Identification of reservoir parameters using flexible representations

I. Berre[1] M. Lien[2] T. Mannseth[2,1]

[1]Department of Mathematics, University of Bergen

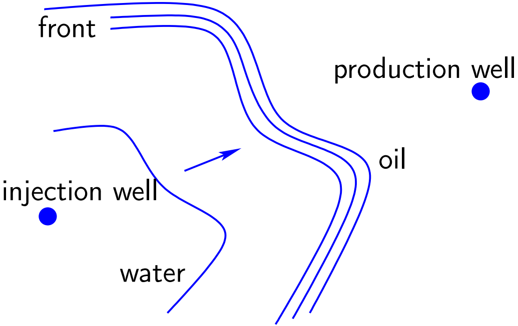
[2]Center for Integrated Petroleum Research, University of Bergen

Bergen Workshop, October 2010

Two-phase flow

Forward model

$$\phi \frac{\partial S}{\partial t} - \nabla \cdot (k\lambda_1 \nabla P_1) = q_1$$
$$-\phi \frac{\partial S}{\partial t} - \nabla \cdot (k\lambda_2 \nabla P_2) = q_2$$



Inverse problem

Inverse problem

Estimate the absolute permeability $k(\mathbf{x})$ based on pressure data in wells when the other model specifications are known.

Common characteristics

- Low spatial data density
- Limited prior information

$\Rightarrow$ But sufficient for identification of large-scale structures

Additional challenges

- Nonlinearity
- Resolving power of data is unknown

Minimization problem

Objective function

Minimize

$$J(k) = [d - m(k)]^T C^{-1} [d - m(k)]$$

with respect to k

Represent $k(x)$ with a function $p(x)$ with a finite number of degrees of freedom.

Representation of the parameter function

Tradeoff

finer spatial resolution $\Leftrightarrow$ increased parameter uncertainty

Relationship

small scale perturbations in $p(x)$ $\Leftrightarrow$ low sensitivity $\Leftrightarrow$ high model nonlinearity

Representation of the parameter function

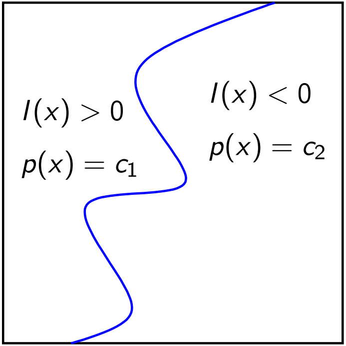
Linear representation (classical inverse problem)

$$p(x) = \Sigma_{j=1}^{N} c_j \Psi_j(x)$$

Composite representation (extended inverse problem)

$$p(x) = c_1 E_1\big(I(x)\big) + c_2 E_2\big(I(x)\big), \quad I(x) = \sum_{k=1}^{K} a_k \theta_k(x)$$

Special case: Level-set representation



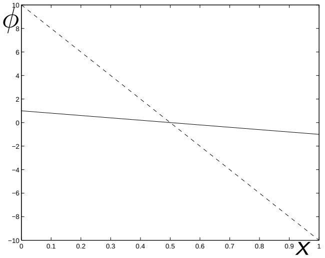
Level-set representation of permeability

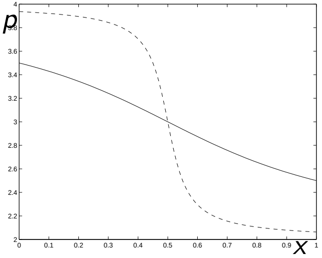
$$p(\mathbf{x}) = c_1 \underbrace{H(I(\mathbf{x}))}_{E_1(I(x))} + c_2 [\underbrace{1 - H(I(\mathbf{x}))}_{E_2(I(x))}]$$

Special case: Smooth Parameter Representation

Parameter function

$$p(x) = c_1 \Big[\underbrace{\frac{1}{2} + \frac{1}{\pi} \tan^{-1} \big(I(x) \big)}_{E_1(I(x))} \Big] + c_2 \Big[\underbrace{\frac{1}{2} - \frac{1}{\pi} \tan^{-1} \big(I(x) \big)}_{E_2(I(x)} \Big]$$





Parameter identification

Identifying parameterization structure and reducing nonlinearity

Composite representation

$$p(x) = c_1 E_1\big(I(x)\big) + c_2 E_2\big(I(x)\big), \quad I(x) = \sum_{k=1}^{K} a_k \theta_k(x)$$

Minimize

$$J(\eta) = [d - m(\eta)]^T C^{-1} [d - m(\eta)], \quad \eta = [c_1, c_2, a_1, \ldots, a_K]$$

Multi-level algorithm

Regular refinement of representation of $I(x)$

- lower sensitivity and lower nonlinearity for larger scales
- reduce the risk of getting trapped in local minima of the objective function

Optimization Method

Levenberg-Marquardt method

$\eta = [c_1, c_2, a_1, \ldots, a_K]$

$$\eta^{h+1} = \eta^h + \Delta\eta^h$$

$\Delta\eta^h$ is the solution of

$$\left(A^T A - \lambda^h I\right)\Delta\eta^h = A^T\left(d - m(\eta^h)\right),$$

Sensitivity matrix:

$$A = \left[\frac{dm_i}{d\eta_j}\right]_{i,j} = \frac{dm}{d\eta}$$

Sensitivity calculations

η	-	unknown coefficients in the representation
u	-	forward model solution vector from solving dynamic problem
m	-	model response (which is compared to data)

$$m = m(u; n), \quad u = u(\eta)$$

Sensitivity matrix

$$A = \left[\frac{dm_i}{d\eta_j}\right]_{i,j} = \frac{dm}{d\eta} = \frac{\partial m}{\partial u}\frac{du}{d\eta} + \frac{\partial m}{\partial \eta}$$

Direct method for sensitivity calculations

Discretized PDE's

$$F(u^{n+1}, u^n; \eta) = 0, \qquad (0)$$

Sensitivity equation

$$\underbrace{\frac{\partial F}{\partial u^{n+1}}}_{Jacobian} \frac{du^{n+1}}{d\eta} = -\frac{\partial F}{\partial u^n}\frac{du^n}{d\eta} - \frac{\partial F}{\partial \eta},$$

$\dim\eta = 2 + K$ linear systems (common coefficient matrix assembled when solving discretized PDE's)

Sensitivity calculations cont.

Chain rule

$$\frac{\partial F}{\partial \eta} = \frac{\partial F}{\partial p}\frac{dp}{d\eta}, \quad \frac{dp}{d\eta} = \left[\frac{\partial p}{\partial c_1}, \frac{\partial p}{\partial c_2}, \frac{\partial p}{\partial a_1}, \ldots, \frac{\partial p}{\partial a_K}\right]$$

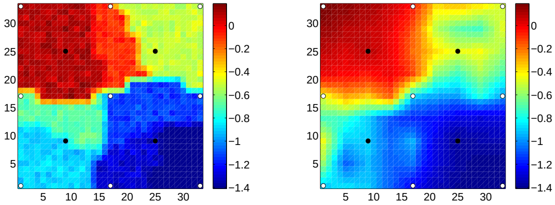
$$p(x) = c_1 E_1\big(I(x)\big) + c_2 E_2\big(I(x)\big)$$

$$I(x) = \sum_{k=1}^{K} a_k \theta_k(x)$$

$$\Rightarrow \quad \frac{\partial p}{\partial c_j} = E_j(I), \quad \frac{\partial p}{\partial a_k} = \frac{\partial p}{\partial I}\frac{\partial I}{\partial a_k}$$

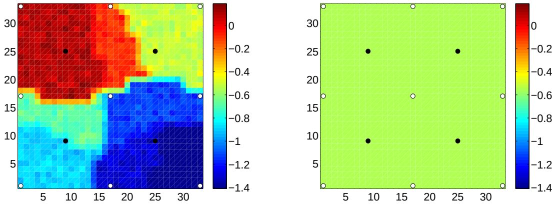
Numerical Results

Smooth Representation



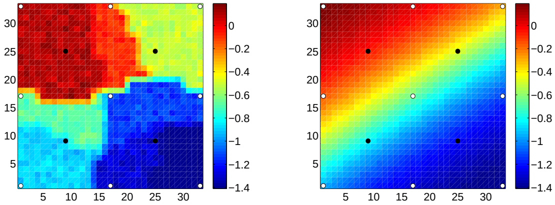
Numerical Results

Smooth Representation



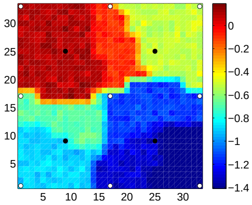
Numerical Results

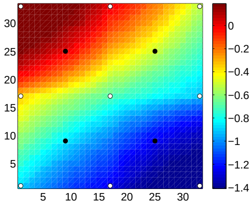
Smooth Representation



Numerical Results

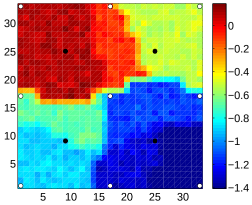
Smooth Representation

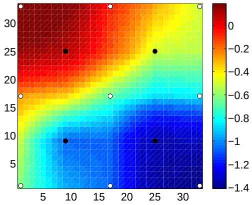




Numerical Results

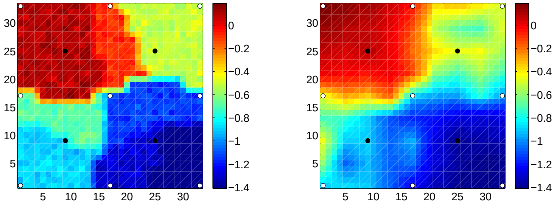
Smooth Representation





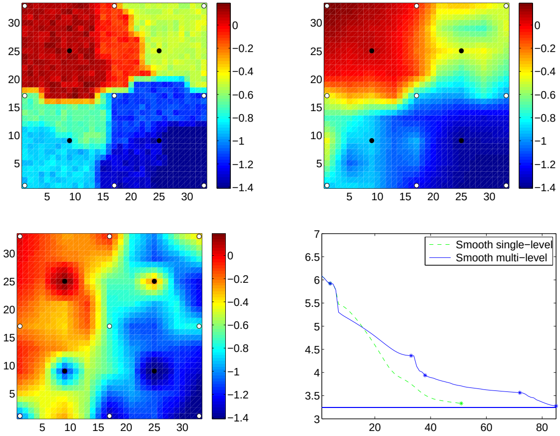
Numerical Results

Smooth Representation



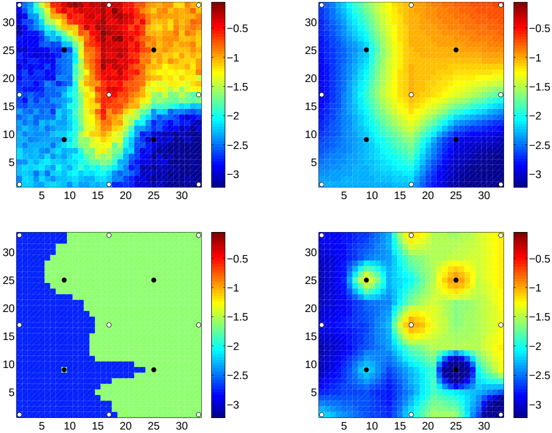
Numerical Results

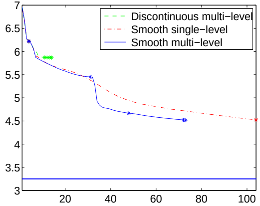
Smooth Representation



Numerical Results

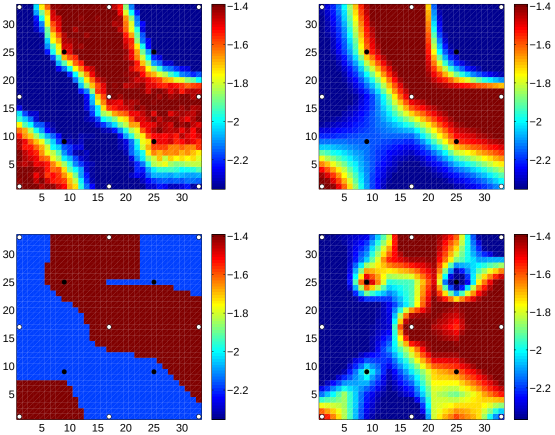
Smooth Representation

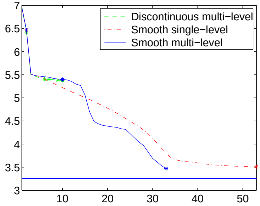




Numerical Results

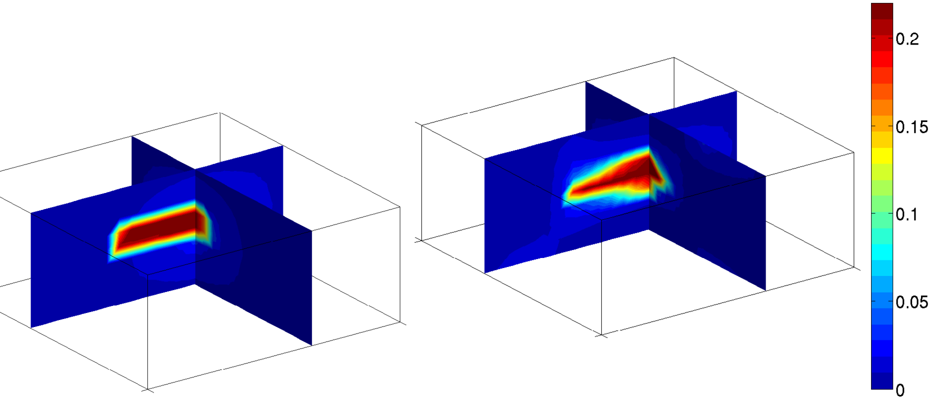
Smooth Representation





Numerical Results

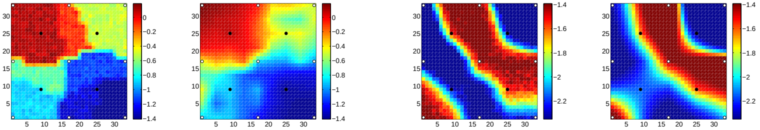
Estimation of electric conductivity based on low frequency electromagnetic data



25% noise

Conclusions

- The composite representation facilitates representation of smooth as well as sharp transitions between region of nearly constant parameter value with a low number of coefficients in the representation
- Gradual refinement stabilize the estimation and avoids local minima in parameter space
- Based on spatially sparsely distributed data, the method has the potential of recovering coarse permeability variations



Future work

Gradient/Sensitivity-based refinement

Objective function:

$$J(a,c) = [d - m(a,c)]^T C^{-1} [d - m(a,c)]$$

Compare

Gradient (magnitude of steepest descent)

$$\nabla J_a = \nabla_I J \frac{\partial I}{\partial a},$$

or approximation to objective function based on a linearized model response

$$J(a,c) \approx [d - m(a,c)]^T (C^{-1} - [C^{-1} A (A^T C A)^{-1} A^T C^{-1}])[d - m(a,c)]$$

for different choices of $\{\theta_k(x)\}_{k=1}^K$.

Testing of adaptive refinement

Linear expansion representation

Gaussian radial basis functions

$$p(x) = \Sigma_{j=1}^{N} c_j \Psi_j(x), \quad \Psi_j(x) = \exp(-\frac{1}{2\sigma_j^2}(x - x_j)^2)$$