# Data-Assimilation Fundamentals:

## A Unified Formulation of the State and Parameter Estimation Problem

Geir Evensen, Femke C. Vossepoel, and Peter Jan van Leeuwen

Available from https://github.com/geirev/Data-Assimilation-Fundamentals.git

Springer Textbooks in Earth Sciences, Geography and Environment

Geir Evensen · Femke C. Vossepoel · Peter Jan van Leeuwen

**Data Assimilation Fundamentals**

This open-access textbook's significant contribution is the unified derivation of data-assimilation techniques from a common fundamental and optimal starting point, namely Bayes' theorem. Unique for this book is the "top-down" derivation of the assimilation methods. It starts from Bayes theorem and gradually introduces the assumptions and approximations needed to arrive at today's popular data-assimilation methods. This strategy is the opposite of most textbooks and reviews on data assimilation that typically take a bottom-up approach to derive a particular assimilation method. E.g., the derivation of the Kalman Filter from control theory and the derivation of the ensemble Kalman Filter as a low-rank approximation of the standard Kalman Filter. The bottom-up approach derives the assimilation methods from different mathematical principles, making it difficult to compare them. Thus, it is unclear which assumptions are made to derive an assimilation method and sometimes even which problem it aspires to solve. The book's top-down approach allows categorizing data-assimilation methods based on the approximations used. This approach enables the user to choose the most suitable method for a particular problem or application. Have you ever wondered about the difference between the ensemble 4DVar and the "ensemble randomized likelihood" (EnRML) methods? Do you know the differences between the ensemble smoother and the ensemble-Kalman smoother? Would you like to understand how a particle flow is related to a particle filter? In this book, we will provide clear answers to several such questions. The book provides the basis for an advanced course in data assimilation. It focuses on the unified derivation of the methods and illustrates their properties on multiple examples. It is suitable for graduate students, post-docs, scientists, and practitioners working in data assimilation.



ISBN 978-3-030-96708-6

9 783030 967086

▸ springer.com

Evensen · Vossepoel · Leeuwen

Data Assimilation Fundamentals

TEXTBOOK

Geir Evensen · Femke C. Vossepoel
Peter Jan van Leeuwen

# Data Assimilation Fundamentals

## A Unified Formulation of the State and Parameter Estimation Problem

OPEN ACCESS

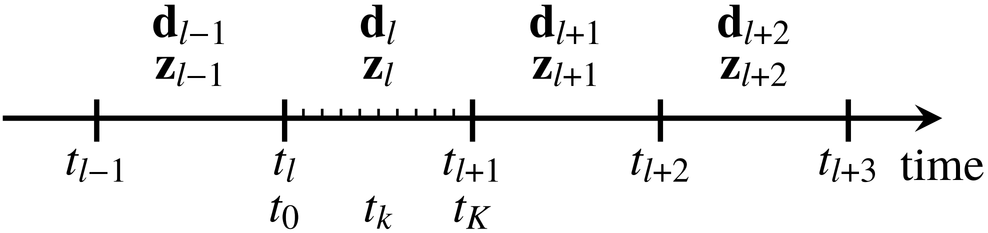
Springer

# We start from Bayes' theorem

$$\boxed{f(\mathcal{Z}|\mathcal{D}) = \frac{f(\mathcal{D}|\mathcal{Z})f(\mathcal{Z})}{f(\mathcal{D})}. \qquad (1)}$$

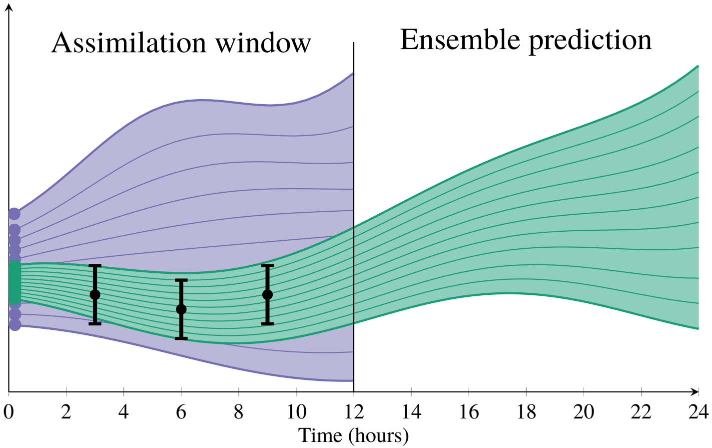
- $\mathcal{Z} = (\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_L)$ is the vector of state variables on all the assimilation windows.
- $\mathcal{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_L)$ is the vector containing all the measurements.

# Split time into data-assimilation windows




- We consider the DA problem for one single window.
- Errors propagate from one window to the next.

# Weather prediction configuration



# Model is Markov process

**Approximation 1 (Model is 1st-order Markov process)**

*We assume the dynamical model is a 1st-order Markov process.*

$$f(\mathbf{z}_l|\mathbf{z}_{l-1},\mathbf{z}_{l-2},\ldots,\mathbf{z}_0) = f(\mathbf{z}_l|\mathbf{z}_{l-1}), \tag{2}$$

# Independent measurements

**Approximation 2 (Independent measurements)**

*We assume that measurements are independent between different assimilation windows.*

Independent measurements have uncorrelated errors

$$f(\mathcal{D}|\mathcal{Z}) = \prod_{l=1}^{L} f(\mathbf{d}_l|\mathbf{z}_l). \tag{3}$$

# Bayes becomes

$$f(\mathcal{Z}|\mathcal{D}) \propto \prod_{l=1}^{L} f(\mathbf{d}_l|\mathbf{z}_l) \prod_{l=1}^{L} f(\mathbf{z}_l|\mathbf{z}_{l-1}) f(\mathbf{z}_0). \tag{4}$$

# Recursive form of Bayes

$$f(\mathbf{z}_1, \mathbf{z}_0|\mathbf{d}_1) = \frac{f(\mathbf{d}_1|\mathbf{z}_1)f(\mathbf{z}_1|\mathbf{z}_0)f(\mathbf{z}_0)}{f(\mathbf{d}_1)}, \tag{5}$$

$$f(\mathbf{z}_2, \mathbf{z}_1, \mathbf{z}_0|\mathbf{d}_1, \mathbf{d}_2) = \frac{f(\mathbf{d}_2|\mathbf{z}_2)f(\mathbf{z}_2|\mathbf{z}_1)f(\mathbf{z}_1, \mathbf{z}_0|\mathbf{d}_1)}{f(\mathbf{d}_2)}, \tag{6}$$

$$\vdots$$

$$f(\mathcal{Z}|\mathcal{D}) = \frac{f(\mathbf{d}_L|\mathbf{z}_L)f(\mathbf{z}_L|\mathbf{z}_{L-1})f(\mathbf{z}_{L-1}, \ldots, \mathbf{z}_0|\mathbf{d}_{L-1}, \ldots, \mathbf{d}_1)}{f(\mathbf{d}_L)}. \tag{7}$$

# Filtering assumption

**Approximation 3 (Filtering assumption)**

*We approximate the full smoother solution with a sequential data-assimilation solution. We only update the solution in the current assimilation window, and we do not project the measurement's information backward in time from one assimilation window to the previous ones.*

# Recursive Bayes' for filtering

$$
\begin{aligned}
f(\mathbf{z}_1|\mathbf{d}_1) &= \frac{f(\mathbf{d}_1|\mathbf{z}_1)\ \int f(\mathbf{z}_1|\mathbf{z}_0) f(\mathbf{z}_0)\, d\mathbf{z}_0}{f(\mathbf{d}_1)} = \frac{f(\mathbf{d}_1|\mathbf{z}_1) f(\mathbf{z}_1)}{f(\mathbf{d}_1)}, \qquad (8)\\
f(\mathbf{z}_2|\mathbf{d}_1, \mathbf{d}_2) &= \frac{f(\mathbf{d}_2|\mathbf{z}_2)\ \int f(\mathbf{z}_2|\mathbf{z}_1) f(\mathbf{z}_1|\mathbf{d}_1)\, d\mathbf{z}_1}{f(\mathbf{d}_2)} = \frac{f(\mathbf{d}_2|\mathbf{z}_2) f(\mathbf{z}_2|\mathbf{d}_1)}{f(\mathbf{d}_2)}, \qquad (9)\\
&\vdots
\end{aligned}
$$

# Or just Bayes' for the assimilation window

$$\boxed{f(\mathbf{z}|\mathbf{d}) = \frac{f(\mathbf{d}|\mathbf{z})f(\mathbf{z})}{f(\mathbf{d})}, \qquad (10)}$$

# Discrete dynamic model with uncertain inputs

$$\mathbf{x}_k = \mathbf{m}(\mathbf{x}_{k-1}, \boldsymbol{\theta}, \mathbf{u}_k, \mathbf{q}_k). \tag{11}$$

- $\mathbf{x}_k$ is the model state.
- $\boldsymbol{\theta}$ are model parameters.
- $\mathbf{u}_k$ are model controls.
- $\mathbf{q}_k$ are model errors.
- Define $\mathbf{x} = (\mathbf{x}_0, \ldots, \mathbf{x}_K)$ as model state over the assimilation window.
- Define $\mathbf{q} = (\mathbf{q}_0, \ldots, \mathbf{q}_K)$ as model errors over the assimilation window.
- Define $\mathbf{u} = (\mathbf{u}_0, \ldots, \mathbf{u}_K)$ as model forcing over the assimilation window.
- Define $\mathbf{z} = (\mathbf{x}, \boldsymbol{\theta}, \mathbf{u}, \mathbf{q})$ as state vector for assimilation problem.

# Parameter estimation vs state estimation

Including the model errors in $\mathbf{z}$ allows us to consider the model and measurement operator as deterministic.

Example:

- Solve for uncertain input parameters.
- Condition on measurements distributed over the assimilation window.

$$\mathbf{y} = \mathbf{g}(\mathbf{z}) = \mathbf{h}\big(\mathbf{m}(\boldsymbol{\theta}, \mathbf{q})\big). \tag{12}$$

$$\mathbf{y} = \mathbf{g}(\mathbf{z}) = \mathbf{h}\big(\mathbf{m}(\mathbf{x}_0, \mathbf{q})\big). \tag{13}$$

# Marginal posterior pdf for perfect models

Nonlinear "perfect" model and measurements

$$\mathbf{y} = \mathbf{g}(\mathbf{z}) \qquad \mathbf{d} = \mathbf{y} + \mathbf{e}$$

Bayesian formulation

$$f(\mathbf{z}, \mathbf{y}|\mathbf{d}) \propto f(\mathbf{d}|\mathbf{y}) f(\mathbf{y}|\mathbf{z}) f(\mathbf{z})$$

Model pdf

$$f(\mathbf{y}|\mathbf{z}) = \delta\big(\mathbf{y} - \mathbf{g}(\mathbf{z})\big)$$

Marginal pdf

$$f(\mathbf{z}|\mathbf{d}) \propto \int f(\mathbf{d}|\mathbf{y}) f(\mathbf{y}|\mathbf{z}) f(\mathbf{z}) d\mathbf{y} = f\big(\mathbf{d}|\mathbf{g}(\mathbf{z})\big) f(\mathbf{z})$$

# Bayes' theorem related to the predicted measurements

**We introduce nonlinearity through the likelihood**

$$f(\mathbf{z}|\mathbf{d}) = \frac{f\big(\mathbf{d}|\mathbf{g}(\mathbf{z})\big)f(\mathbf{z})}{f(\mathbf{d})}. \tag{14}$$

The MAP solution

# Gaussian assumption

**Approximation 4 (Gaussian prior and likelihood)**

*We assume that the prior distributions of the state vector's components* $\mathbf{z}$ *and observation errors* $\boldsymbol{\epsilon}$ *are both Gaussian distributed.*

$$f(\mathbf{z}|\mathbf{d}) \propto \mathsf{exp}\left\{-\mathcal{J}(\mathbf{z})\right\}, \tag{15}$$

# Leads to a cost-function formulation for the MAP solution

**Cost function**

$$\mathcal{J}(\mathbf{z}) = \frac{1}{2}\left(\mathbf{z} - \mathbf{z}^{\mathrm{f}}\right)^{\mathrm{T}} \mathbf{C}_{zz}^{-1} \left(\mathbf{z} - \mathbf{z}^{\mathrm{f}}\right) + \frac{1}{2}\left(\mathbf{g}(\mathbf{z}) - \mathbf{d}\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \left(\mathbf{g}(\mathbf{z}) - \mathbf{d}\right). \tag{16}$$

**The gradient set to zero**

$$\mathbf{C}_{zz}^{-1}\left(\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}\right) + \nabla_{\mathbf{z}}\mathbf{g}\left(\mathbf{z}^{\mathrm{a}}\right) \mathbf{C}_{dd}^{-1} \left(\mathbf{g}\left(\mathbf{z}^{\mathrm{a}}\right) - \mathbf{d}\right) = 0. \tag{17}$$

- There is no explicit solution of the gradient equation.

# Gauss-Newton methods solves for the MAP estimate

**Gauss-Newton iteration**

$$\mathbf{z}^{i+1} = \mathbf{z}^{i} - \gamma^{i}\left(\mathbf{C}_{zz}^{-1} + {\mathbf{G}^{i}}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\mathbf{G}^{i}\right)^{-1}\left(\mathbf{C}_{zz}^{-1}(\mathbf{z}^{i} - \mathbf{z}^{\mathrm{f}}) + {\mathbf{G}^{i}}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\big(\mathbf{g}(\mathbf{z}^{i}) - \mathbf{d}\big)\right). \tag{18}$$

- The incremental formulation is sometimes more convenient.

# Incremental Gauss-Newton methods

**Quadratic cost function for the increments**

$$\mathcal{J}(\delta\mathbf{z}) = \frac{1}{2}\left(\delta\mathbf{z} - \boldsymbol{\xi}^i\right)^{\mathrm{T}} \mathbf{C}_{zz}^{-1} \left(\delta\mathbf{z} - \boldsymbol{\xi}^i\right) + \frac{1}{2}\left(\mathbf{G}^i\delta\mathbf{z} - \boldsymbol{\eta}^i\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \left(\mathbf{G}^i\delta\mathbf{z} - \boldsymbol{\eta}^i\right). \tag{19}$$

with

$$\mathbf{z}^{i+1} = \mathbf{z}^i + \delta\mathbf{z}, \tag{20}$$

$$\boldsymbol{\eta}^i = \mathbf{d} - \mathbf{g}\left(\mathbf{z}^i\right), \tag{21}$$

$$\boldsymbol{\xi}^i = \mathbf{z}^{\mathrm{f}} - \mathbf{z}^i. \tag{22}$$

- Sequence of linear iterates.
- Solved by SC-4DVar, WC-4DVar, and Representer method.

Standard strong constraint 4DVar

# Standard SC-4DVar

Model with initial condition and poorly known parameter

$$\mathbf{x}_0 = \mathbf{x}_0^{\mathrm{f}} + \mathbf{x}_0', \tag{23}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta}^{\mathrm{f}} + \boldsymbol{\theta}', \tag{24}$$

$$\mathbf{x}_{k+1} = \mathbf{m}(\mathbf{x}_k, \boldsymbol{\theta}), \tag{25}$$

Measurements

$$\mathbf{d} = \mathbf{h}(\mathbf{x}) + \mathbf{e}. \tag{26}$$

# Problem formulation

State vector and covariance matrix

$$\mathbf{z} = \begin{pmatrix} \mathbf{x}_0 \\ \boldsymbol{\theta} \end{pmatrix} \quad \text{and} \quad \mathbf{C}_{zz} = \begin{pmatrix} \mathbf{C}_{x_0x_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{\theta\theta} \end{pmatrix}, \tag{27}$$

**SC-4DVar costfunction**

$$\mathcal{J}(\mathbf{z}) = \frac{1}{2}\left(\mathbf{z} - \mathbf{z}^{\mathrm{f}}\right)^{\mathrm{T}} \mathbf{C}_{zz}^{-1} \left(\mathbf{z} - \mathbf{z}^{\mathrm{f}}\right) + \frac{1}{2}\left(\mathbf{h}(\mathbf{x}) - \mathbf{d}\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \left(\mathbf{h}(\mathbf{x}) - \mathbf{d}\right), \tag{28}$$

Solve for initial condition and parameter that minimize the cost function

# Lagrangian formulation

$$
\begin{aligned}
\mathcal{L}(\mathbf{x}_0, \ldots, \mathbf{x}_{K+1}, \boldsymbol{\theta}, \boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{K+1}) = & \frac{1}{2}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{f}}\right)^{\mathrm{T}} \mathbf{C}_{x_0 x_0}^{-1} \left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{f}}\right) \\
& + \frac{1}{2}\left(\boldsymbol{\theta} - \boldsymbol{\theta}^{\mathrm{f}}\right)^{\mathrm{T}} \mathbf{C}_{\theta\theta}^{-1} \left(\boldsymbol{\theta} - \boldsymbol{\theta}^{\mathrm{f}}\right) \\
& + \frac{1}{2}\left(\mathbf{h}(\mathbf{x}) - \mathbf{d}\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \left(\mathbf{h}(\mathbf{x}) - \mathbf{d}\right) \\
& + \sum_{k=0}^{K} \boldsymbol{\lambda}_{k+1}^{\mathrm{T}} \left(\mathbf{x}_{k+1} - \mathbf{m}(\mathbf{x}_k, \boldsymbol{\theta})\right).
\end{aligned} \tag{29}
$$

We include the perfect model by introducing a Lagrangian multiplier $\boldsymbol{\lambda}$.

# Gradient of Lagrangian

$$\nabla_{\mathbf{x}_k}\mathcal{L}(\mathbf{x},\boldsymbol{\theta},\boldsymbol{\lambda}) = \mathbf{H}_k^{\mathrm{T}}\,\mathbf{C}_{dd}^{-1}\left(\mathbf{h}(\mathbf{x})-\mathbf{d}\right) + \boldsymbol{\lambda}_k - \mathbf{M}_{x,k}^{\mathrm{T}}\boldsymbol{\lambda}_{k+1}, \tag{30}$$

$$\nabla_{\mathbf{x}_{K+1}}\mathcal{L}(\mathbf{z},\mathbf{x},\boldsymbol{\lambda}) = \boldsymbol{\lambda}_{K+1}, \tag{31}$$

$$\begin{aligned}\nabla_{\mathbf{x}_0}\mathcal{L}(\mathbf{x},\boldsymbol{\theta},\boldsymbol{\lambda}) &= \mathbf{C}_{zz}^{-1}\left(\mathbf{x}_0-\mathbf{x}_0^{\mathrm{f}}\right) - \mathbf{M}_{x,0}^{\mathrm{T}}\boldsymbol{\lambda}_1 \\ &= \mathbf{C}_{zz}^{-1}\left(\mathbf{x}_0-\mathbf{x}_0^{\mathrm{f}}\right) - \boldsymbol{\lambda}_0,\end{aligned} \tag{32}$$

$$\nabla_{\boldsymbol{\theta}}\mathcal{L}(\mathbf{x},\boldsymbol{\theta},\boldsymbol{\lambda}) = \mathbf{C}_{\theta\theta}^{-1}\left(\boldsymbol{\theta}-\boldsymbol{\theta}^{\mathrm{f}}\right) - \sum_{k=0}^{K}\mathbf{M}_{\theta,k}^{\mathrm{T}}\boldsymbol{\lambda}_{k+1}, \tag{33}$$

$$\nabla_{\boldsymbol{\lambda}_k}\mathcal{L}(\mathbf{x},\boldsymbol{\theta},\boldsymbol{\lambda}) = \mathbf{x}_{k+1} - \mathbf{m}(\mathbf{x}_k,\boldsymbol{\theta}). \tag{34}$$

# Euler-Lagrange equation(s)

Forward model

$$\mathbf{x}_0 = \mathbf{x}_0^{\mathrm{f}} + \mathbf{C}_{x_0 x_0} \boldsymbol{\lambda}_0, \tag{35}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta}^{\mathrm{f}} + \mathbf{C}_{\theta\theta} \sum_{k=0}^{K} \mathbf{M}_{\theta,k}^{\mathrm{T}} \boldsymbol{\lambda}_{k+1}, \tag{36}$$

$$\mathbf{x}_{k+1} = \mathbf{m}(\mathbf{x}_k, \boldsymbol{\theta}), \tag{37}$$

Backward model for the adjoint variable

$$\boldsymbol{\lambda}_{K+1} = 0, \tag{38}$$

$$\boldsymbol{\lambda}_k = \mathbf{M}_{x,k}^{\mathrm{T}} \boldsymbol{\lambda}_{k+1} - \mathbf{H}_k^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \big(\mathbf{h}(\mathbf{x}) - \mathbf{d}\big). \tag{39}$$

Coupled two-point boundary-value problem in time.

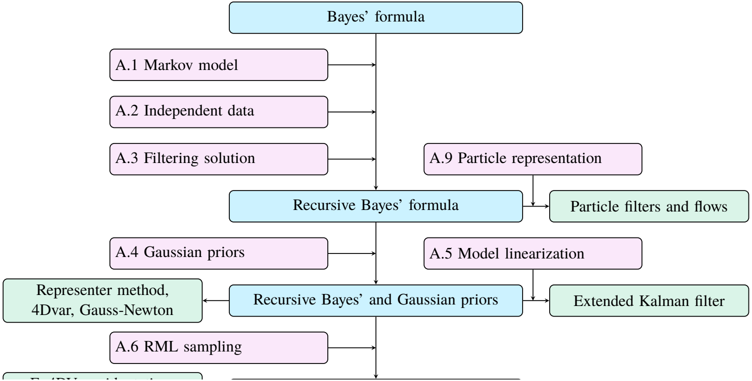
# SC-4DVar algorithm

```
Input: z^f ∈ ℜ^n; d ∈ ℜ^m                      ▷ Prior initial conditions and observations
x_0 = x_0^f                                     ▷ Initialization of x_0
θ = θ^f                                         ▷ Initialization of θ
repeat                                          ▷ Iteration loop
    for k = 0 : K do                            ▷ Integrate forward model
        x_{k+1} = m(x_k, θ)
    end for
    λ_{K+1} = 0
    for k = K : 0 do                            ▷ Integrate backward adjoint model
        λ_k = M^T_{x,k} λ_{k+1} − H^T_k C^{-1}_{dd} (Hx − η)
    end for
    x_0 ← x_0 − γB∇_{x_0} L(x, θ, λ)            ▷ Update x_0 using Eq. (32)
    θ ← θ − γB∇_θ L(x, θ, λ)                    ▷ Update θ using Eq. (33)
until convergence
```

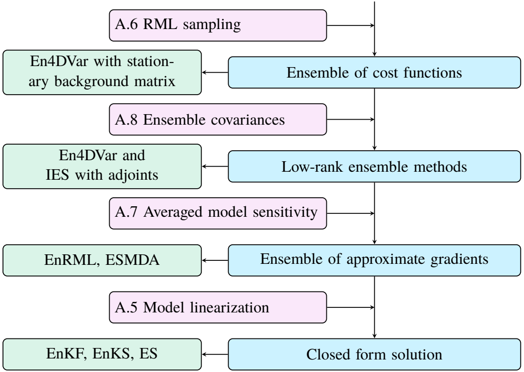
Equivalently, in math notation:

1. Input: $\mathbf{z}^{\mathrm{f}} \in \Re^{n}$; $\mathbf{d} \in \Re^{m}$ ▷ Prior initial conditions and observations
2. $\mathbf{x}_0 = \mathbf{x}_0^{\mathrm{f}}$ ▷ Initialization of $\mathbf{x}_0$
3. $\boldsymbol{\theta} = \boldsymbol{\theta}^{\mathrm{f}}$ ▷ Initialization of $\boldsymbol{\theta}$
4. **repeat** ▷ Iteration loop
5. **for** $k = 0 : K$ **do** ▷ Integrate forward model
6. $\mathbf{x}_{k+1} = \mathbf{m}(\mathbf{x}_k, \boldsymbol{\theta})$
7. **end for**
8. $\boldsymbol{\lambda}_{K+1} = 0$
9. **for** $k = K : 0$ **do** ▷ Integrate backward adjoint model
10. $\boldsymbol{\lambda}_k = \mathbf{M}^{\mathrm{T}}_{x,k} \boldsymbol{\lambda}_{k+1} - \mathbf{H}_k^{\mathrm{T}} \mathbf{C}_{dd}^{-1} (\mathbf{H}\mathbf{x} - \boldsymbol{\eta})$
11. **end for**
12. $\mathbf{x}_0 \leftarrow \mathbf{x}_0 - \gamma \mathbf{B} \nabla_{x_0} \mathcal{L}(\mathbf{x}, \boldsymbol{\theta}, \boldsymbol{\lambda})$ ▷ Update $\mathbf{x}_0$ using Eq. (32)
13. $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \gamma \mathbf{B} \nabla_{\theta} \mathcal{L}(\mathbf{x}, \boldsymbol{\theta}, \boldsymbol{\lambda})$ ▷ Update $\boldsymbol{\theta}$ using Eq. (33)
14. **until** convergence

# Overview of approximations and methods



# Overview of approximations and methods



# Linearization

# Starting point

The gradient in Eq. (17) set to zero

$$\mathbf{C}_{zz}^{-1}\left(\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}\right) + \nabla_{\mathbf{z}}\mathbf{g}\left(\mathbf{z}^{\mathrm{a}}\right)\mathbf{C}_{dd}^{-1}\left(\mathbf{g}\left(\mathbf{z}^{\mathrm{a}}\right) - \mathbf{d}\right) = 0. \tag{40}$$

- There is no explicit solution of the gradient equation.

# Linearization leads to an approximate explicit solution

### Approximation 5 (Linearization)

*Linearize* $\mathbf{g}(\mathbf{z})$ *around the prior estimate* $\mathbf{z}^{\mathrm{f}}$,

$$\mathbf{g}(\mathbf{z}) \approx \mathbf{g}(\mathbf{z}^{\mathrm{f}}) + \mathbf{G}(\mathbf{z} - \mathbf{z}^{\mathrm{f}}), \tag{41}$$

*and approximate the gradient evaluated at the prior estimate*

$$\nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z}) \approx \mathbf{G}^{\mathrm{T}}, \tag{42}$$

*where we have defined*

$$\mathbf{G}^{\mathrm{T}} = \nabla_{\mathbf{z}}\mathbf{g}(\mathbf{z})\big|_{\mathbf{z}=\mathbf{z}^{\mathrm{f}}}. \tag{43}$$

$\mathbf{G}$ *is the tangent-linear operator of* $\mathbf{g}(\mathbf{z})$ *and* $\mathbf{G}^{\mathrm{T}}$ *is its adjoint.*

$$\mathbf{M}_k^{\mathrm{T}} = \nabla_{\mathbf{z}}\mathbf{m}(\mathbf{z})\big|_{\mathbf{z}=\mathbf{z}_k} \quad \text{and} \quad \mathbf{H}^{\mathrm{T}} = \nabla_{\mathbf{m}(\mathbf{z})}\mathbf{h}(\mathbf{m}(\mathbf{z}))\big|_{\mathbf{z}=\mathbf{z}_k}. \tag{44}$$

# Solution by linearization

The linearization in Approx. 5 leads to the following form of (40)

$$\mathbf{C}_{zz}^{-1}\left(\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}\right) + \mathbf{G}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\left(\mathbf{g}(\mathbf{z}^{\mathrm{f}}) + \mathbf{G}(\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}) - \mathbf{d}\right) = 0, \tag{45}$$

or

$$\left(\mathbf{C}_{zz}^{-1} + \mathbf{G}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\mathbf{G}\right)\left(\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}\right) = \mathbf{G}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\left(\mathbf{d} - \mathbf{g}(\mathbf{z}^{\mathrm{f}})\right) = 0. \tag{46}$$

which we can solve for $\mathbf{z}^{\mathrm{a}} - \mathbf{z}^{\mathrm{f}}$ and get

$$\mathbf{z}^{\mathrm{a}} = \mathbf{z}^{\mathrm{f}} + \left(\mathbf{C}_{zz}^{-1} + \mathbf{G}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\mathbf{G}\right)^{-1}\mathbf{G}^{\mathrm{T}}\mathbf{C}_{dd}^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right). \tag{47}$$

# Alternative form using Woodbury

**Woodbury corollaries**

$$\left(\mathbf{C}^{-1} + \mathbf{G}^{\mathrm{T}}\mathbf{D}^{-1}\mathbf{G}\right)^{-1} = \mathbf{C} - \mathbf{C}\mathbf{G}^{\mathrm{T}}(\mathbf{G}\mathbf{C}\mathbf{G}^{\mathrm{T}} + \mathbf{D})^{-1}\mathbf{G}\mathbf{C}, \tag{48}$$

$$(\mathbf{G}^{\mathrm{T}}\mathbf{D}^{-1}\mathbf{G} + \mathbf{C}^{-1})^{-1}\mathbf{G}^{\mathrm{T}}\mathbf{D}^{-1} = \mathbf{C}\mathbf{G}^{\mathrm{T}}(\mathbf{G}\mathbf{C}\mathbf{G}^{\mathrm{T}} + \mathbf{D})^{-1}, \tag{49}$$

Using Eq. (49) we rewrite Eq. (47) as

**Closed form solution by linearization**

$$\mathbf{z}^{\mathrm{a}} = \mathbf{z}^{\mathrm{f}} + \mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}}\left(\mathbf{G}\mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right). \tag{50}$$

# Predicted measurements

Assume linear model and measurement operator $\mathbf{G} = \mathbf{H}\boldsymbol{\mathcal{M}}$ and a state vector $\mathbf{z} = \mathbf{x}_0$.

The predicted measurements then become

$$\mathbf{y} = \mathbf{G}\mathbf{z} = \mathbf{H}\begin{pmatrix} \mathbf{z} \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_K \end{pmatrix} = \mathbf{H}\begin{pmatrix} \mathbf{z} \\ \mathbf{M}_1\mathbf{z} \\ \vdots \\ \mathbf{M}_K \dots \mathbf{M}_1\mathbf{z} \end{pmatrix} = \mathbf{H}\begin{pmatrix} \mathbf{I} \\ \mathbf{M}_1 \\ \vdots \\ \mathbf{M}_K \dots \mathbf{M}_1 \end{pmatrix}\mathbf{z} = \mathbf{H}\boldsymbol{\mathcal{M}}\mathbf{z}, \tag{51}$$

# Nonliner predicted measurements

Assume nonlinear model and measurement operator $\mathbf{g}(\mathbf{z}) = \mathbf{h}(\mathbf{m}(\mathbf{z}))$ and a state vector $\mathbf{z} = \mathbf{x}_0$.

The predicted measurements then becomes

$$\mathbf{y} = \mathbf{g}(\mathbf{z}) = \mathbf{h}\begin{pmatrix} \mathbf{z} \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_K \end{pmatrix} = \mathbf{h}\begin{pmatrix} \mathbf{z} \\ \mathbf{m}_1(\mathbf{z}) \\ \vdots \\ \mathbf{m}_K(\cdots(\mathbf{m}_2(\mathbf{m}_1(\mathbf{z}))\cdots) \end{pmatrix} = \mathbf{h}(\mathbf{m}(\mathbf{z})) \tag{52}$$

Thus, we can compute the update of $\mathbf{z} = \mathbf{x}_0$ using data distributed over the assimilation window.

# Update over the assimilation window

Multiply Eq. (50) by $\boldsymbol{\mathcal{M}}$ to get

$$\boldsymbol{\mathcal{M}}\mathbf{z}^{\mathrm{a}} = \boldsymbol{\mathcal{M}}\mathbf{z}^{\mathrm{f}} + \boldsymbol{\mathcal{M}}\mathbf{C}_{zz}\boldsymbol{\mathcal{M}}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\left(\mathbf{G}\mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right). \tag{53}$$

We can write this equation as

$$\begin{pmatrix} \mathbf{z}^{\mathrm{a}} \\ \mathbf{x}_1^{\mathrm{a}} \\ \vdots \\ \mathbf{x}_K^{\mathrm{a}} \end{pmatrix} = \begin{pmatrix} \mathbf{z}^{\mathrm{f}} \\ \mathbf{x}_1^{\mathrm{f}} \\ \vdots \\ \mathbf{x}_K^{\mathrm{f}} \end{pmatrix} + \begin{pmatrix} \mathbf{C}_{zz} & \dots & \mathbf{C}_{zx_K} \\ \mathbf{C}_{x_1z} & \dots & \mathbf{C}_{x_1x_K} \\ \vdots & \ddots & \vdots \\ \mathbf{C}_{x_Kz} & \dots & \mathbf{C}_{x_Kx_K} \end{pmatrix}\mathbf{H}^{\mathrm{T}}\left(\mathbf{G}\mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right). \tag{54}$$

This gives a smoother update of the model solution over the whole assimilation window

If we are only interested in the solution at the time $t_K$, we can compute

$$\mathbf{x}_K^{\mathrm{a}} = \mathbf{x}_K^{\mathrm{f}} + \left(\mathbf{C}_{x_Kz} \quad \dots \quad \mathbf{C}_{x_Kx_K}\right)\mathbf{H}^{\mathrm{T}}\left(\mathbf{G}\mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right) \tag{55}$$

$$= \mathbf{x}_K^{\mathrm{f}} + \mathbf{C}_{x_Ky}\left(\mathbf{C}_{yy} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d} - \mathbf{g}\left(\mathbf{z}^{\mathrm{f}}\right)\right). \tag{56}$$

## Randomized-maximum-likelihood sampling

# Randomized Maximum Likelihood sampling

**Approximation 6 (RML sampling)**

*In the weakly nonlinear case, we can approximately sample the posterior pdf with Gaussian priors by minimizing the ensemble of cost functions defined by Eq. (58).*

ps: it's really Randomized MAP sampling, or rather just approximate sampling of the posterior pdf.

# RML minimizes an ensemble of cost functions

We define realizations

$$\mathbf{z}_j^{\mathrm{f}} \leftarrow \mathcal{N}(\mathbf{z}^{\mathrm{f}}, \mathbf{C}_{zz}) \quad \text{and} \quad \mathbf{d}_j \leftarrow \mathcal{N}(\mathbf{d}, \mathbf{C}_{dd}) \tag{57}$$

**Ensemble of cost functions**

$$\mathcal{J}(\mathbf{z}_j) = \frac{1}{2}\left(\mathbf{z}_j - \mathbf{z}_j^{\mathrm{f}}\right)^{\mathrm{T}} \mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j - \mathbf{z}_j^{\mathrm{f}}\right) + \frac{1}{2}\left(\mathbf{g}(\mathbf{z}_j) - \mathbf{d}_j\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1}\left(\mathbf{g}(\mathbf{z}_j) - \mathbf{d}_j\right), \tag{58}$$

**Ensemble of gradients set to zero**

$$\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j - \mathbf{z}_j^{\mathrm{f}}\right) + \nabla_{\mathbf{z}}\mathbf{g}\left(\mathbf{z}_j\right)\mathbf{C}_{dd}^{-1}\left(\mathbf{g}(\mathbf{z}_j) - \mathbf{d}_j\right) = 0. \tag{59}$$

Thus, we must solve $N$ independent minimizations.

# Solutions methods using the tangent linear model $\mathbf{G}$

**Ensemble of incremental 4DVars**

$$\mathcal{J}(\delta \mathbf{z}_j) = \frac{1}{2}\left(\delta \mathbf{z}_j - \boldsymbol{\xi}_j^i\right)^{\mathrm{T}} \mathbf{C}_{zz}^{-1} \left(\delta \mathbf{z}_j - \boldsymbol{\xi}_j^i\right) + \frac{1}{2}\left(\mathbf{G}_j^i \delta \mathbf{z}_j - \boldsymbol{\eta}_j^i\right)^{\mathrm{T}} \mathbf{C}_{dd}^{-1} \left(\mathbf{G}_j^i \delta \mathbf{z}_j - \boldsymbol{\eta}_j^i\right). \tag{60}$$

**Ensemble of GN iterations**

$$\mathbf{z}_j^{i+1} = \mathbf{z}_j^i - \gamma \left(\mathbf{C}_{zz}^{-1} + \mathbf{G}_j^{i\,\mathrm{T}} \mathbf{C}_{dd}^{-1} \mathbf{G}_j^i\right)^{-1} \left(\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \mathbf{G}_j^{i\,\mathrm{T}} \mathbf{C}_{dd}^{-1}\left(\mathbf{g}\left(\mathbf{z}_j^i\right) - \mathbf{d}_j\right)\right), \tag{61}$$

**The linear Approximation 5 leads to an Ensemble of Kalman-filter updates**

$$\mathbf{z}_j^{\mathrm{a}} = \mathbf{z}_j^{\mathrm{f}} + \mathbf{C}_{zz} \mathbf{G}_j^{\mathrm{T}} \left(\mathbf{G}_j \mathbf{C}_{zz} \mathbf{G}_j^{\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1} \left(\mathbf{d}_j - \mathbf{g}\left(\mathbf{z}_j^{\mathrm{f}}\right)\right). \tag{62}$$

# Comments

- SC-4DVar and WC-4DVar solves for the MAP estimate over the assimilation window.
- RML sampling using (En)SC-4DVar and (En)WC-4DVar would aim to sample the posterior.
- It is possible to propagate error statistics using ensemble integrations.
- We could have a consistent method using ensemble "background" covariances.
- Still uses the tangent linear and adjoint models.
- What is the benefit of computing update over a finite data-assimilation window?

Replacing the adjoints with an averaged model sensitivity

# Use an averaged model sensitivity to avoid adjoints

**Approximation 7 (Best-fit averaged model sensitivity)**

*Interpret $\mathbf{G}_j$ in Eq. (62) and $\mathbf{G}_j^i$ in Eq. (61) as the sensitivity matrix in linear regression and represent them using the definition*

$$\mathbf{G}_j \approx \mathbf{G} \triangleq \mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}. \tag{63}$$

*We approximate the individual model sensitivities with a common averaged sensitivity used for all realizations.*

# Explanation of the linear regression formula

Define Taylor expansion of $g(x)$ around the ensemble mean

$$g\left(x\right) \approx g\left(\bar{x}\right) + g'\left(\bar{x}\right)\left(x - \bar{x}\right). \tag{64}$$

$$\begin{aligned}
C^{\mathrm{e}}_{xy} &= \overline{\left(x - \bar{x}\right)\left(y - \bar{y}\right)} \\
&= \overline{\left(x - \bar{x}\right)\left(g\left(x\right) - \overline{g\left(x\right)}\right)} \\
&\approx \overline{\left(x - \bar{x}\right)\left(g\left(\bar{x}\right) + g'\left(\bar{x}\right)\left(x - \bar{x}\right) - \overline{\left(g\left(\bar{x}\right) + g'\left(\bar{x}\right)\left(x - \bar{x}\right)\right)}\right)} \\
&= g'\left(\bar{x}\right)\overline{\left(x - \bar{x}\right)^2} \\
&= g'\left(\bar{x}\right)C^{\mathrm{e}}_{xx},
\end{aligned} \tag{65}$$

# Gauss-Newton iterations with averaged model sensitivity

Rewrite the Gauss-Newton iteration in Eq. (61) as

$$\mathbf{z}_j^{i+1} = \mathbf{z}_j^i - \gamma\left(\mathbf{C}_{zz}^{-1} + \mathbf{G}_j^{i\mathrm{T}}\mathbf{C}_{dd}^{-1}\mathbf{G}_j^i\right)^{-1}\left(\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \mathbf{G}_j^{i\mathrm{T}}\mathbf{C}_{dd}^{-1}\left(\mathbf{g}\left(\mathbf{z}_j^i\right) - \mathbf{d}_j\right)\right), \tag{66}$$

$$\approx \mathbf{z}_j^i - \gamma\left(\mathbf{C}_{zz}^{-1} + \mathbf{G}^{i\mathrm{T}}\mathbf{C}_{dd}^{-1}\mathbf{G}^i\right)^{-1}\left(\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \mathbf{G}^{i\mathrm{T}}\mathbf{C}_{dd}^{-1}\left(\mathbf{g}(\mathbf{z}_i) - \mathbf{d}_j\right)\right) \tag{67}$$

$$= \mathbf{z}_j^i - \gamma\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \gamma\mathbf{C}_{zz}\mathbf{G}^{i\mathrm{T}}\left(\mathbf{G}^i\mathbf{C}_{zz}\mathbf{G}^{i\mathrm{T}} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{G}^i\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) - \left(\mathbf{g}\left(\mathbf{z}_j^i\right) - \mathbf{d}_j\right)\right), \tag{68}$$

where we have used the corollaries from Eqs. (48) and (49).

# Expression in terms of covariances

We have

$$\mathbf{G}\mathbf{C}_{zz} = \mathbf{C}_{yz}, \tag{69}$$

$$\mathbf{G}\mathbf{C}_{zz}\mathbf{G}^{\mathrm{T}} = \mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\mathbf{C}_{zy} \neq \mathbf{C}_{yy}. \tag{70}$$

Gauss-Newton Eq. (68)

$$\mathbf{z}_j^{i+1} = \mathbf{z}_j^i - \gamma\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \gamma\mathbf{C}_{zy}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\mathbf{C}_{zy} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) - \left(\mathbf{g}(\mathbf{z}_j^i) - \mathbf{d}_j\right)\right), \tag{71}$$

EKF update Eq. (62)

$$\mathbf{z}_j = \mathbf{z}_j^{\mathrm{f}} + \mathbf{C}_{zy}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\mathbf{C}_{zy} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d}_j - \mathbf{g}\left(\mathbf{z}_j^{\mathrm{f}}\right)\right). \tag{72}$$

# First GN-step equals EKF update

First step of Gauss-Newton Eq. (68) with $\gamma = 1$ and $i = 1$:

$$\mathbf{z}_j^{i+1} = \mathbf{z}_j^i - \left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) + \mathbf{C}_{zy}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\mathbf{C}_{zy} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\left(\mathbf{z}_j^i - \mathbf{z}_j^{\mathrm{f}}\right) - \left(\mathbf{g}(\mathbf{z}_j^i) - \mathbf{d}_j\right)\right), \tag{73}$$

$$= \mathbf{z}_j^{\mathrm{f}} + \mathbf{C}_{zy}\left(\mathbf{C}_{yz}\mathbf{C}_{zz}^{-1}\mathbf{C}_{zy} + \mathbf{C}_{dd}\right)^{-1}\left(\mathbf{d}_j - \mathbf{g}\left(\mathbf{z}_j^{\mathrm{f}}\right)\right). \tag{74}$$

# Ensemble representation of covariances

**Approximation 8 (Ensemble approximation)**

*It is possible to approximately represent a covariance matrix by a low-rank ensemble of states with fewer realizations than the state dimension.*

# Ensemble representation of all covariances

Ensemble matrices

$$\mathbf{Z} = \left(\mathbf{z}_1,\ \mathbf{z}_2,\ \dots\ ,\ \mathbf{z}_N\right), \tag{75}$$

$$\mathbf{D} = \left(\mathbf{d}_1,\ \mathbf{d}_2,\ \dots\ ,\ \mathbf{d}_N\right), \tag{76}$$

$$\boldsymbol{\Upsilon} = \mathbf{g}(\mathbf{Z}). \tag{77}$$

Define the projection $\mathbf{\Pi} \in \Re^{N\times N}$

$$\mathbf{\Pi} = \left(\mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\right)\Big/\sqrt{N-1}. \tag{78}$$

Ensemble anomalies and covariances

$$\mathbf{A} = \mathbf{Z}\mathbf{\Pi}, \qquad \mathbf{C}_{zz} \approx \overline{\mathbf{C}}_{zz} = \mathbf{A}\mathbf{A}^{\mathrm{T}}, \tag{79}$$

$$\mathbf{E} = \mathbf{D}\mathbf{\Pi}, \qquad \mathbf{C}_{dd} \approx \overline{\mathbf{C}}_{dd} = \mathbf{E}\mathbf{E}^{\mathrm{T}}, \tag{80}$$

$$\mathbf{Y} = \boldsymbol{\Upsilon}\mathbf{\Pi}, \qquad \mathbf{C}_{zy} \approx \overline{\mathbf{C}}_{zy} = \mathbf{A}\mathbf{Y}^{\mathrm{T}}. \tag{81}$$

# Ensemble Kalman Filter (EnKF) update

Identical to one iteration of Subspace EnRML with step length $\gamma = 1.0$.

$$\mathbf{Z}^{\mathrm{a}} = \mathbf{Z}^{\mathrm{f}} + \mathbf{A}\mathbf{Y}^{\mathrm{T}}\left(\mathbf{Y}\mathbf{Y}^{\mathrm{T}} + \mathbf{E}\mathbf{E}^{\mathrm{T}}\right)^{-1}\left(\mathbf{D} - \mathbf{g}\left(\mathbf{Z}^{\mathrm{f}}\right)\right) \tag{82}$$

Alternative interpretation using

$$\mathbf{W} = \mathbf{Y}^{\mathrm{T}}\left(\mathbf{Y}\mathbf{Y}^{\mathrm{T}} + \mathbf{E}\mathbf{E}^{\mathrm{T}}\right)^{-1}\left(\mathbf{D} - \mathbf{g}\left(\mathbf{Z}^{\mathrm{f}}\right)\right), \tag{83}$$

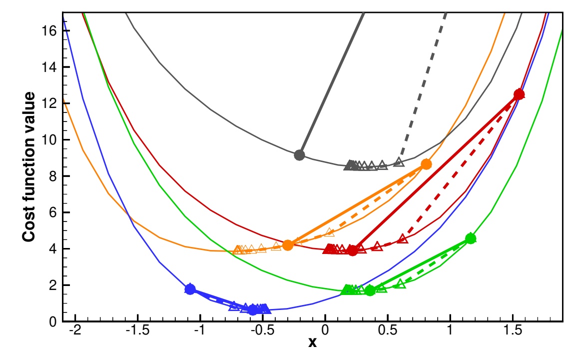
to get

$$\mathbf{Z}^{\mathrm{a}} = \mathbf{Z}^{\mathrm{f}}\left(\mathbf{I} + \mathbf{W}/\sqrt{N-1}\right) \tag{84}$$

But note that

$$\mathbf{Y} = \begin{cases} \mathbf{Y} & \text{for } n \geq N-1 \\ \mathbf{Y}\mathbf{A}^{\dagger}\mathbf{A} & \text{for } n < N-1. \end{cases} \tag{85}$$

# EnKF and EnRML illustration: Non-linear model




- EnRML gets closer to minimum than the linear EnKF update.
- Approximate sampling of posterior pdf.

# An alternative ESMDA uses tapering of likelihood

Approximate sampling of $f(\mathbf{x}|\mathbf{d})$ by gradually introducing the measurements (Neal, 1996)

$$
\begin{aligned}
f(\mathbf{x}|\mathbf{d}) &= f(\mathbf{d}|\mathbf{y}) f(\mathbf{x}) \\
&= f(\mathbf{d}|\mathbf{y})^{\left(\sum_{i=1}^{N} \frac{1}{\alpha_i}\right)} f(\mathbf{x}) \quad \text{with} \quad \sum_{i=1}^{N} \frac{1}{\alpha_i} = 1 \\
&= f(\mathbf{d}|\mathbf{y})^{\frac{1}{\alpha_N}} \cdots f(\mathbf{d}|\mathbf{y})^{\frac{1}{\alpha_2}} f(\mathbf{d}|\mathbf{y})^{\frac{1}{\alpha_1}} f(\mathbf{x})
\end{aligned}
$$

We compute $N$ recursive ES/EnKF steps with "inflated" observation errors.

- Small updates reduce impact of the linear approximation.
- ESMDA is identical to ES in the linear case.
- Remember to resample measurement perturbations for each update step.

# Some publications:

- Ensemble Randomized Maximum Likelihood EnRML (Chen and Oliver, 2013).
- Ensemble DA with multiple updates ESMDA (Emerick and Reynolds, 2013).
- Analysis of iterative ensemble smoothers (Evensen, 2018).
- IES with model errors (Evensen, 2019).
- Ensemble subspace RML (Evensen et al., 2019; Raanes et al., 2019).

# Ensemble subspace RML implementation

We can use the algorithm for ES, ESMDA, and EnRML.

- For ES call once with step length $\gamma = 1$ and $\mathbf{W} = 0$.
- For ESMDA call in each MDA-step with resampled and inflated $\mathbf{D}$, $\gamma = 1$ and $\mathbf{W} = 0$.
- From Evensen et al. (2019); Raanes et al. (2019)

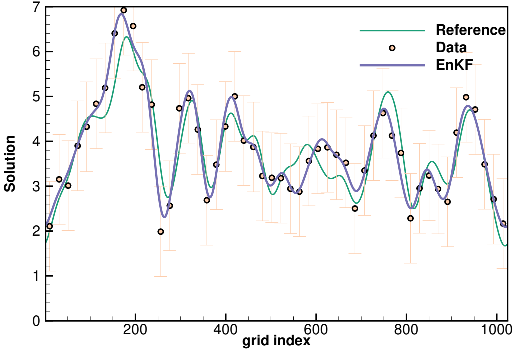
GPL licenced software

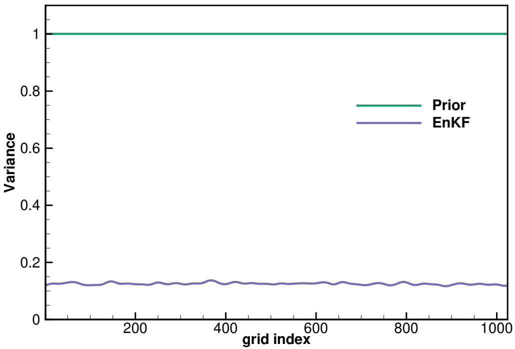
- `https://github.com/equinor/iterative_ensemble_smoother`
- `https://github.com/equinor/ert`
- `https://github.com/Python-Ensemble-Toolbox`

## Impact of measurement-error correlations on analysis update

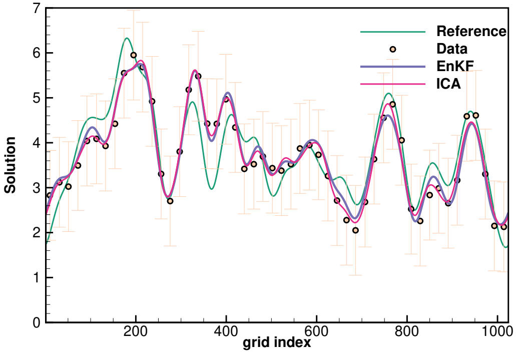
From Evensen (2021)

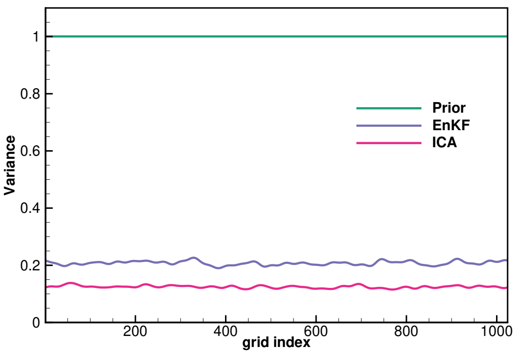
# Analysis with uncorrelated measurement errors



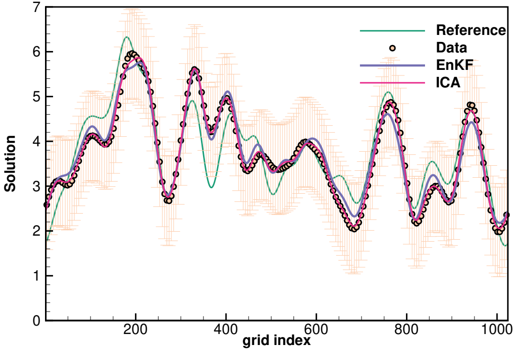



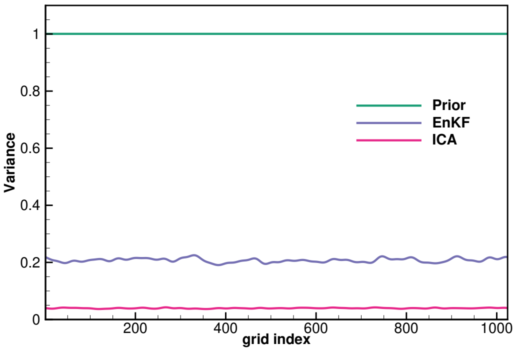
# Impact of correlated measurement errors



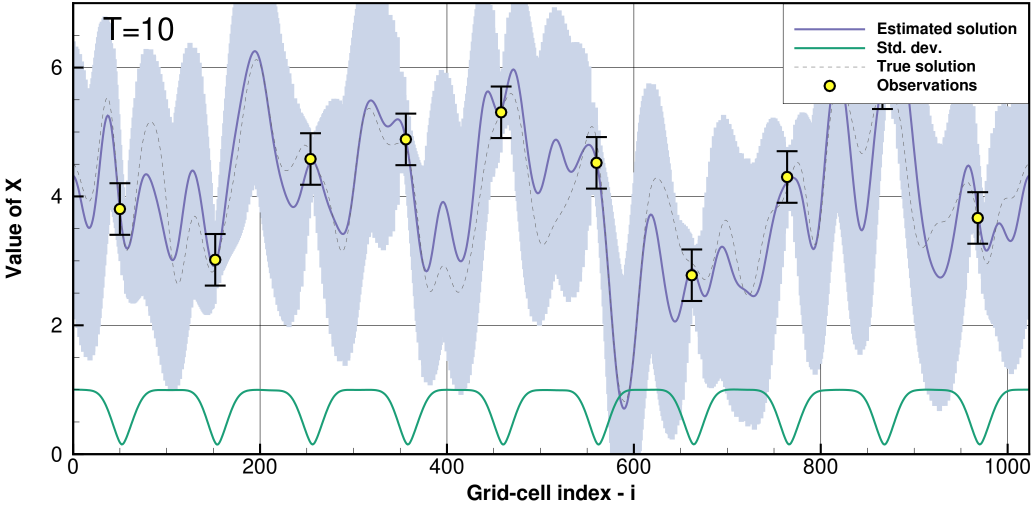
# Correlated errors with many measurements



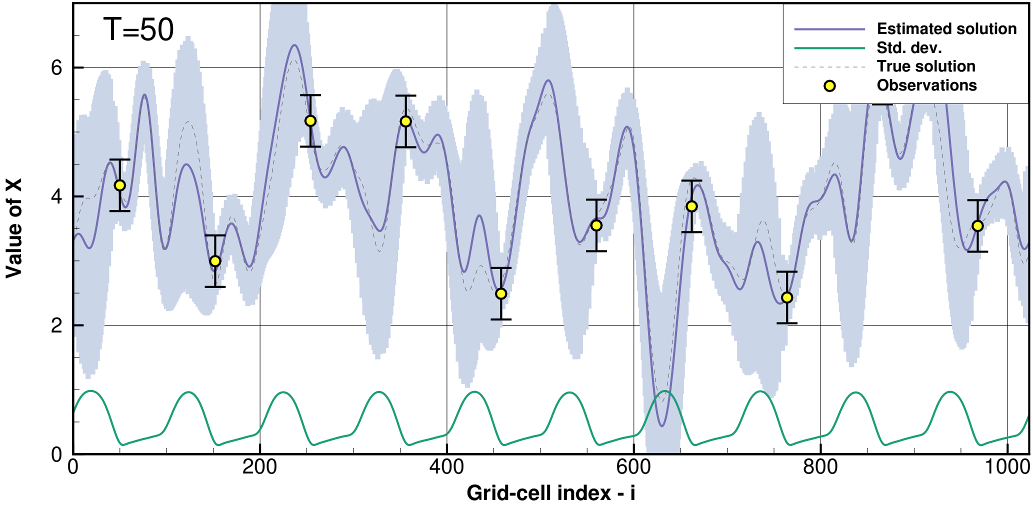



# EnKF for an advection equation

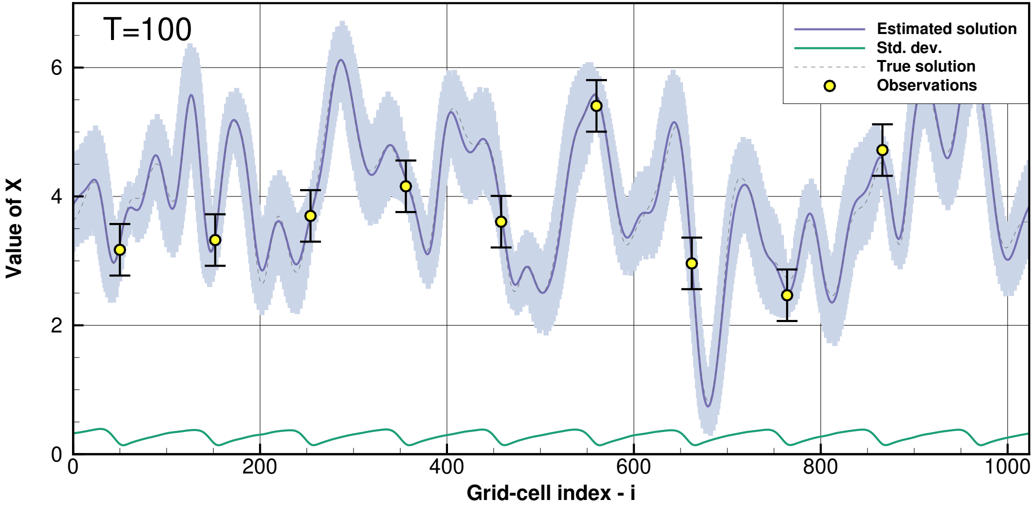
# EnKF with the advection equation after two updates



# EnKF with the advection equation after ten updates



# EnKF with the advection equation after twenty updates



# EnKF with the advection equation: Animation

Animation

# EnKF with the Lorenz equations

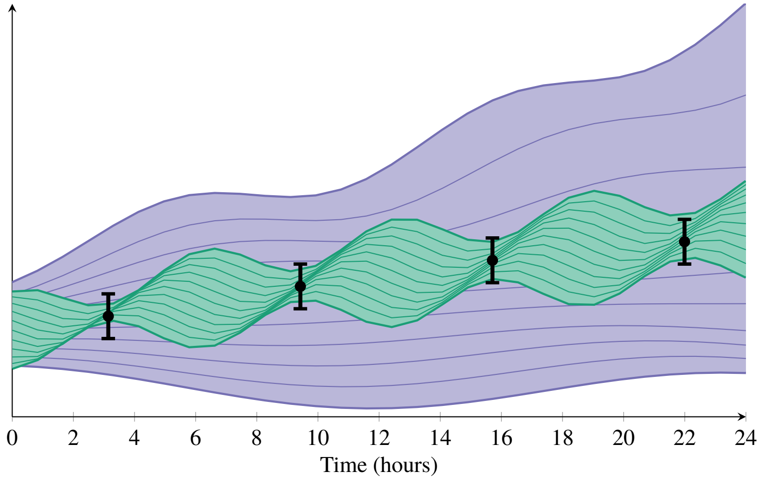
# EnKF with the Lorenz model

$$\frac{\partial x}{\partial t} = \sigma(y - x), \tag{86}$$

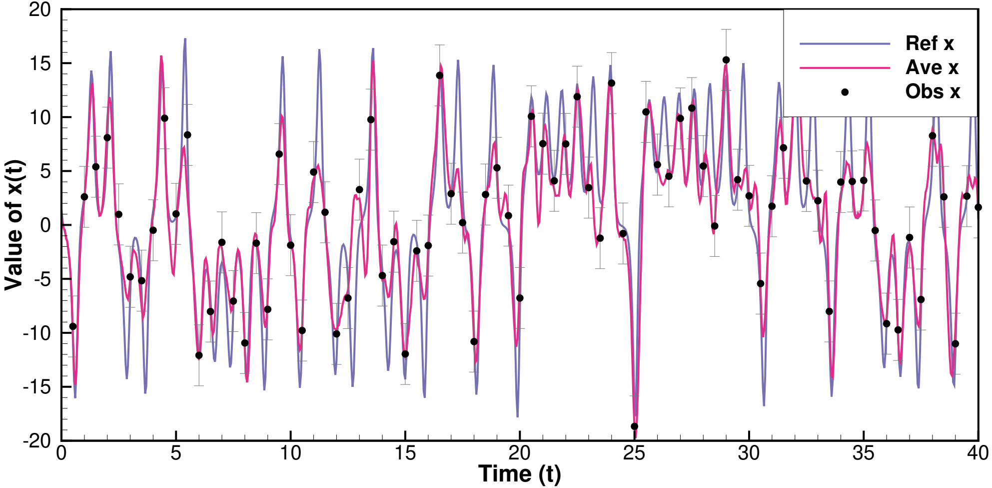
$$\frac{\partial y}{\partial t} = \rho x - y - xz, \tag{87}$$

$$\frac{\partial z}{\partial t} = xy - \beta z. \tag{88}$$

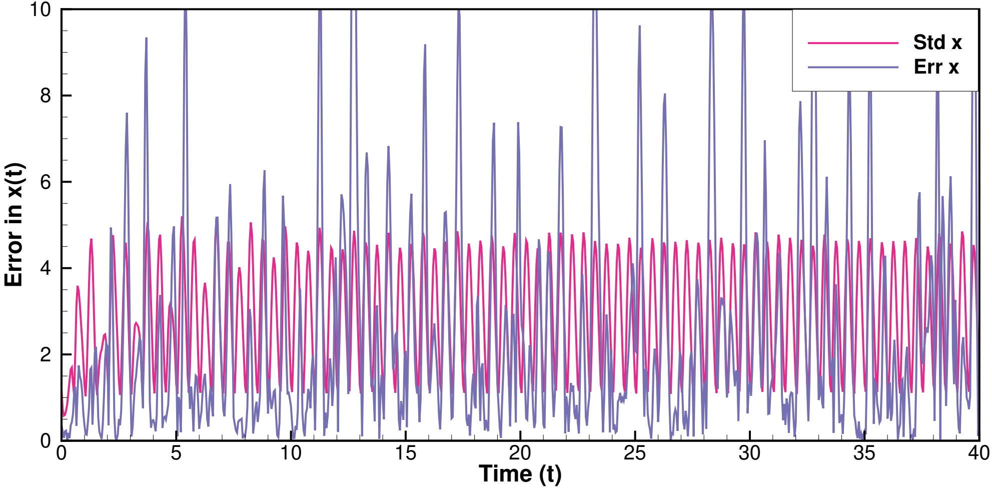
# General smoother formulation



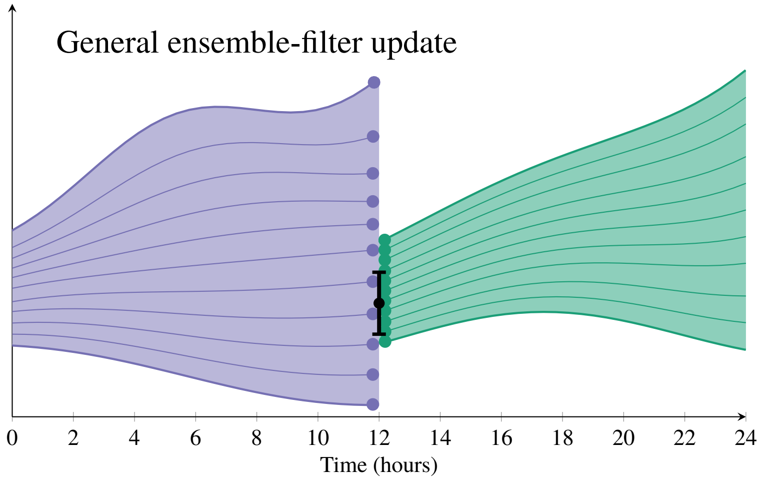
# ES with the Lorenz model: estimate



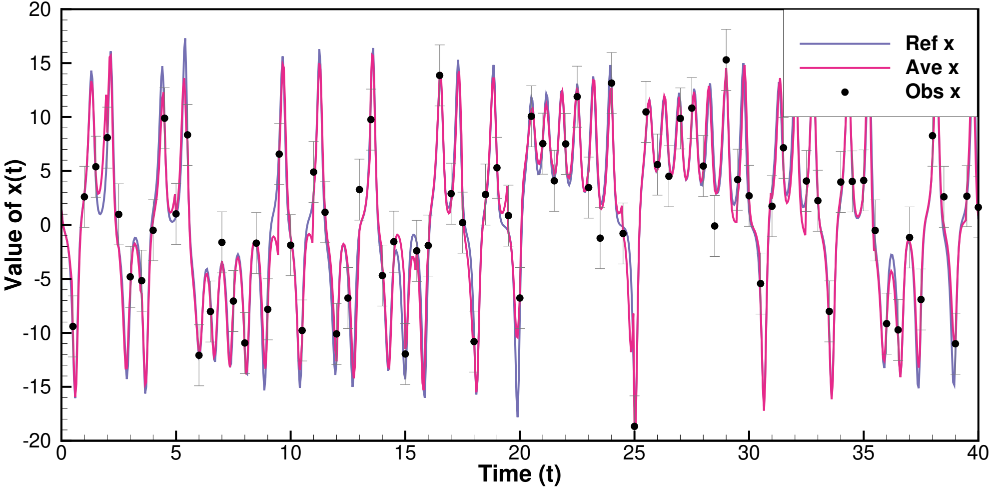
# ES with the Lorenz model: error estimate



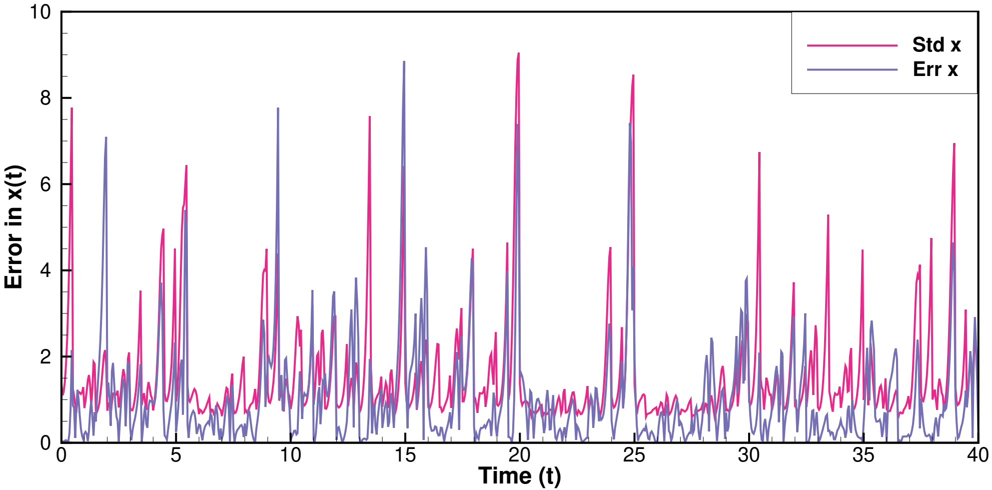
# General filter formulation



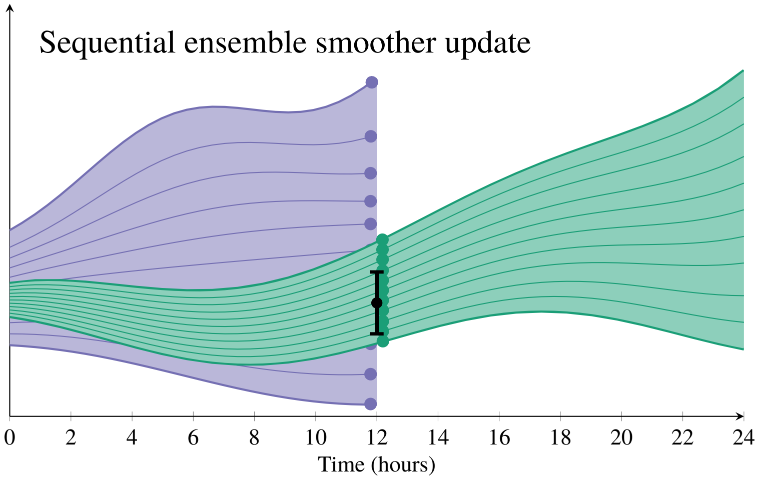
# EnKF with the Lorenz model: estimate



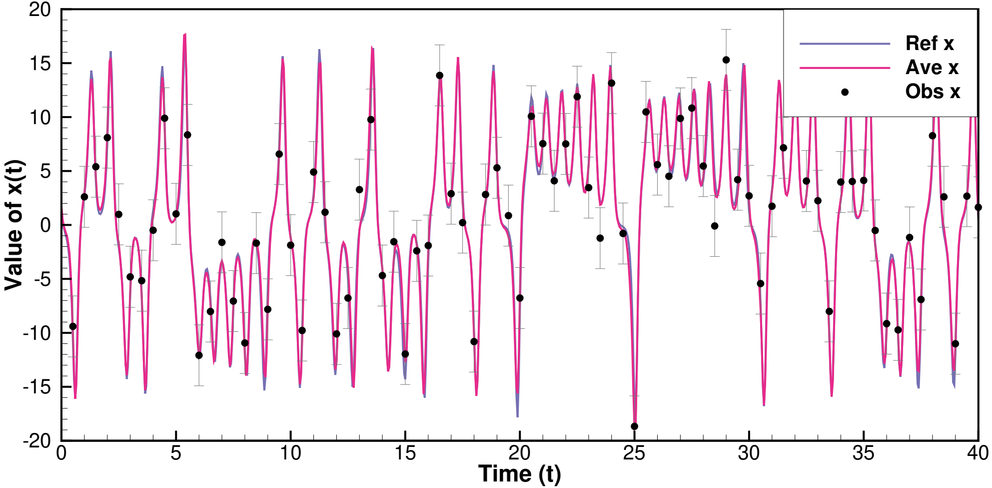
# EnKF with the Lorenz model: error estimate



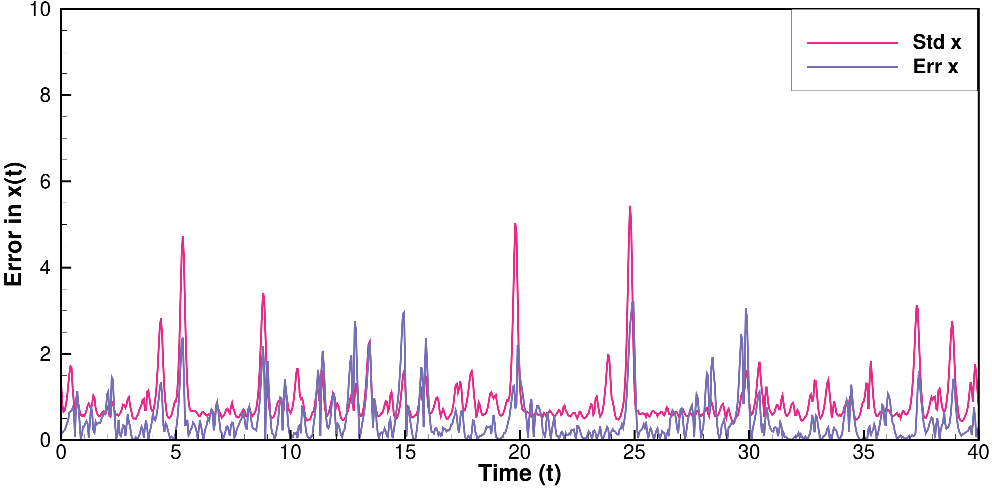
# Recursive smoother formulation



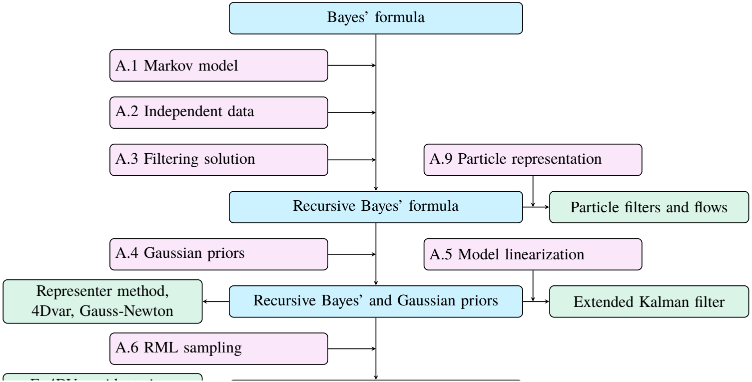
# EnKS with the Lorenz model: estimate



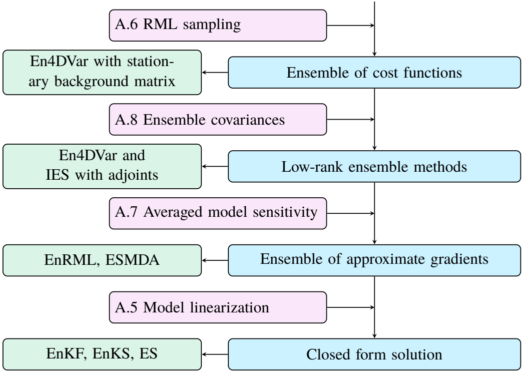
# EnKS with the Lorenz model: error estimate



# Graphic overview



# Graphic overview



# Particle filters for nonlinear data assimilation

**Approximation 9 (Particle representation of the pdfs)**

*It is possible to approximate a probability density function by a finite ensemble of $N$ model states (or particles) as*

$$f(\mathbf{z}) \approx \sum_{j=1}^{N} \frac{1}{N} \delta(\mathbf{z} - \mathbf{z}_j), \tag{89}$$

*where $\delta(\cdot)$ denotes the Dirac-delta function.*

# Examples with model controls

Model forced by controls $\mathbf{u}$

$$\mathbf{x}(t) = \mathbf{m}(\mathbf{x}_0, \boldsymbol{\theta}, \mathbf{u}) = \mathbf{m}(\mathbf{z}) \tag{90}$$

State vector consists of intial conditions $\mathbf{x}_0$, static parameters $\boldsymbol{\theta}$, and controls $\mathbf{u}_k$.

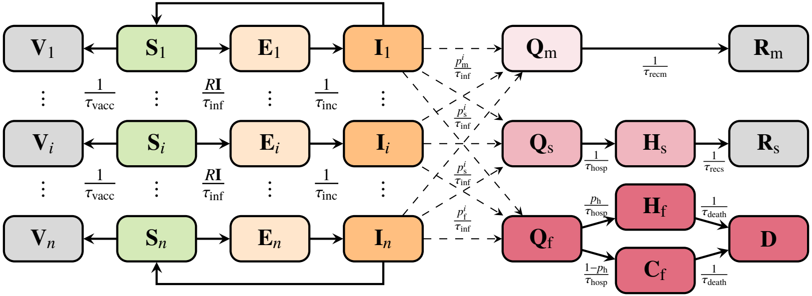
- Covid model
- Petrolium application

We have used ESMDA and subspace EnRML

Covid example

Data assimilation and parameter estimation in an extended SEIR model

# Extended SEIR model




- We add age classes to model age-specific infection and death rates.
- We differentiate between mild, severe, and fatal symptoms.
- We model those with fatal symptoms who die in care homes.

# Extended SEIR model

$$\frac{\partial \mathbf{V}_i}{\partial t} = \frac{v_i}{\tau_{\text{vacc}}}\mathbf{S}_i \tag{91}$$

$$\frac{\partial \mathbf{S}_i}{\partial t} = -\left(\sum_{j=1}^{n}\frac{R_{ij}\mathbf{I}_j}{\tau_{\text{inf}}}\right)\mathbf{S}_i - \frac{v_i}{\tau_{\text{vacc}}}\mathbf{S}_i \tag{92}$$

$$\frac{\partial \mathbf{E}_i}{\partial t} = \left(\sum_{j=1}^{n}\frac{R_{ij}\mathbf{I}_j}{\tau_{\text{inf}}}\right)\mathbf{S}_i - \frac{1}{\tau_{\text{inc}}}\mathbf{E}_i \tag{93}$$

$$\frac{\partial \mathbf{I}_i}{\partial t} = \frac{1}{\tau_{\text{inc}}}\mathbf{E}_i - \frac{1}{\tau_{\text{inf}}}\mathbf{I}_i \tag{94}$$

$$\frac{\partial \mathbf{Q}_{\text{m}}}{\partial t} = \sum_{i=1}^{n_{\text{a}}}\frac{p_{\text{m}}^i}{\tau_{\text{inf}}}\mathbf{I}_i - \frac{1}{\tau_{\text{recm}}}\mathbf{Q}_{\text{m}} \tag{95}$$

$$\frac{\partial \mathbf{Q}_{\text{s}}}{\partial t} = \sum_{i=1}^{n_{\text{a}}}\frac{p_{\text{s}}^i}{\tau_{\text{inf}}}\mathbf{I}_i - \frac{1}{\tau_{\text{hosp}}}\mathbf{Q}_{\text{s}} \tag{96}$$

$$\frac{\partial \mathbf{Q}_{\text{f}}}{\partial t} = \sum_{i=1}^{n_{\text{a}}}\frac{p_{\text{f}}^i}{\tau_{\text{inf}}}\mathbf{I}_i - \frac{1}{\tau_{\text{hosp}}}\mathbf{Q}_{\text{f}} \tag{97}$$

$$\frac{\partial \mathbf{H}_{\text{s}}}{\partial t} = \frac{1}{\tau_{\text{hosp}}}\mathbf{Q}_{\text{s}} - \frac{1}{\tau_{\text{recs}}}\mathbf{H}_{\text{s}} \tag{98}$$

$$\frac{\partial \mathbf{H}_{\text{f}}}{\partial t} = \frac{p_{\text{h}}}{\tau_{\text{hosp}}}\mathbf{Q}_{\text{f}} - \frac{1}{\tau_{\text{death}}}\mathbf{H}_{\text{f}} \tag{99}$$

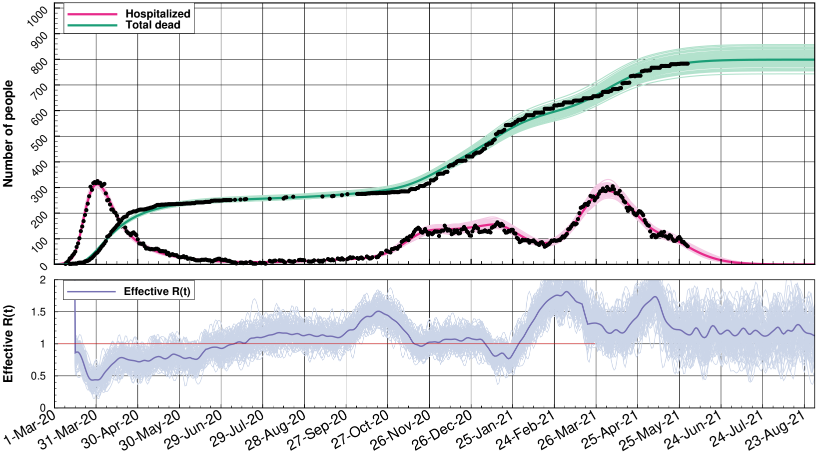
$$\frac{\partial \mathbf{C}_{\text{f}}}{\partial t} = \frac{(1-p_{\text{h}})}{\tau_{\text{hosp}}}\mathbf{Q}_{\text{f}} - \frac{1}{\tau_{\text{death}}}\mathbf{C}_{\text{f}} \tag{100}$$

$$\frac{\partial \mathbf{R}_{\text{m}}}{\partial t} = \frac{1}{\tau_{\text{recm}}}\mathbf{Q}_{\text{m}} \tag{101}$$

$$\frac{\partial \mathbf{R}_{\text{s}}}{\partial t} = \frac{1}{\tau_{\text{recs}}}\mathbf{H}_{\text{s}} \tag{102}$$

$$\frac{\partial \mathbf{D}}{\partial t} = \frac{1}{\tau_{\text{death}}}\mathbf{H}_{\text{f}} + \frac{1}{\tau_{\text{death}}}\mathbf{C}_{\text{f}} \tag{103}$$

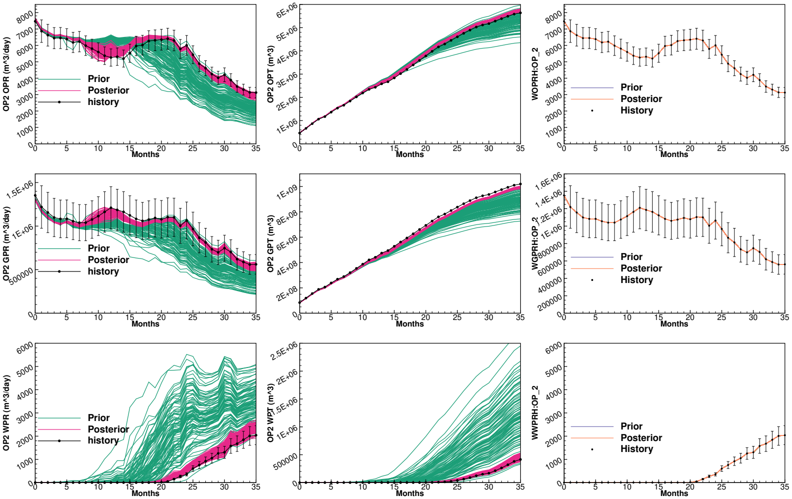
# Covid prediction for Norway




Available from: https://github.com/geirev/Presentations

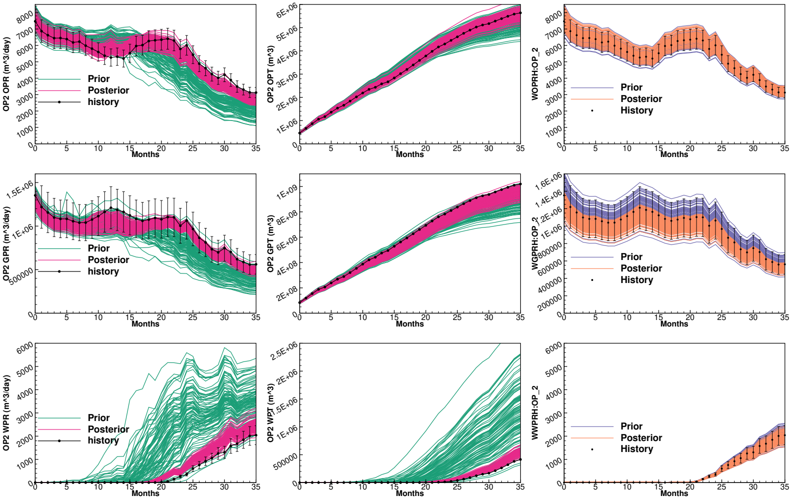
History matching including control uncertainties

# HM without control uncertainties



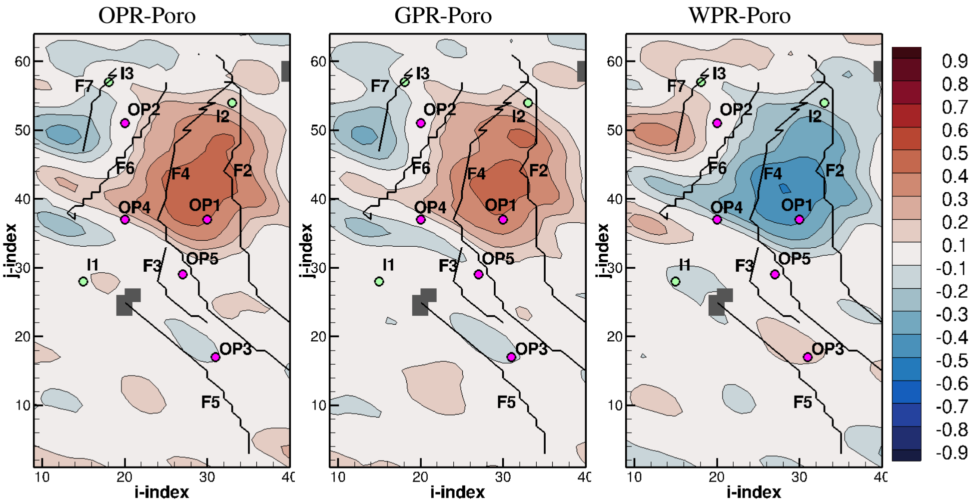

Available from: `https://github.com/geirev/Presentations`

# HM including control uncertainties

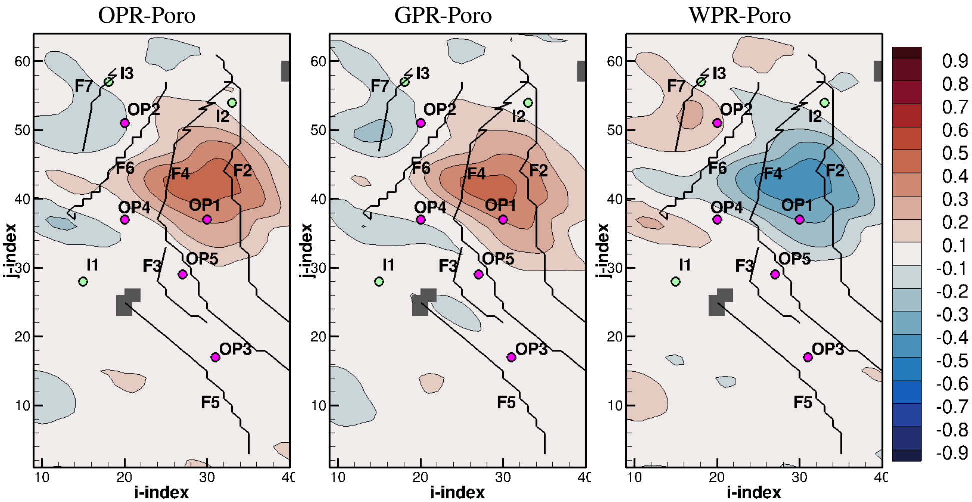


Spurious correlations

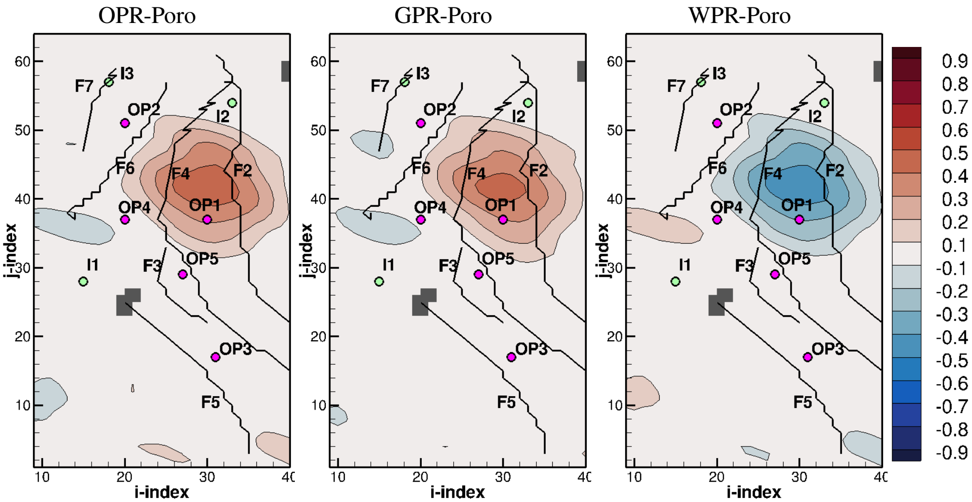
# Ensemble "rate-porosity" correlations $N = 100$



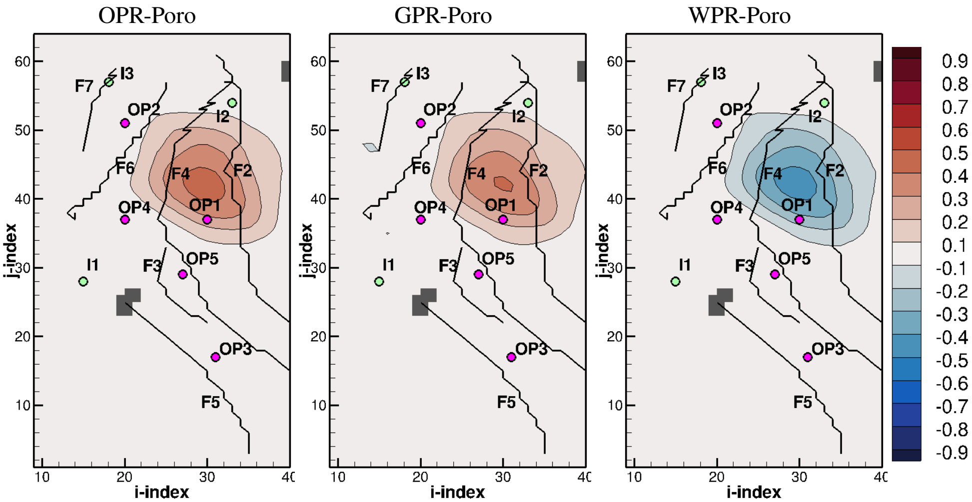
# Ensemble "rate-porosity" correlations $N = 200$



# Ensemble “rate-porosity” correlations $N = 400$



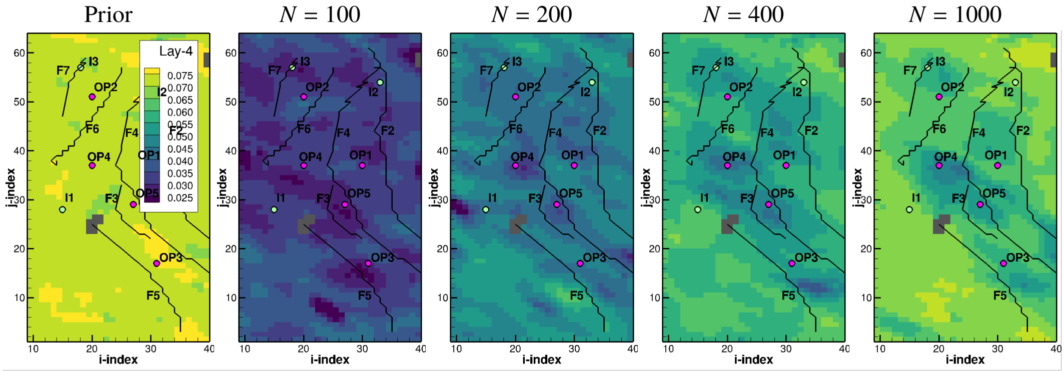
# Ensemble "rate-porosity" correlations $N = 1000$



Impact of spurious correlations on global update

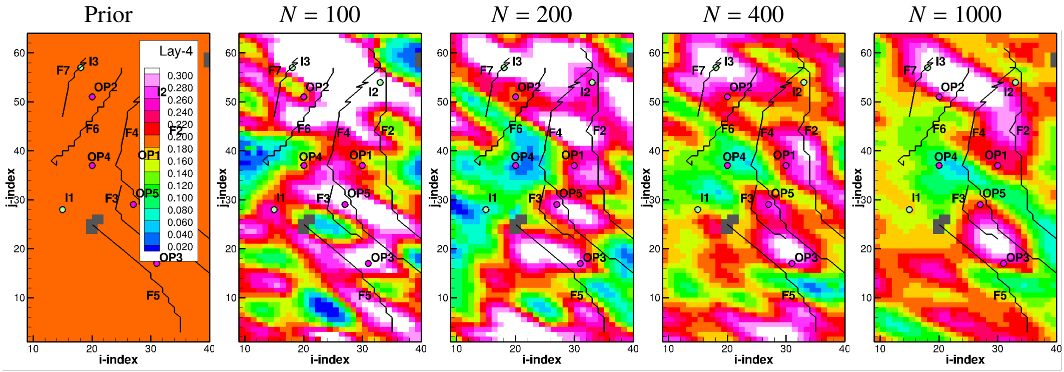
# Global update: Porosity std. dev., layer 4

Diagonal $\mathbf{C}_{dd}$ and including control uncertainties

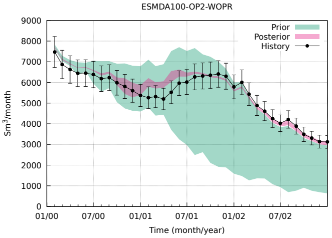


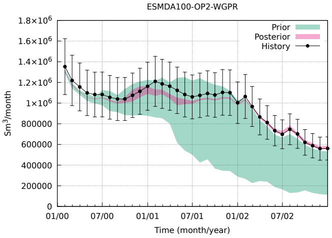
# Global update: Porosity mean, layer 4

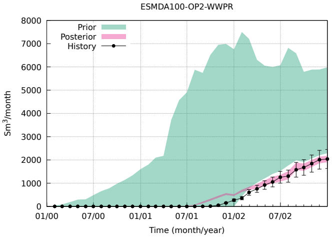
Diagonal $\mathbf{C}_{dd}$ and including control uncertainties

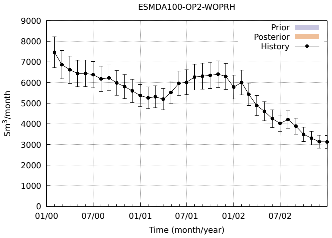


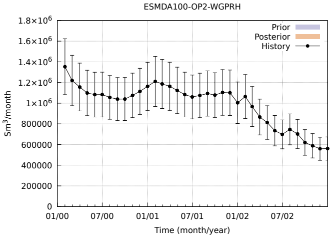
# ESMDA100: Global analysis $N = 100$

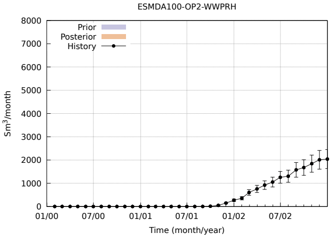


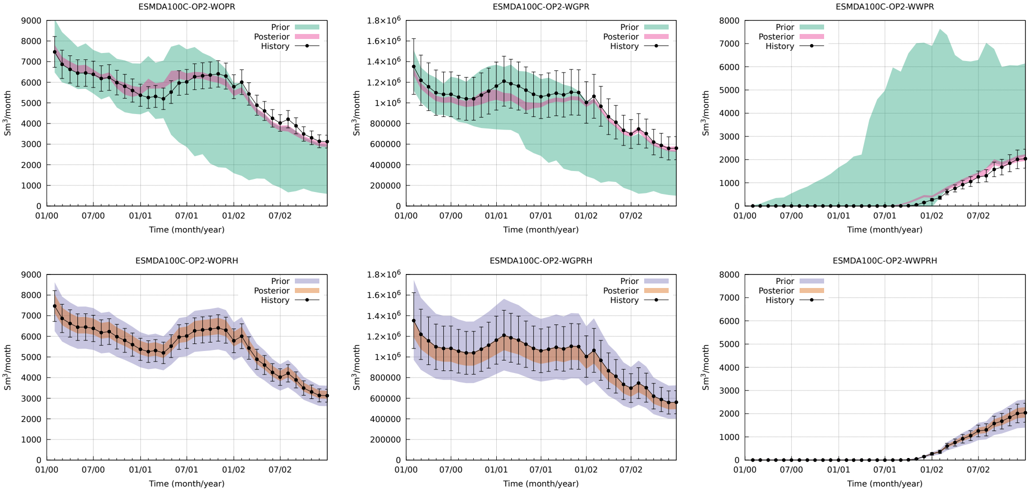



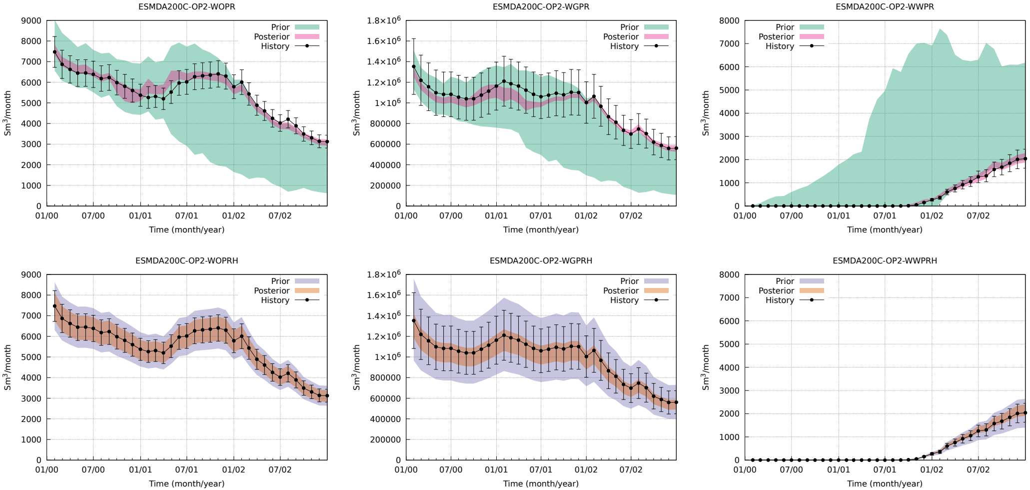




Available from: `https://github.com/geirev/Presentations`

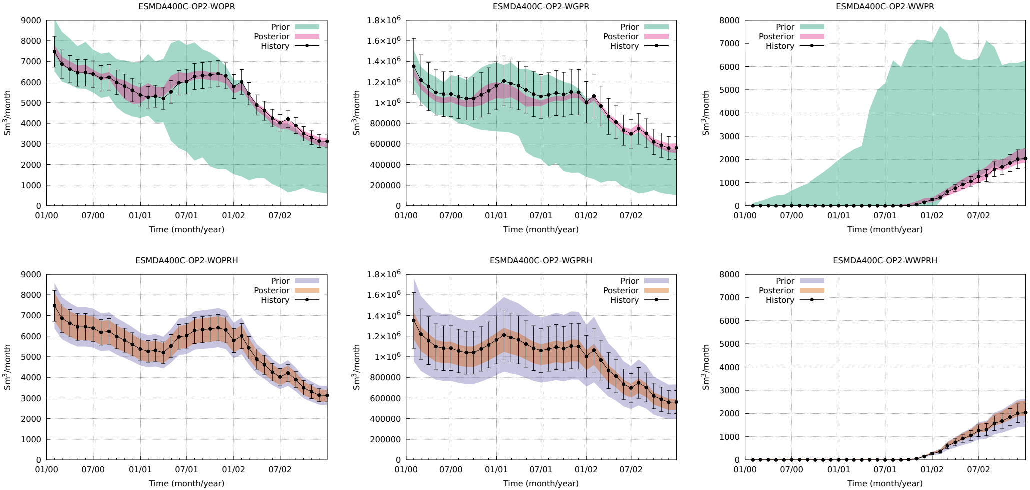
# ESMDA100C: Global analysis $N = 100$



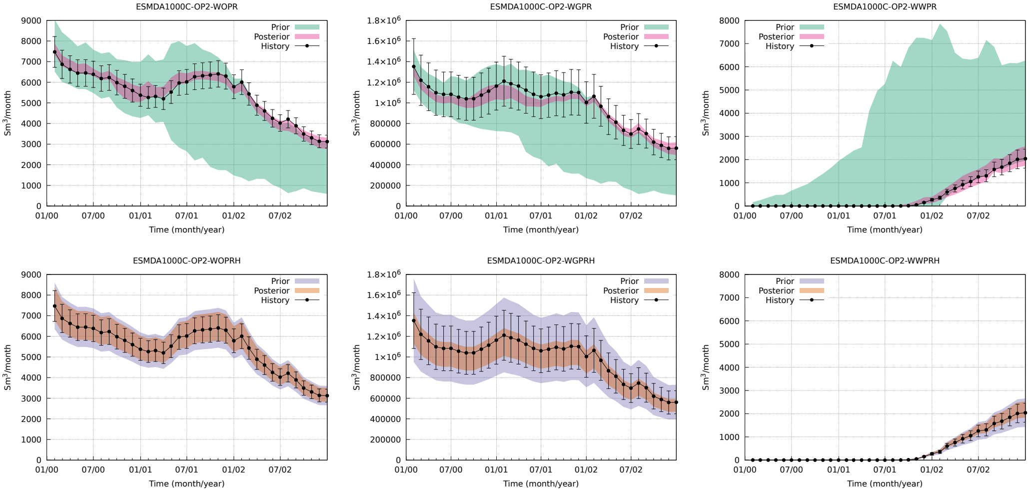
# ESMDA200C: Global analysis $N = 200$



# ESMDA400C: Global analysis $N = 400$



# ESMDA1000C: Global analysis $N = 1000$




Available from: https://github.com/geirev/Presentations

Chen, Y. and D. S. Oliver. Levenberg-Marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. *Computat Geosci*, 17:689–703, 2013. doi:10.1007/s10596-013-9351-5.

Emerick, A. A. and A. C. Reynolds. Ensemble smoother with multiple data assimilation. *Computers and Geosciences*, 55:3–15, 2013. doi:10.1016/j.cageo.2012.03.011.

Evensen, G. Analysis of iterative ensemble smoothers for solving inverse problems. *Computat Geosci*, 22(3):885–908, 2018. doi:10.1007/s10596-018-9731-y.

Evensen, G. Accounting for model errors in iterative ensemble smoothers. *Computat Geosci*, 23(4):761–775, 2019. doi:10.1007/s10596-019-9819-z.

Evensen, G. Formulating the history matching problem with consistent error statistics. *Computat Geosci*, 25:945–970, 2021. doi:10.1007/s10596-021-10032-7.

Evensen, G., P. N. Raanes, A. S. Stordal, and J. Hove. Efficient implementation of an iterative ensemble smoother for data assimilation and reservoir history matching. *Frontiers in Applied Mathematics and Statistics*, 5:47, 2019. doi:10.3389/fams.2019.00047.

Neal, R. M. Sampling from multimodal distributions using tempered transitions. *Statistics and Computing*, 6(4):353–366, 1996. doi:10.1007/BF00143556.

Raanes, P. N., A. S. Stordal, and G. Evensen. Revising the stochastic iterative ensemble smoother. *Nonlin. Processes Geophys*, 26:325–338, 2019. doi:10.5194/npg-2019-10.