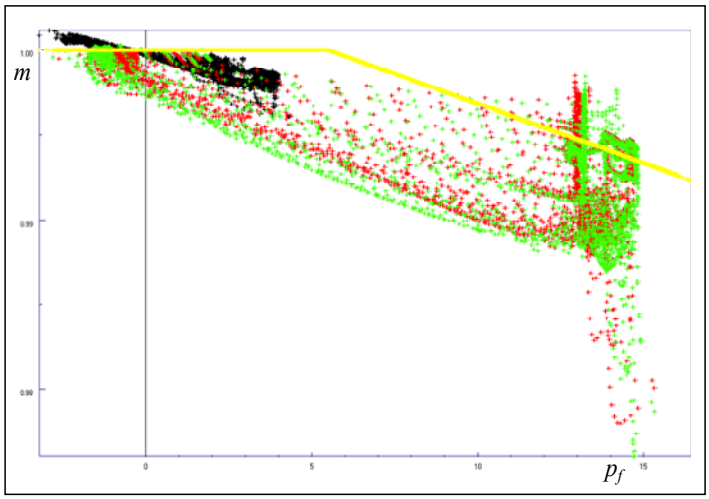
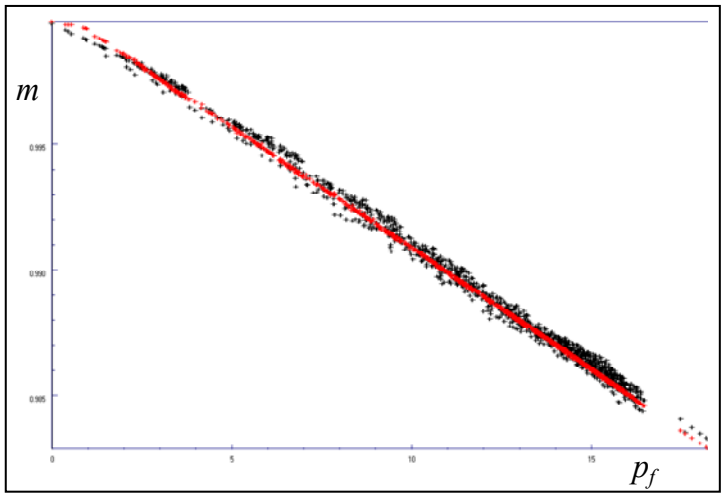


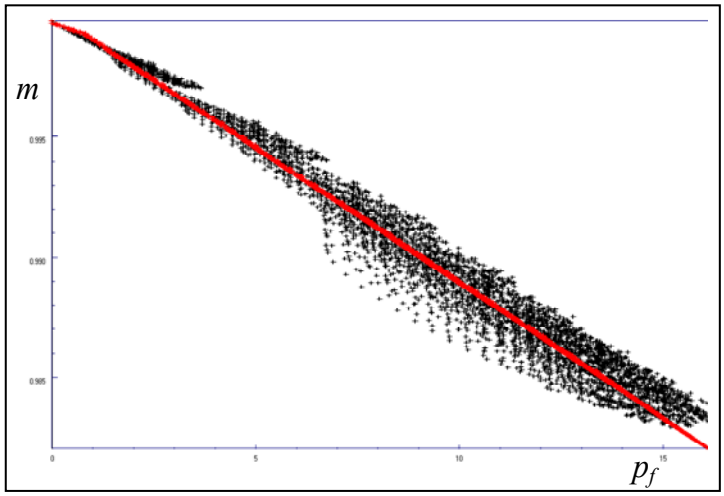
Figure A6. Original soil region 6 (Pinchout layer)

Appendix B. Results for individual (pseudo) regions

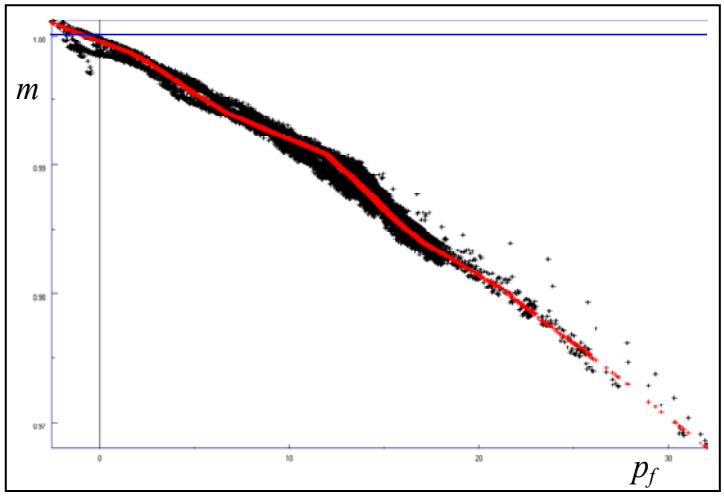
Except where otherwise noticed, black is used for the Visage-computed m_ε point cloud of the sub-region in question, and red for the corresponding ECLIPSE-computed m_{pf} point cloud. All the figures show the points for all 12 stress steps. Hence, the final match between exact and flow simulator computed compaction is shown in the figures, but also apparent is the construction of the m_{pf} curves from the sub-region point clouds. Special considerations in the construction process will be commented on. (The horizontal blue line, where present, is the line $m = 1$.)



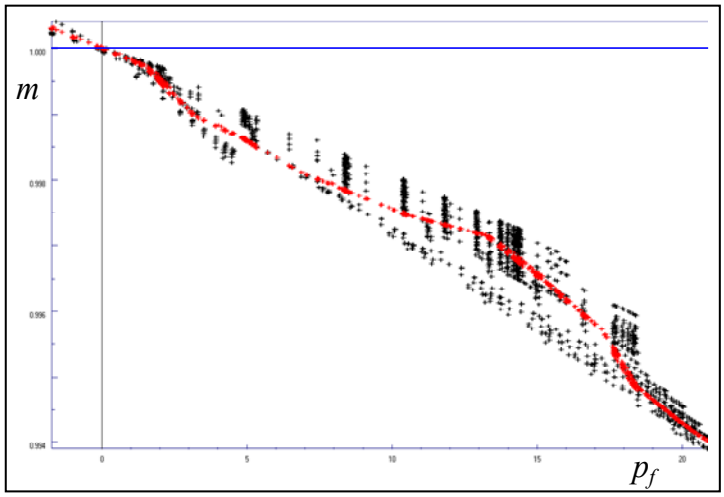
B1. Sub-region 39 (from transition)
Straightforward construction and excellent match. It should be noted that the majority of the sub-regions are matched as good as this one, and therefore are not further discussed, as we concentrate on those with interesting features.



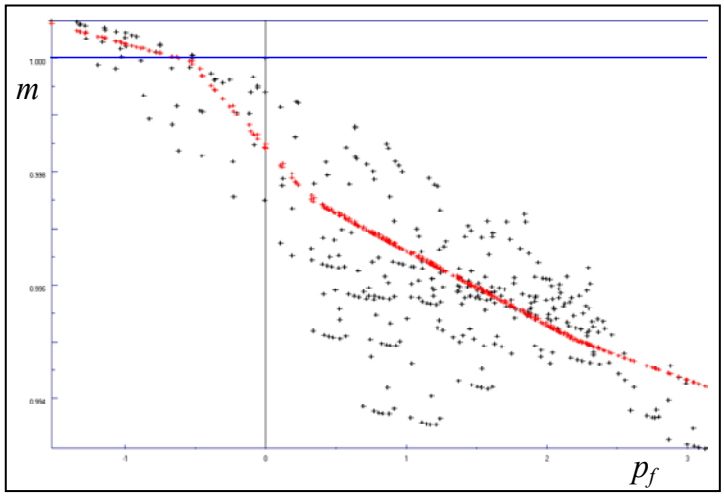
B2. Sub-region 36 (from transition)
Also reasonably straightforward, but with a somewhat larger spread than Figure B1. This is due to the sub-region being at an internal soil boundary, and hence contains some outliers.



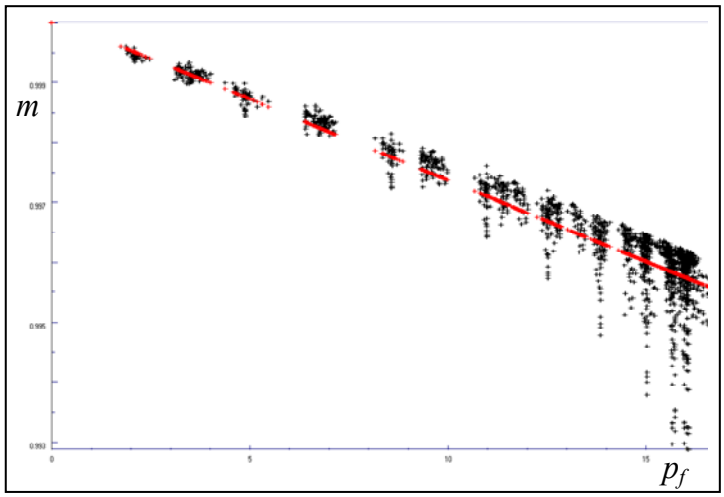
B3. Sub-region 13 (from matrix)
Relatively well defined curve and good match. Some points near load zero have a deviating trend, which has been disregarded.



B4. Sub-region 24 (from matrix)
Somewhat larger spread. The piecewise regression honours the trends from the densest parts of the point clouds, so one trend (the lower) has been missed.

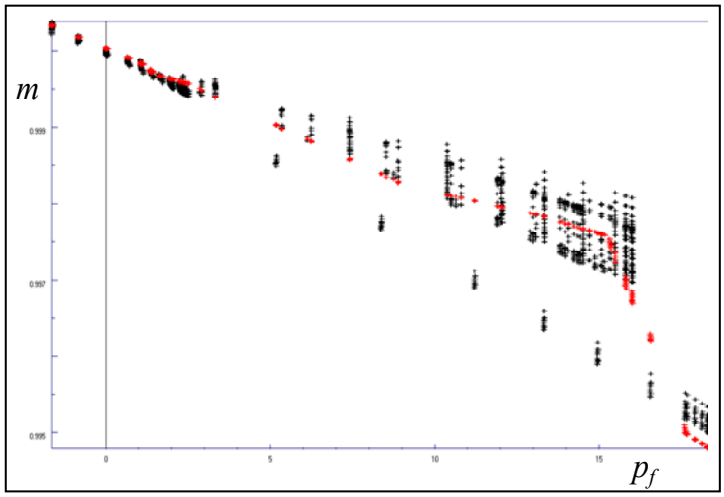


B5. Sub-region 1 (from matrix)
Construction is based on relatively few points, and being a sub-region boundary, many of the points are outliers (larger spread is typical for boundary regions, whether external or internal). Note that the curve does *not* pass through (0, 1)

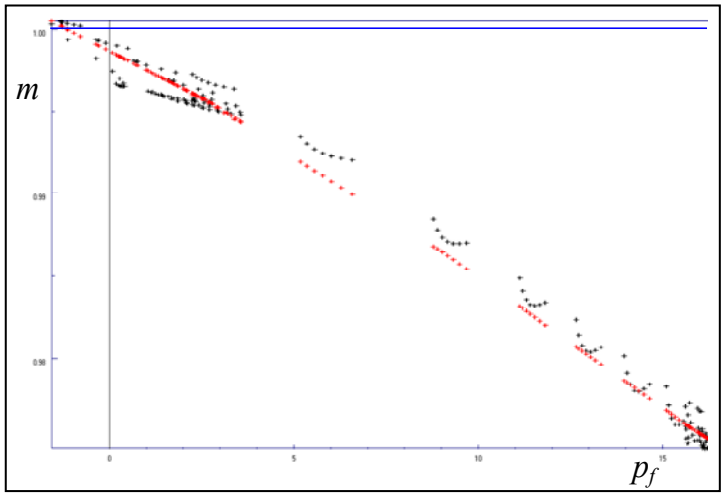


B6: 26

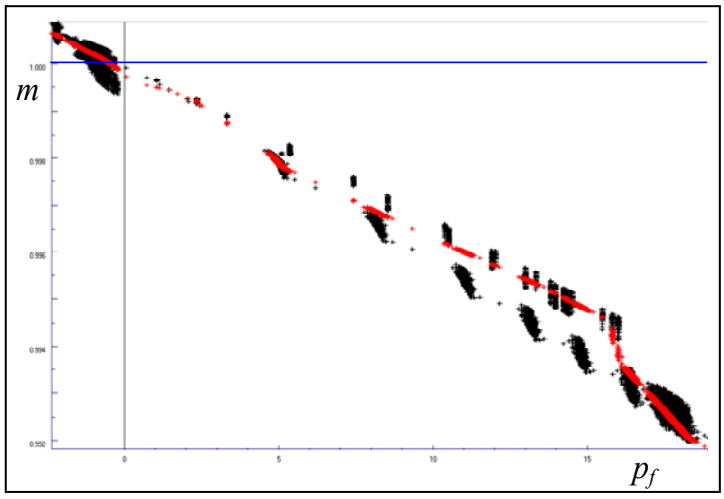
B6. Sub-region 26 (from hardground)
Another boundary sub-region (internal), neighbour to the lower envelope. This region contains especially many outliers. Note how these have little influence on the curve, due to sparsity.



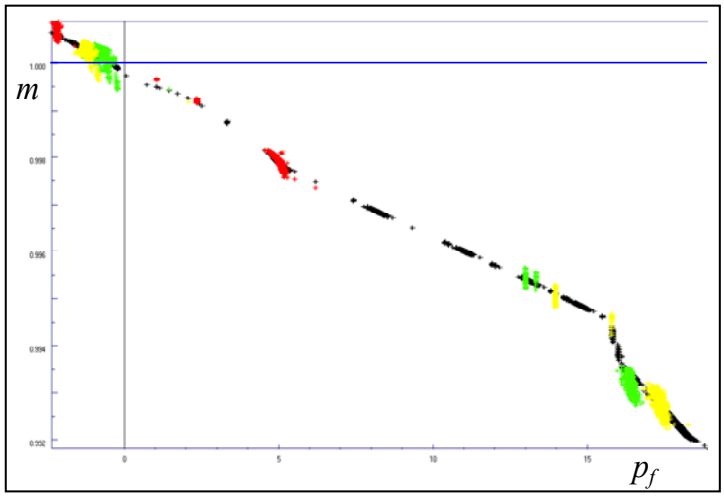
B7. Sub-region 60 (from transition)
Lower envelope boundary. For the most part, the outliers have been disregarded, but in a small interval they have been erroneously honoured, due to lack of points on the "main" trend. (If the load range in the tuning run is sufficiently large, the "wrong" part of the curve will never be used anyway.)



B8. Sub-region 112 (from pinchout)
Construction based on fewer points than desired. Such situations are handled by extending the sub-intervals for regression until a certain trend has been detected. (Extension both w.r.t. the number of points and length of interval)



B9. Sub-region 55 (from transition)
This is a case where apparently an erroneous trend has been computed, as the "lower" trend should obviously have been selected (larger point density). The "paradox" is resolved in the next figure

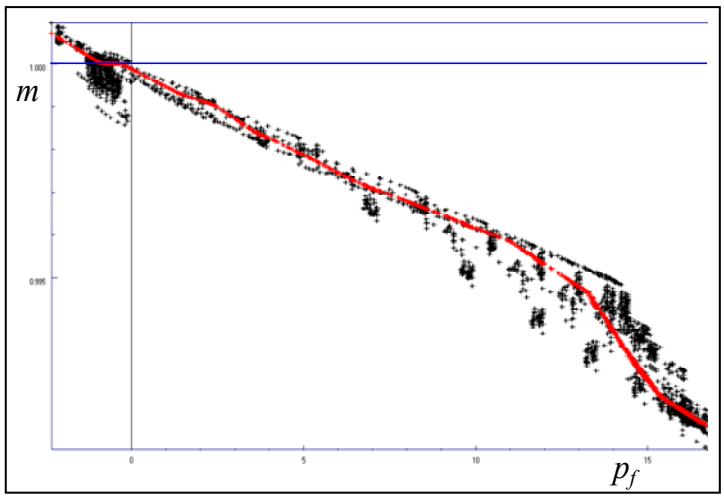


B10. Sub-region 55 (from transition)

The same sub-regions as in Figure B9, but now only the point clouds that were actually used in the construction are shown (tuning run stress steps 1, 2, and 3 in red, green, yellow, m_{pf} in black). So although the constructed m_{pf} undoubtedly is wrong, it honours the tuning run results accurately. Cases like this may obviously occur now and then. In this case, the most likely explanation is,

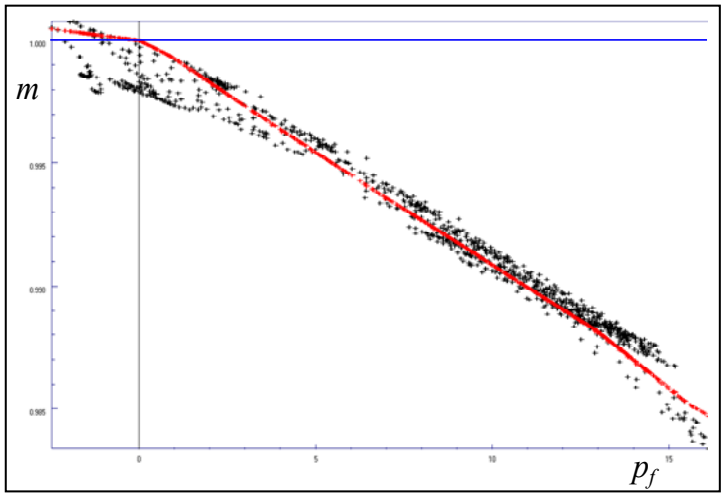
The model includes different soil descriptions for the North-South and East-West fractures, but only one transition zone has been defined. In Figure B9 the two distinct trends can be attributed to transition matrix ↔ N/S fractures, and matrix ↔ E/W fractures.

But requesting a single *m*-curve forces the procedure to choose one of the trends. The ambiguity could probably have been resolved by splitting the transition zone into N/S transition and E/W transition. Hence, as a rule of thumb, sufficiently many soil regions should be defined initially.
(Ref. also Figure A3)



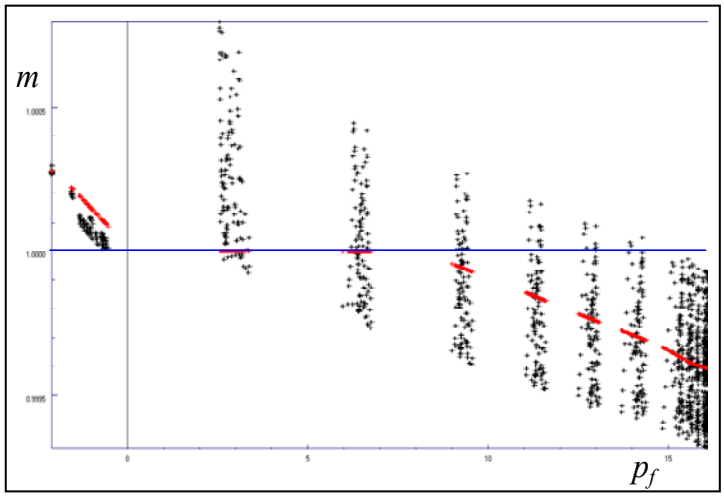
B11. Sub-region 131 (from pinchout)

Large spread, and partly not-so-well defined trend. Main illustrative point is the large "blob" of points in the unloading interval, showing that this sub-region has been subject to much unloading and reloading, which does not contribute to the determination of *m*-curve trends.



B12. Sub-region 120 (from pinchout)

The irregular unloading and reloading extends far into the load interval, and contributes to obscure the actual trend. In such cases, the algorithm searches for false trends and discards all the points which are classified as secondary loading.



B13. Sub-region 35 (from hardground)

Internal boundary, neighbour to upper envelope. In addition to the large spread, many points have *m*-values larger than unity, although the load is positive. This is clearly due to *dilation*, which cannot be modelled by the flow simulator rock tables (pore volume multipliers must be monotonic non-increasing with load). The best that can be done is to set $m = 1$ if the computed regression is larger than unity. In most cases, this will not be a large source of error, since the stress simulator will compute the correct compaction anyway. However, this is an example where accurate compaction cannot be obtained from the flow simulator, which is why omitting the rock mechanics simulator part should be discouraged.