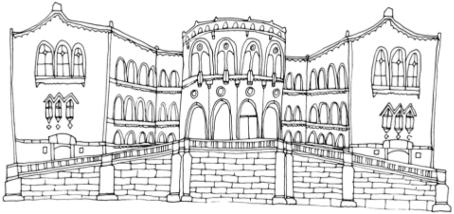
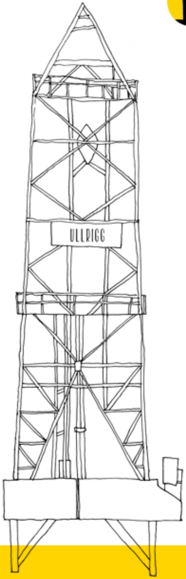
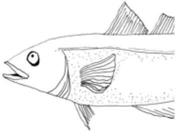
History matching real production and seismic data for the Norne field

EnKF Workshop 2018

Rolf J. Lorentzen, Tuhin Bhakta, Dario Grana, Xiaodong Luo, Randi Valestrand, Geir Nævdal, Ivar Sandø







Introduction

- The full norne model is history matched using real production and seismic data
- Initial ensemble generated using Gaussian random fields
- Updates PORO, PERMX, NTG, MULTZ, MULTFLT, MULTREGT, KRW/KRG, OWC
- Clay content defined as VCLAY = 1 - NTG
- Sequential assimilation (production $\rightarrow$ seismic)
- Seismic data inverted for acoustic impedance at four points in time
- Iterative ensemble smoother, RLM-MAC, used (Luo et. al, SPE-176023-PA)
- Sparse representation using wavelets (data reduced by 86 %)
- Correlation based localization

Seismic data inversion and transformation

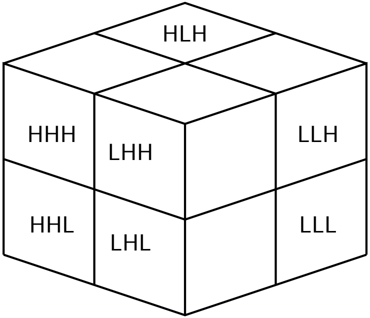
- Time shift correction:
  Alfonzo et al. 2017
- Linearized Bayesian approach:
  Buland and Omre, 2003: $S_{\mathrm{base}} = Gy_{\mathrm{base}} + e$
- Time to depth conversion:
  Provided Norne velocity model
- Upscaling:
  Petrel software
- Difference and averaging:
  $\overline{\Delta z}_p^o$

Petro-elastic model

- Estimate mineral bulk and shear moduli:
  $[K_s, G_s] \leftarrow \mathrm{Hashin-Shtrikman}(K_{\mathrm{quartz}}, G_{\mathrm{quartz}}, K_{\mathrm{clay}}, G_{\mathrm{clay}}, V_{\mathrm{clay}})$
- Dry rock bulk and shear moduli (empirical):
  $[K_{\mathrm{dry}}, G_{\mathrm{dry}}] \leftarrow f(p, p_{\mathrm{ini}}, \phi)$
- Fluid substitution:
  $[K_{\mathrm{sat}}, G_{\mathrm{sat}}] \leftarrow \mathrm{Gassman}(K_{\mathrm{dry}}, G_{\mathrm{dry}}, K_s, G_s)$
- P-wave velocity and rock density:
  $[v_p, \rho_{\mathrm{sat}}] \leftarrow \mathrm{Mavko}(K_{\mathrm{sat}}, G_{\mathrm{sat}})$
  $z_p = v_p \times \rho_{\mathrm{sat}}$

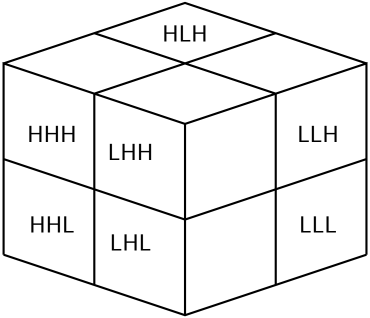
Sparse representation and image denoising

1. Transform seismic observations:
   $c_S \leftarrow \mathrm{DWT}(\overline{\Delta z}_p^o)$



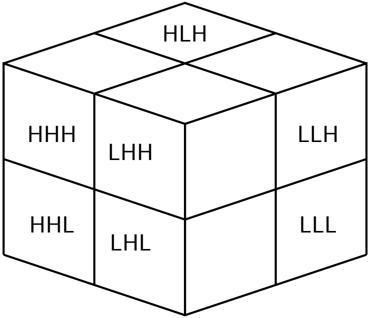
Sparse representation and image denoising

1. Transform seismic observations:
   $c_S \leftarrow \mathrm{DWT}(\overline{\Delta z}_p^o)$
2. Estimate noise in each subband (MAD):
   $\sigma_S = \mathrm{median}(|c_S - \mathrm{median}(c_S)|)/0.6745$



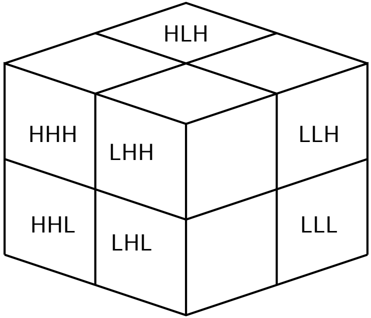
Sparse representation and image denoising

1. Transform seismic observations:
   $c_S \leftarrow \mathrm{DWT}(\overline{\Delta z}_p^o)$
2. Estimate noise in each subband (MAD):
   $\sigma_S = \mathrm{median}(|c_S - \mathrm{median}(c_S)|)/0.6745$
3. Compute standard deviation for coefficients:
   $\hat{\sigma}_S = \mathrm{std}(c_S)$



Sparse representation and image denoising

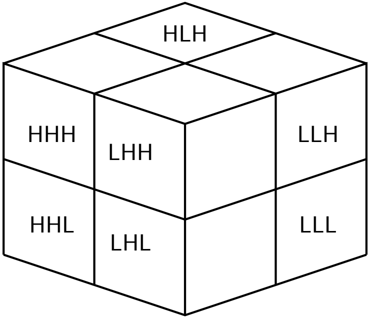
1. Transform seismic observations:
   $c_S \leftarrow \mathrm{DWT}(\overline{\Delta z}_p^o)$
2. Estimate noise in each subband (MAD):
   $\sigma_S = \mathrm{median}(|c_S - \mathrm{median}(c_S)|)/0.6745$
3. Compute standard deviation for coefficients:
   $\hat{\sigma}_S = \mathrm{std}(c_S)$
4. Compute truncation value (Bayesian shrinkage):
   $T_S = \frac{\sigma_S^2}{\sqrt{|\sigma_S^2 - \hat{\sigma}_S^2|}}$



IRIS

Sparse representation and image denoising

1. Transform seismic observations:
   $c_S \leftarrow \mathrm{DWT}(\overline{\Delta z_p^o})$
2. Estimate noise in each subband (MAD):
   $\sigma_S = \mathrm{median}(|c_S - \mathrm{median}(c_S)|)/0.6745$
3. Compute standard deviation for coefficients:
   $\hat{\sigma}_S = \mathrm{std}(c_S)$
4. Compute truncation value (Bayesian shrinkage):
   $T_S = \frac{\sigma_S^2}{\sqrt{|\sigma_S^2 - \hat{\sigma}_S^2|}}$
5. Apply hard thresholding:
   $c_s \rightarrow c_s > T_S, \quad d^o = c(\mathcal{I})$



Ensemble smoother

$$m_j^{i+1} = m_j^i + S_m^i (S_d^i)^T [S_d^i (S_d^i)^T + \gamma^i C_d]^{-1} \times [d^o + \epsilon_j - d_j^i]$$

$\downarrow$ TSVD

$$m_j^{i+1} = m_j^i + \tilde{K}^i \Delta \tilde{d}_j^i$$

$\Delta \tilde{d}_j^i \in \mathbb{R}^{p \times 1}, \ p \leq N$: Projected ("effective") data innovation.

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l): $\rho_{kl} \in \mathbb{R}^{N_m \times p}$

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l): $\rho_{kl} \in \mathbb{R}^{N_m \times p}$
2. Transform all correlations for each observation (ρ_l), and return high-frequency coefficients: $c_l^H \leftarrow \mathrm{DWT}(\rho_l)$

IRIS

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l):
   $\rho_{kl} \in \mathbb{R}^{N_m \times p}$
2. Transform all correlations for each observation (ρ_l), and return high-frequency coefficients:
   $c_l^H \leftarrow \mathrm{DWT}(\rho_l)$
3. Estimate noise in coefficients (MAD):
   $\sigma_l = \mathrm{median}(|c_l^H - \mathrm{median}(c_l^H)|)/0.6745$

IRIS

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l): $\rho_{kl} \in \mathbb{R}^{N_m \times p}$
2. Transform all correlations for each observation (ρ_l), and return high-frequency coefficients: $c_l^H \leftarrow \mathrm{DWT}(\rho_l)$
3. Estimate noise in coefficients (MAD): $\sigma_l = \mathrm{median}(|c_l^H - \mathrm{median}(c_l^H)|)/0.6745$
4. Compute truncation value (universal rule): $\lambda_l = \max(\sqrt{2 \ln n(\rho_l)}\sigma_l)$

IRIS

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l): $\rho_{kl} \in \mathbb{R}^{N_m \times p}$
2. Transform all correlations for each observation (ρ_l), and return high-frequency coefficients: $c_l^H \leftarrow \mathrm{DWT}(\rho_l)$
3. Estimate noise in coefficients (MAD): $\sigma_l = \mathrm{median}(|c_l^H - \mathrm{median}(c_l^H)|)/0.6745$
4. Compute truncation value (universal rule): $\lambda_l = \max(\sqrt{2 \ln n(\rho_l)}\sigma_l)$
5. Compute truncation matrix: $\xi_{kl} = 1, \ \mathrm{if} |\rho_{kl}| \geq \lambda_l, \ \ 0 \ \ \mathrm{otherwise}$

IRIS

Correlation based localization

1. Compute sample correlation between parameters (k) and "effective" measurements (l):
   $\rho_{kl} \in \mathbb{R}^{N_m \times p}$
2. Transform all correlations for each observation (ρ_l), and return high-frequency coefficients:
   $c_l^H \leftarrow \mathrm{DWT}(\rho_l)$
3. Estimate noise in coefficients (MAD):
   $\sigma_l = \mathrm{median}(|c_l^H - \mathrm{median}(c_l^H)|)/0.6745$
4. Compute truncation value (universal rule):
   $\lambda_l = \max(\sqrt{2 \ln n(\rho_l)}\sigma_l)$
5. Compute truncation matrix:
   $\xi_{kl} = 1, \text{ if} |\rho_{kl}| \geq \lambda_l, \; 0 \text{ otherwise}$
6. Updated Kalman gain matrix (see also Luo and Bhakta, 2017):
   $\hat{K} = \xi \circ \tilde{K}$

Measurement operator

The observation operator $\mathcal{G}$ comprises several steps summarized as:

1. running the reservoir simulator using m_j to compute dynamic variables (pressure and saturation)

Measurement operator

The observation operator $\mathcal{G}$ comprises several steps summarized as:

1. running the reservoir simulator using m_j to compute dynamic variables (pressure and saturation)
2. running the PEM to compute the acoustic impedance, $z_{p,j}$, at all survey times

Measurement operator

The observation operator $\mathcal{G}$ comprises several steps summarized as:

1. running the reservoir simulator using m_j to compute dynamic variables (pressure and saturation)
2. running the PEM to compute the acoustic impedance, $z_{p,j}$, at all survey times
3. compute differences and average over formation layers to get $\overline{\Delta z}_{p,j}$

Measurement operator

The observation operator $\mathcal{G}$ comprises several steps summarized as:

1. running the reservoir simulator using m_j to compute dynamic variables (pressure and saturation)
2. running the PEM to compute the acoustic impedance, $z_{p,j}$, at all survey times
3. compute differences and average over formation layers to get $\overline{\Delta z}_{p,j}$
4. applying the DWT to get c_j

Measurement operator

The observation operator $\mathcal{G}$ comprises several steps summarized as:

1. running the reservoir simulator using m_j to compute dynamic variables (pressure and saturation)
2. running the PEM to compute the acoustic impedance, $z_{p,j}$, at all survey times
3. compute differences and average over formation layers to get $\overline{\Delta z}_{p,j}$
4. applying the DWT to get c_j
5. using the leading indices $\mathcal{I}$ to get $d_j = c_j(\mathcal{I})$

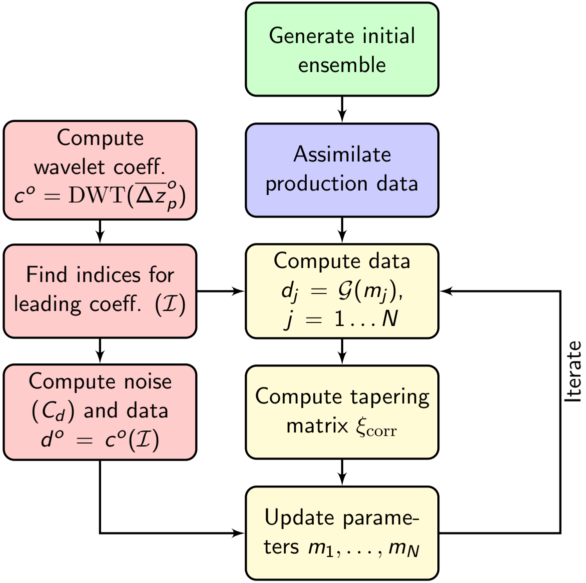
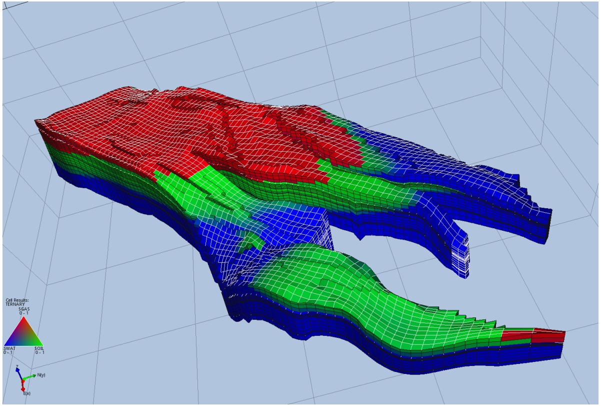
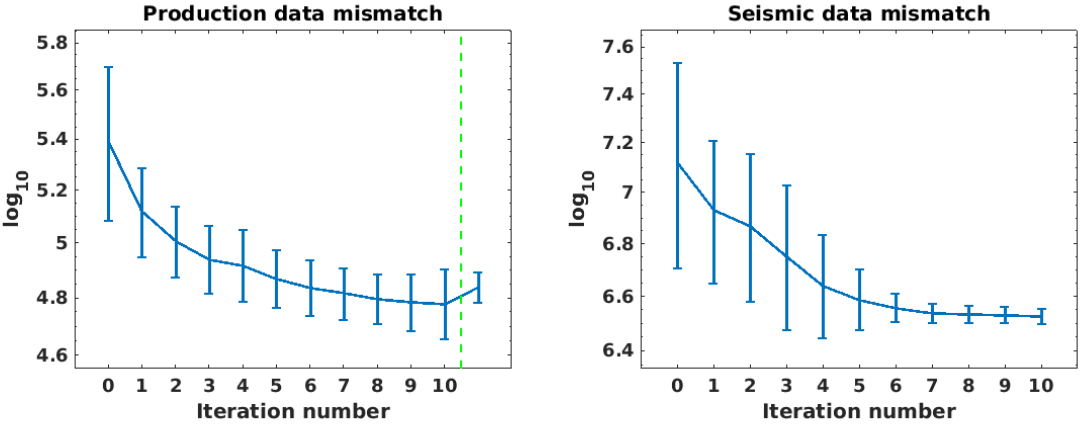


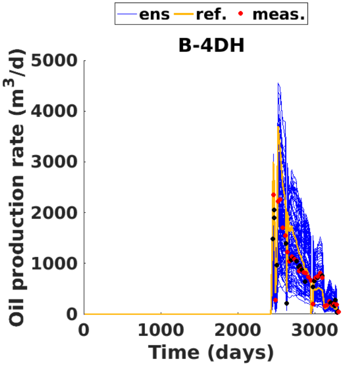
Figure: Workflow for assimilating seismic data.

Norne field

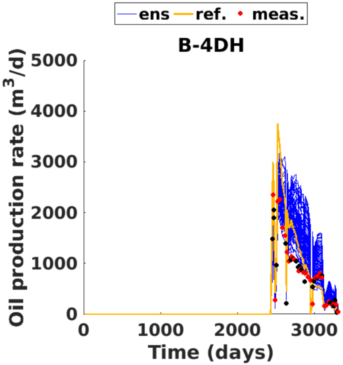
- Grid size:
  46 x 112 x 22
  (113344)
- Active cells:
  44927
- Wells:
  9 injectors,
  27 producers
- Production: 3312 days



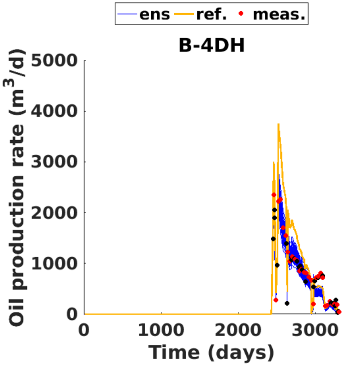

Production data mismatch
5.8
5.6
5.4
5.2
5
4.8
4.6
log10
0 1 2 3 4 5 6 7 8 9 10
Iteration number
Seismic data mismatch
7.6
7.4
7.2
7
6.8
6.6
6.4
log10
0 1 2 3 4 5 6 7 8 9 10
Iteration number



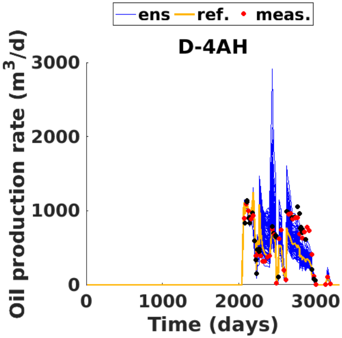
Initial



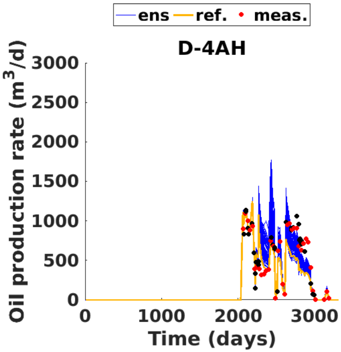
Production



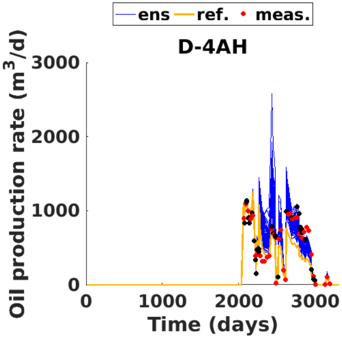
Seismic



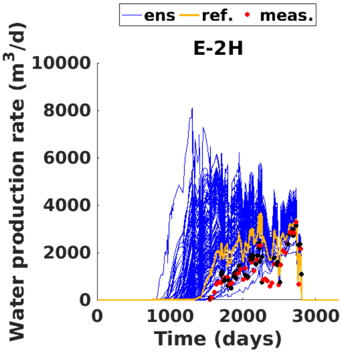
Initial



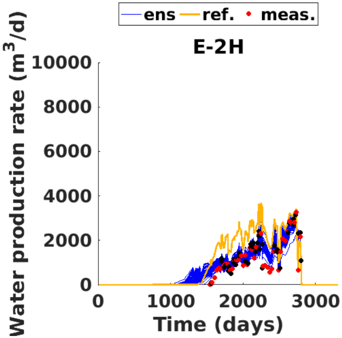
Production



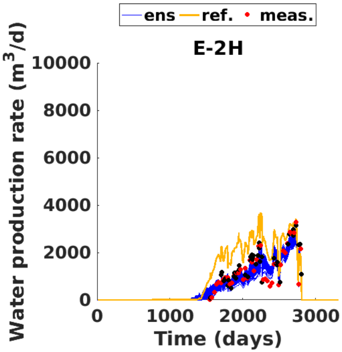
Seismic



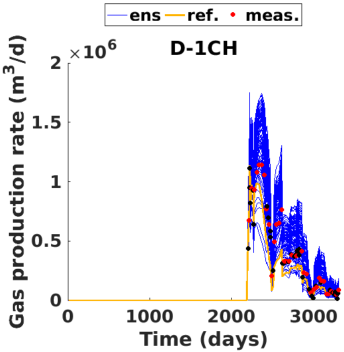
Initial



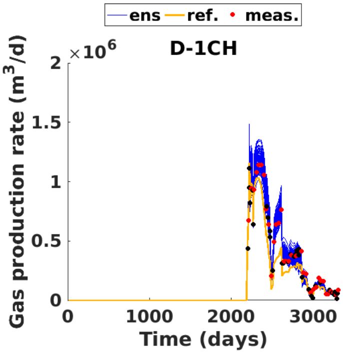
Production



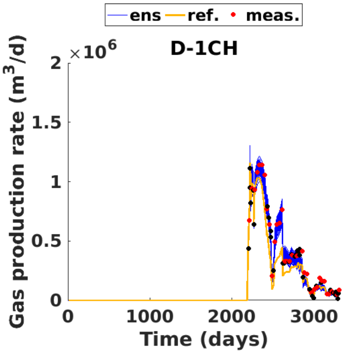
Seismic



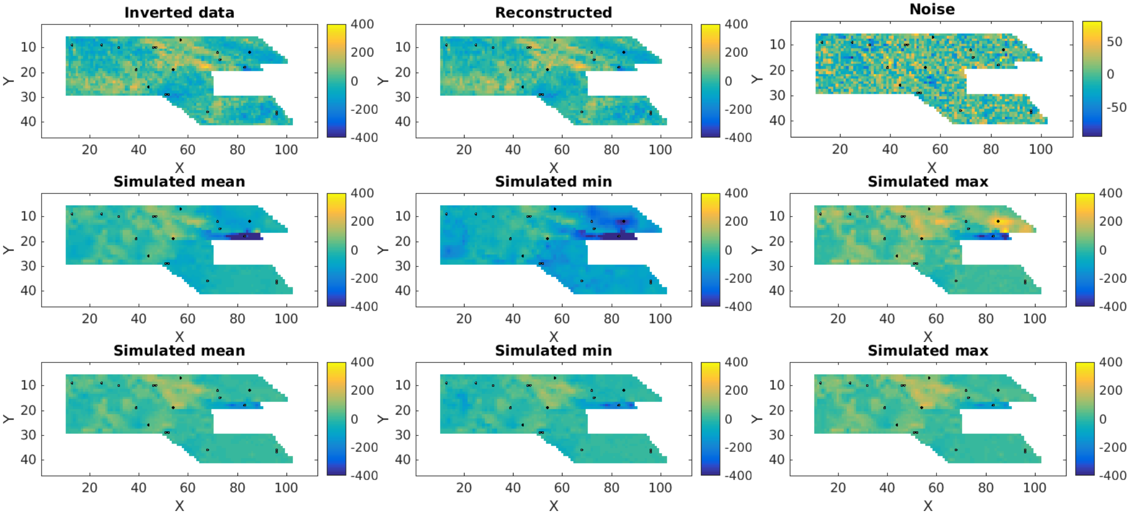
Initial



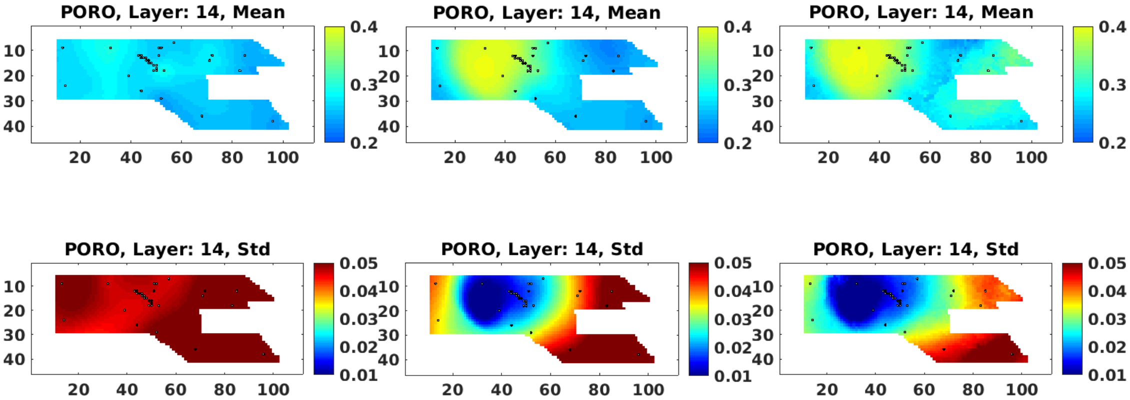
Production



Seismic



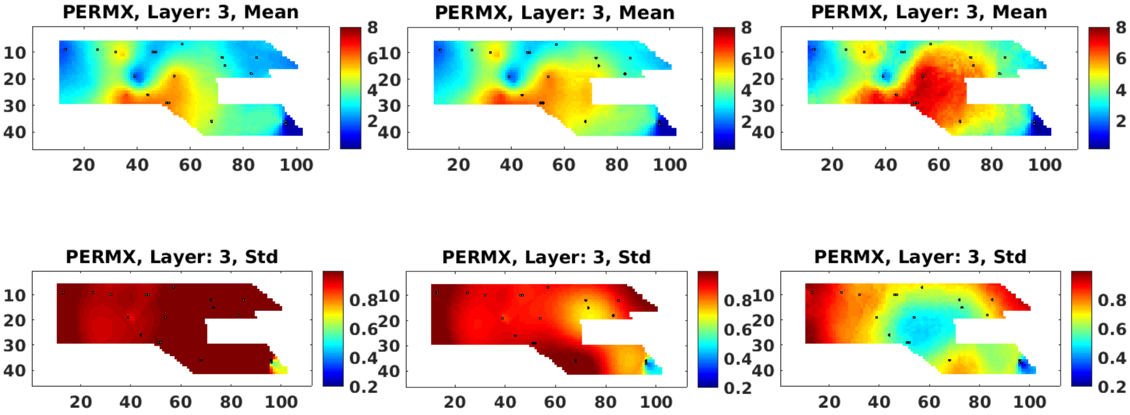
Top: real data. Middle: production. Bottom: seismic.



Initial

Production

Seismic

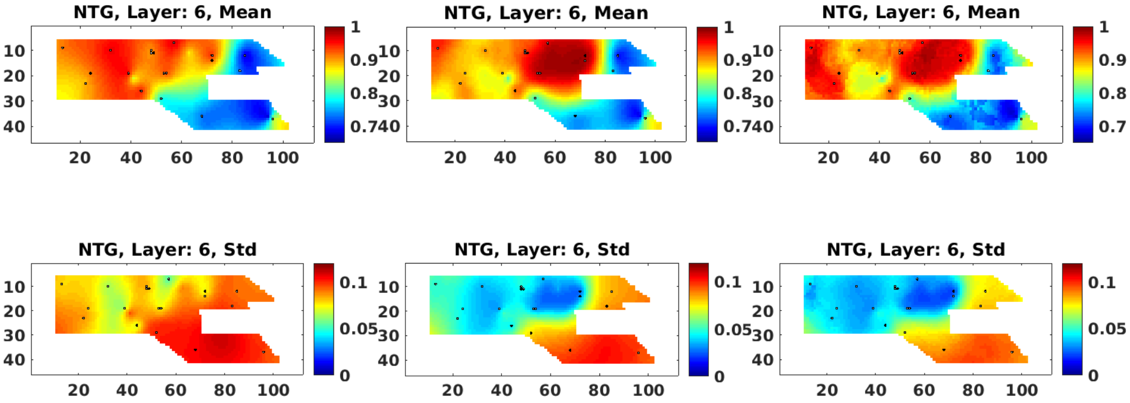


Initial

Production

Seismic

IRIS



Initial

Production

Seismic

Summary / Conclusions

- A workflow for history matching real production and seismic data is presented
- Clay content and other petrophysical parameters updated
- Data match improved for both production and seismic data
- Updated static fields are geologically credible
- Potential for simulating infill wells or EOR strategies

Acknowledgments

We thank

- Schlumberger and CGG for providing academic software licenses to ECLIPSE and HampsonRussell, respectively.
- Main financial support from Eni, Petrobras, and Total, as well as the Research Council of Norway (PETROMAKS2).
- Partial financial support from The National IOR Centre of Norway.