Mitglied der Helmholtz-Gemeinschaft

JÜLICH FORSCHUNGSZENTRUM

Short tutorial on data assimilation

23 June 2015 | **Wolfgang Kurtz & Harrie-Jan Hendricks Franssen**

Institute of Bio- and Geosciences IBG-3 (Agrosphere), Forschungszentrum Jülich GmbH

Centre for High-Performance Computing in Terrestrial Systems, Geoverbund ABC/J, Jülich

Data assimilation

Optimal merging of (uncertain) model predictions with (uncertain) measurements.

Data assimilation

Optimal merging of (uncertain) model predictions with (uncertain) measurements.

Models are applied in a variety of earth system disciplines:

- Atmosphere
- Oceanography
- Land surface
- Surface water/ groundwater
- Glaciology
- Radiative transfer models
- Vegetation dynamics/ Biogeochemistry

Why are model predictions uncertain?

Model structural errors:

- Richards equation in land surface models
- Soil respiration in land surface models: simple black-box concept

Why are model predictions uncertain?

Model structural errors:

- Richards equation in land surface models
- Soil respiration in land surface models: simple black-box concept

Parameter errors:

- Soil hydraulic parameters like saturated conductivity or porosity
- Ecosystem parameters like rooting depth

Why are model predictions uncertain?

Model structural errors:

- Richards equation in land surface models
- Soil respiration in land surface models: simple black-box concept

Parameter errors:

- Soil hydraulic parameters like saturated conductivity or porosity
- Ecosystem parameters like rooting depth

Errors in model forcings:

- Precipitation
- Short wave radiation

Why are model predictions uncertain?

Model structural errors:

- Richards equation in land surface models
- Soil respiration in land surface models: simple black-box concept

Parameter errors:

- Soil hydraulic parameters like saturated conductivity or porosity
- Ecosystem parameters like rooting depth

Errors in model forcings:

- Precipitation
- Short wave radiation

Errors in initial conditions:

- Initial soil moisture content
- Carbon pools

Measurement data

- Provide information on model states
- Provide (indirectly) information on parameters and model forcings
- Data are more valuable if they provide information over larger spatial and temporal scales

Measurement data

- Provide information on model states
- Provide (indirectly) information on parameters and model forcings
- Data are more valuable if they provide information over larger spatial and temporal scales

Important limitations

- Information is always incomplete: not everywhere, not always
- Random measurement errors (e.g., instrument precision)
- Systematic measurement errors (e.g., LAI from SMOS)
- Complicated relationship between what is measured and the quantity of interest (e.g., brightness temperature from SMOS and soil moisture content)

Bayes Law

$$\overbrace{p(x \mid y)}^{\text{Posterior}} = \frac{\overbrace{p(x)}^{\text{Prior}}\,\overbrace{p(y \mid x)}^{\text{Likelihood}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

Bayes Law

$$\overbrace{p(x \mid y)}^{\text{Posterior}} = \frac{\overbrace{p(x)}^{\text{Prior}}\,\overbrace{p(y \mid x)}^{\text{Likelihood}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

Often a Gaussian distribution is assumed as prior:

$$p(x) \propto exp\left(-\tfrac{1}{2}(x-\mu)^{\mathsf{T}} C^{-1}(x-\mu)\right)$$

The likelihood in case of a Gaussian assumption is given by:

$$p(y \mid x) \propto exp\left(-\tfrac{1}{2}(y-Hx)^{\mathsf{T}} R^{-1}(y-Hx)\right)$$

Bayes Law

$$\overbrace{p(x \mid y)}^{\text{Posterior}} = \frac{\overbrace{p(x)}^{\text{Prior}}\overbrace{p(y \mid x)}^{\text{Likelihood}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

Often a Gaussian distribution is assumed as prior:

$$p(x) \propto exp\left(-\tfrac{1}{2}(x-\mu)^{\intercal}C^{-1}(x-\mu)\right)$$

The likelihood in case of a Gaussian assumption is given by:

$$p(y \mid x) \propto exp\left(-\tfrac{1}{2}(y-Hx)^{\intercal}R^{-1}(y-Hx)\right)$$

$\Rightarrow$ **Kalman filter and variational DA follow as solutions**

DA methods

- Inverse modelling / Variational DA
- Markov Chain Monte Carlo (MCMC)
- Ensemble Kalman Filter (EnKF)
- Particle Filter (PF)
- Ensemble Kalman Smoother (EnKS)/ Particle Smoother (PS)

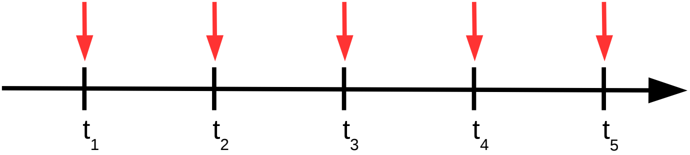
DA methods

- Inverse modelling / Variational DA
- Markov Chain Monte Carlo (MCMC)
- Ensemble Kalman Filter (EnKF)
- Particle Filter (PF)
- Ensemble Kalman Smoother (EnKS)/ Particle Smoother (PS)

Differ with respect to e.g.:

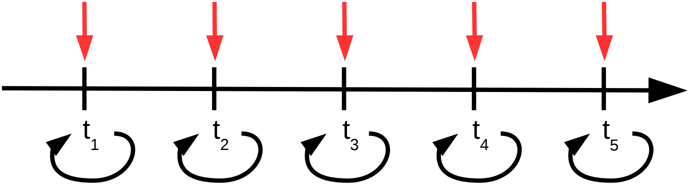
- Temporal treatment of observations
- Intrinsic assumptions
- Computational cost
- Uncertainty quantification

Observations in variational DA/ MCMC



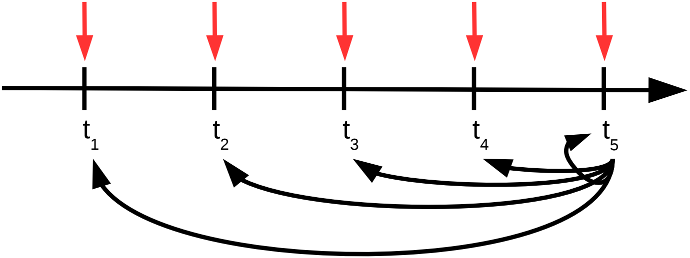
Measurements are processed in batch to update states/parameters for all time steps.

Observations in filters (EnKF/ PF)



Incoming measurements are only used to update states/parameters at current time step.

Observations in smoothers (EnKS/ PS)

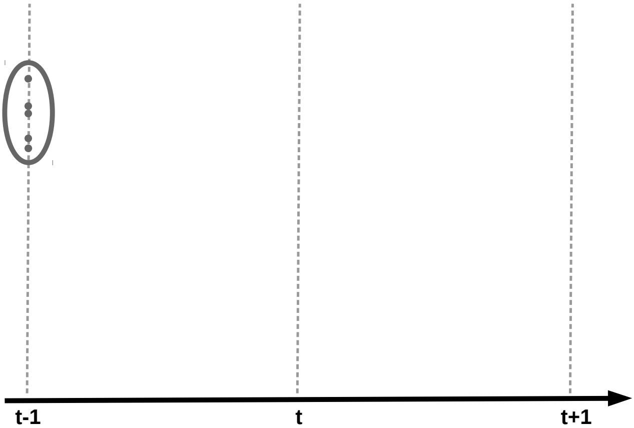


Incoming measurements are used to update states/parameters at current and previous time steps.

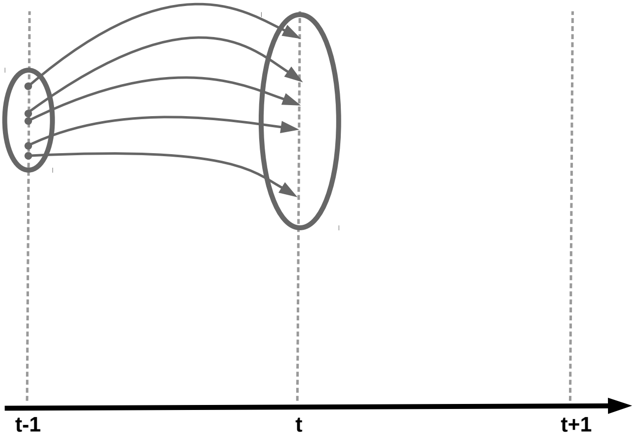
DA methods

- **Markov Chain Monte Carlo**
  very general, but expensive
- **Inverse modelling/ Variational DA**
  Gaussian assumption, uncertainty estimates relatively poor
- **Particle Filter**
  Markovian assumption (sequential), expensive, uncertainty estimates relatively poor related to filter collapse
- **Ensemble Kalman Filter**
  Marcovian assumption (sequential), Gaussian assumption, efficient, better uncertainty estimates than for gradient based inverse
- **Ensemble Kalman Smoother**
  Gaussian assumption, but data over longer time period

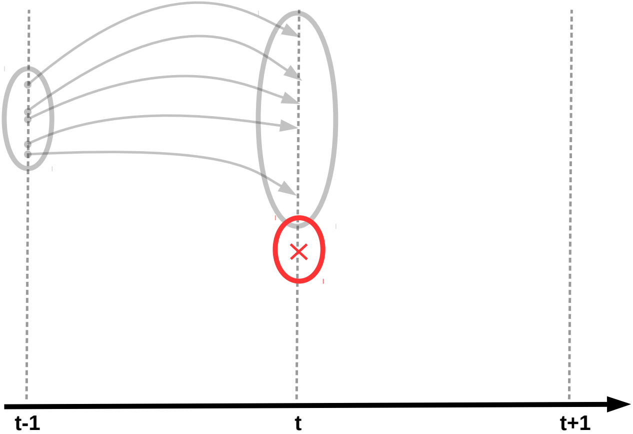
Ensemble Kalman Filter



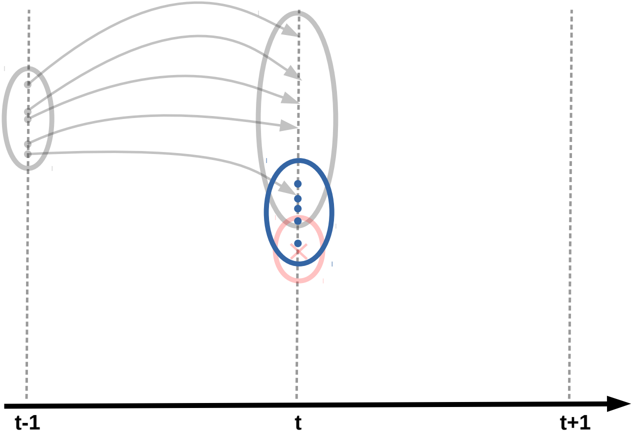
Ensemble Kalman Filter



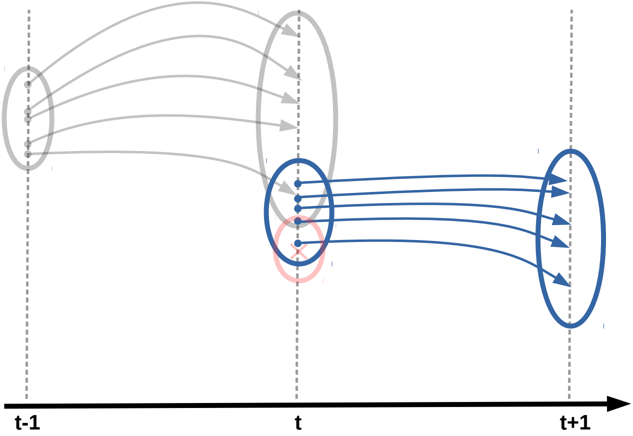
Ensemble Kalman Filter



Ensemble Kalman Filter



Ensemble Kalman Filter



Ensemble Kalman filter - Analysis scheme

Prediction equation

$$x_i^t = M(x_i^{t-1}, p_i, q_i) + \omega_i^t$$

x	=	model states
p	=	model parameters
q	=	model forcings
ω	=	model errors
M	=	(non-linear) forward model
t	=	time
i	=	model realization

Ensemble Kalman filter - Analysis scheme

Measurement equation

$$\tilde{y}_i = Hx_i + \epsilon_i$$

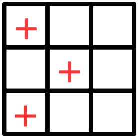
x	=	model states
$\tilde{y}$	=	simulation at observation point
H	=	measurement operator
ϵ	=	measurement error
i	=	model realization

Measurement operator

7	8	9
4	5	6
1	2	3

Model grid

Measurement locations



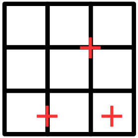
$$
\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \tilde{y}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{bmatrix}
$$

Measurement operator

7	8	9
4	5	6
1	2	3

Model grid

Measurement locations not at cell centers



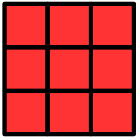
$$\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \tilde{y}_3 \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0.25 & 0.25 \end{bmatrix} \begin{bmatrix} X_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{bmatrix}$$

Measurement operator

7	8	9
4	5	6
1	2	3

Model grid

One remote sensing measurement



$$\begin{bmatrix}\tilde{y}_1\end{bmatrix} = \begin{bmatrix}\frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & \frac{1}{9}\end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{bmatrix}$$

Ensemble Kalman filter - Analysis scheme

Updating equation

$$x_i^+ = x_i^t + K(y - \tilde{y}_i)$$

x	=	model states (predicted)
x	=	model states (updated)
K	=	Kalman gain
y	=	measurement
$\tilde{y}$	=	simulation at observation point
i	=	model realization

Ensemble Kalman filter - Analysis scheme

Kalman gain

$$K = C_{x\tilde{y}}(HC_{x\tilde{y}} + R)^{-1}$$

K	=	Kalman gain
$C_{x\tilde{y}}$	=	covariance matrix of states and simulated measurements
R	=	measurement error covariance matrix

Covariance matrix

7	8	9
4	5	6
1	2	3

Model grid

Measurement locations

+		
	+	
+		

$$C_{x\tilde{y}} = \begin{bmatrix} C_{x_1\tilde{y}_1} & C_{x_1\tilde{y}_2} & C_{x_1\tilde{y}_3} \\ C_{x_2\tilde{y}_1} & C_{x_2\tilde{y}_2} & C_{x_2\tilde{y}_3} \\ C_{x_3\tilde{y}_1} & C_{x_3\tilde{y}_2} & C_{x_3\tilde{y}_3} \\ C_{x_4\tilde{y}_1} & C_{x_4\tilde{y}_2} & C_{x_4\tilde{y}_3} \\ C_{x_5\tilde{y}_1} & C_{x_5\tilde{y}_2} & C_{x_5\tilde{y}_3} \\ C_{x_6\tilde{y}_1} & C_{x_6\tilde{y}_2} & C_{x_6\tilde{y}_3} \\ C_{x_7\tilde{y}_1} & C_{x_7\tilde{y}_2} & C_{x_7\tilde{y}_3} \\ C_{x_8\tilde{y}_1} & C_{x_8\tilde{y}_2} & C_{x_8\tilde{y}_3} \\ C_{x_9\tilde{y}_1} & C_{x_9\tilde{y}_2} & C_{x_9\tilde{y}_3} \end{bmatrix}$$

Covariance matrix

7	8	9
4	5	6
1	2	3

Model grid

Measurement locations

+		
	+	
+		

$$C_{x\tilde{y}} = \begin{bmatrix} C_{x_1\tilde{y}_1} & C_{x_1\tilde{y}_2} & C_{x_1\tilde{y}_3} \\ C_{x_2\tilde{y}_1} & C_{x_2\tilde{y}_2} & C_{x_2\tilde{y}_3} \\ C_{x_3\tilde{y}_1} & C_{x_3\tilde{y}_2} & C_{x_3\tilde{y}_3} \\ C_{x_4\tilde{y}_1} & C_{x_4\tilde{y}_2} & C_{x_4\tilde{y}_3} \\ C_{x_5\tilde{y}_1} & C_{x_5\tilde{y}_2} & C_{x_5\tilde{y}_3} \\ C_{x_6\tilde{y}_1} & C_{x_6\tilde{y}_2} & C_{x_6\tilde{y}_3} \\ C_{x_7\tilde{y}_1} & C_{x_7\tilde{y}_2} & C_{x_7\tilde{y}_3} \\ C_{x_8\tilde{y}_1} & C_{x_8\tilde{y}_2} & C_{x_8\tilde{y}_3} \\ C_{x_9\tilde{y}_1} & C_{x_9\tilde{y}_2} & C_{x_9\tilde{y}_3} \end{bmatrix}$$

$$HC_{x\tilde{y}} = \begin{bmatrix} C_{\tilde{y}_1\tilde{y}_1} & C_{\tilde{y}_1\tilde{y}_2} & C_{\tilde{y}_1\tilde{y}_3} \\ C_{\tilde{y}_2\tilde{y}_1} & C_{\tilde{y}_2\tilde{y}_2} & C_{\tilde{y}_2\tilde{y}_3} \\ C_{\tilde{y}_3\tilde{y}_1} & C_{\tilde{y}_3\tilde{y}_2} & C_{\tilde{y}_3\tilde{y}_3} \end{bmatrix}$$

$$R = \begin{bmatrix} C_{\epsilon_1\epsilon_1} & C_{\epsilon_1\epsilon_2} & C_{\epsilon_1\epsilon_3} \\ C_{\epsilon_2\epsilon_1} & C_{\epsilon_2\epsilon_2} & C_{\epsilon_2\epsilon_3} \\ C_{\epsilon_3\epsilon_1} & C_{\epsilon_3\epsilon_2} & C_{\epsilon_3\epsilon_3} \end{bmatrix}$$

Kalman gain

- Kalman gain weigths model prediciton uncertainty and measurement uncertainty. For a scalar (one point):

$$K = \frac{\sigma^2_{sim}}{\sigma^2_{sim} + \sigma^2_{obs}}$$

- Model prediction uncertainty estimated by covariance matrix $C_{x\tilde{y}}$ (from the ensemble)
- Kalman gain matrix also determines how a measurement affects the surroundings and corrects surrounding states:
  - Depending on the strength of spatial correlation, a measurement might correct the states in the neighbourhood strongly, or only weakly
  - Spatial correlations depend on model physics, but also correlations of static parameters like land use or soil properties

EnKF update

Filter performance dependent on:

- Relation between model uncertainty and measurement uncertainty
- Update frequency
- Ensemble size
- Amount of observations

Joint state-parameter update

- Measurement data can also be used to update model parameters jointly with the states

Joint state-parameter update

- Measurement data can also be used to update model parameters jointly with the states
- Parameters are appended to the state vector:

$$x = \begin{bmatrix} s \\ p \end{bmatrix}$$

x	=	state-parameter vector
s	=	state vector
p	=	parameter vector

Joint state-parameter update

- Measurement data can also be used to update model parameters jointly with the states
- Parameters are appended to the state vector:

  $$x = \begin{bmatrix} s \\ p \end{bmatrix}$$

x	=	state-parameter vector
s	=	state vector
p	=	parameter vector

- Covariance matrix $C_{x\tilde{y}}$ then also contains covariances between states and parameters

Joint state-parameter update

- Measurement data can also be used to update model parameters jointly with the states
- Parameters are appended to the state vector:

$$x = \begin{bmatrix} s \\ p \end{bmatrix}$$

x	=	state-parameter vector
s	=	state vector
p	=	parameter vector

- Covariance matrix $C_{x\tilde{y}}$ then also contains covariances between states and parameters
- Measurements are used to update parameters indirectly

Data assimilation software

Data assimilation software

- Data Assimilation Research Testbed (DART)
  https://www.image.ucar.edu/DAReS/DART/
- Parallel Data Assimilation Framework (PDAF)
  http://pdaf.awi.de/trac/wiki
- OpenDA
  http://www.openda.org/joomla/index.php

Data assimilation software

- Data Assimilation Research Testbed (DART)
  https://www.image.ucar.edu/DAReS/DART/
- Parallel Data Assimilation Framework (PDAF)
  http://pdaf.awi.de/trac/wiki
- OpenDA
  http://www.openda.org/joomla/index.php

Differ with respect to e.g.:

- Implementation strategy (Fortran, Java, ...)
- Available filter methods
- Model coupling
- Parallelism
- Additional utilities (localization, covariance inflation, measurement operators,...)

Coupling to DA software

User needs to link model with DA and provide certain functionality

- Allow ensemble propagation of different model realizations
- Extract state(-parameter) vector from model output
- Provide measurements and measurement operators
- Redirect updated states vector (parameters) to model input for next time step

Coupling to DA software

Offline coupling

- Data transfer between model and DA module via input/output files
- Requires utilties to read/write the required input/output files
- Program could be proprietary (no source code needed)
- Easy to implement
- Performance degradation due to high I/O

Coupling to DA software

Online coupling

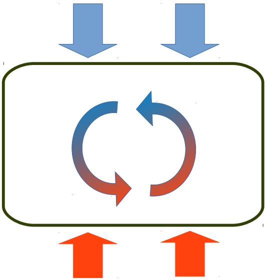
- Data transfer between model and DA module via main memory
- DA module is wrapped around program
- Source code required
- Programming effort depends on model
- Usually faster than offline coupling

Lorenz 63 system

$$\frac{dx}{dt} = \sigma(y - x) \qquad (1)$$

$$\frac{dy}{dt} = x(\rho - z) - y \qquad (2)$$

$$\frac{dz}{dt} = xy - \beta z \qquad (3)$$



x : convective flow

y : horizontal temperature distribution

z : vertical temperature distribution

σ : viscosity / thermal conductivity

ρ : temperature difference top/bottom

β : width to height ratio of cell